



HAL
open science

Contribution à la régulation coopérative des espaces conflictuels entre véhicules autonomes

Alexandre Lombard

► **To cite this version:**

Alexandre Lombard. Contribution à la régulation coopérative des espaces conflictuels entre véhicules autonomes. Automatique. Université Bourgogne Franche-Comté, 2017. Français. NNT : 2017UBFCA020 . tel-01868191

HAL Id: tel-01868191

<https://theses.hal.science/tel-01868191>

Submitted on 5 Sep 2018

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.

SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques

UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

Contribution à la régulation coopérative des espaces conflictuels entre véhicules autonomes

Modèle et prototype de l'intersection autonome

 ALEXANDRE LOMBARD

SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

N° 8 | 0 | 8 | 7 | 1

THÈSE présentée par

ALEXANDRE LOMBARD

pour obtenir le

Grade de Docteur de

l'Université de Technologie de Belfort-Montbéliard

Spécialité : **Informatique**

Contribution à la régulation coopérative des espaces conflictuels entre véhicules autonomes

Modèle et prototype de l'intersection autonome

Unité de Recherche :
Laboratoire Électronique, Informatique et Image (Le2i)

Soutenue publiquement le 11 décembre 2017 devant le Jury composé de :

ANDRES PEREZ-URIBE	Président	Professeur, IICT
OLIVIER SIMONIN	Rapporteur	Professeur des Universités, CITI - Inria Lab.
NOUR-EDDIN EL FAOUZI	Rapporteur	Professeur des Universités, IFSTTAR - LICIT
SÉBASTIEN GLASER	Examineur	Professeur des Universités, IFSTTAR - LIVIC
OUSSAMA BARAKAT	Examineur	Maître de conférences, UFR Sciences et Techniques
RACHID BOUYEKHF	Examineur	Maître de conférences, UTBM
ABDELLAH EL MOUDNI	Directeur de thèse	Professeur des Universités, UTBM
ABDELJALIL ABBAS-TURKI	Co-directeur de thèse	Maître de conférences, UTBM

REMERCIEMENTS

Je tiens tout d'abord à remercier mon directeur de thèse, le Professeur Abdellah El Moudni pour m'avoir permis de mener mes recherches, tout en me laissant une grande liberté dans mon travail. Je remercie tout particulièrement Abdeljalil Abbas-Turki, mon co-directeur de thèse. Sans son encadrement et ses conseils, ce manuscrit n'aurait pas pu exister. Son expérience, ses encouragements, son imagination, et surtout, nos discussions, m'ont permis de développer mes idées et mon envie d'approfondir le sujet.

Je remercie les Professeurs Olivier Simonin et Nour-Eddin El Faouzi pour avoir accepté de rapporter sur ce travail. Je remercie également le Professeur Sébastien Glaser, le Professeur Andres Perez-Urbe, le Maître de conférences HDR Oussama Barakat, ainsi que le Maître de conférences HDR Rachid Bouyekhf, pour m'avoir fait l'honneur d'accepter d'être examinateurs de ma thèse.

Je remercie également toutes les personnes avec lesquelles j'ai pu travailler au laboratoire pour leur bonne humeur et leur sympathie, en particulier Olivier Lamotte, Nicolas Gaud, Philippe Descamps, Yassine Ruichek, Franck Gechter, Stéphane Galland, Fabrice Lauri, Vincent Hilaire, Abder Koukam, Ariane Glatigny, et aussi tout ceux qui ont partagé mon bureau et sont devenus des amis, Florent Perronnet, Thomas Martinet, Cédric Boittin, Charles Gaumet, Jocelyn Buisson, Lionel Corneglio, Thomas Piotrowski, George Georges, et bien d'autres que je n'oublie pas.

Enfin, je tiens à remercier ma soeur Aurianne, mes parents Eric et Nathalie, ainsi que toute ma famille, qui m'ont soutenu et n'ont cessé de m'encourager dans la poursuite de mes études et de mon travail.

SOMMAIRE

I	Contexte	1
1	Introduction générale	3
1.1	Contexte	4
1.2	Objectifs de la thèse	5
1.3	Plan de la thèse	6
2	Intersection autonome et coopérative	7
2.1	Définitions et principes généraux	8
2.1.1	Infrastructure routière et intersection	8
2.1.2	Mesures du trafic	9
2.2	Régulation du trafic aux intersections	9
2.2.1	Approche classique	10
2.2.1.1	Feux tricolores à cycle fixe	10
2.2.1.2	Feux tricolores adaptatifs	10
2.2.1.3	Discussion et perspectives	11
2.2.2	Approche coopérative	12
2.2.2.1	Architecture décentralisée	13
2.2.2.2	Architecture centralisée	14
2.2.2.3	Architecture semi-centralisée	17
2.2.2.4	Discussion	17
2.2.3	Politiques de régulation coopérative	19
2.2.3.1	Premier arrivé, premier servi	19
2.2.3.2	Programmation dynamique, meta-heuristiques	19
2.2.3.3	Politique d'évacuation distribuée	19
2.2.4	Discussion	20
2.3	Véhicule autonome	21
2.3.1	Architecture matérielle et logicielle	21
2.3.1.1	Capteurs	22
2.3.1.2	Systèmes d'aide à la conduite	23

2.3.1.3	Communication sans fil	24
2.3.2	Modélisation de la conduite	25
2.4	Modélisation et simulation du système	27
2.4.1	Ingénierie dirigée par les modèles	28
2.4.2	Conception orientée composant	28
2.4.3	Behavior-driven development	29
2.4.4	Systèmes multi-agent	29
2.4.5	Discussion : application à l'intersection autonome	30
2.5	Modèle d'architecture d'une intersection autonome	30
2.5.1	Diagramme de composants	31
2.6	Conclusion	33
II	Contributions à la conception de l'intersection autonome	35
3	Commande du véhicule	37
3.1	Contrôle longitudinal	38
3.1.1	Solutions classiques	39
3.1.1.1	Régulateur Proportionnel, Intégrateur, Dérivateur (PID)	39
3.1.1.2	Model Predictive Controller (MPC)	40
3.1.1.3	Le modèle de Gipps	40
3.1.1.4	Intelligent Driver Model (IDM) et Cooperative IDM	41
3.1.1.5	Autres modèles de suivi	42
3.1.1.6	Reaction Time - ACC (RT-ACC)	42
3.1.2	RT-ACC : validation expérimentale	43
3.1.2.1	Validation en simulation : comparaison CIDM et RT-ACC	43
3.1.2.2	Validation en conditions réelles	44
3.1.3	Conclusion (contrôle longitudinal)	46
3.2	Contrôle latéral	46
3.2.1	Solutions classiques	46
3.2.1.1	Pure poursuite	47
3.2.1.2	Méthode de Stanley	47
3.2.1.3	Commande linéaire quadratique	47
3.2.1.4	Méthode de Linderoth	48
3.2.2	Contribution à la commande latérale	48
3.2.3	Expérimentations et mesures	49

3.2.3.1	Simulations	49
3.2.3.2	Tests réels	52
3.2.4	Conclusion (contrôle latéral)	53
3.3	Adaptation de la vitesse par rapport à la trajectoire	54
3.3.1	Principe	54
3.3.2	Exemple d'application	55
3.3.3	Simulation et résultats	55
3.4	Conclusion	57
4	Intersection autonome et inter-blocage	59
4.1	Communication sans-fil et origines de l'inter-blocage	60
4.1.1	Modélisation de la communication sans-fil : latence et pertes	61
4.1.2	Inter-blocage d'une intersection autonome	62
4.2	Résolution de l'inter-blocage	64
4.2.1	Algorithme Collision-Deadlock-Free (CDF)	66
4.2.2	Absence de collision	67
4.2.3	Résolution de l'inter-blocage	68
4.3	Tests et résultats	70
4.4	Conclusion	71
5	Implémentation de l'intersection autonome	73
5.1	Méthodologie et outils logiciels	73
5.2	Architecture matérielle	75
5.2.1	Préparation des véhicules	75
5.2.1.1	Accélération	75
5.2.1.2	Freinage	76
5.2.1.3	Commande volant	77
5.2.1.4	Capteurs internes	77
5.2.1.5	Alimentation électrique et arrêt d'urgence	78
5.2.2	Capteurs externes et communication sans-fil	78
5.2.3	Ordinateur embarqué	79
5.2.4	Synthèse	80
5.3	Architecture logicielle	80
5.3.1	Capteurs et actionneurs	80
5.3.2	Communication	80
5.4	Définitions des conditions d'expérimentation	82

5.4.1	Objectifs de l'expérimentation	82
5.4.2	Scénario d'essai	82
5.4.3	Processus de test	83
5.5	Expérimentation et résultats	84
5.5.1	Suivi de trajectoire	84
5.5.2	Intersection autonome	85
5.6	Conclusion	86
III	Généralisation de l'intersection autonome	89
6	De l'intersection au changement de voie coopératif	91
6.1	Généralisation du concept d'intersection	92
6.1.1	Définitions	92
6.1.2	Intersection mobile	92
6.2	Adaptation au changement de voie	93
6.2.1	Modèle théorique du changement de voie	93
6.2.1.1	MOBIL	93
6.2.1.2	Modèle SUMO	94
6.2.1.3	Synthèse	94
6.2.2	Automated Cooperative Lane Change (ACLC)	94
6.2.2.1	Principe général	94
6.2.2.2	Définition de la zone mobile d'intersection	95
6.2.2.3	Critère d'acceptation	97
6.2.2.4	Synthèse	98
6.2.3	Simulation et résultats	98
6.3	Stratégie d'optimisation	100
6.3.1	Définition du problème	100
6.3.2	Stratégie d'optimisation mémétique	102
6.3.2.1	Algorithme	103
6.3.2.2	Simulation et résultats	106
6.4	Conclusion	109
IV	Conclusion et perspectives	113
7	Conclusion générale	115

<i>SOMMAIRE</i>	xi
7.1 Bilan	115
7.2 Perspectives	116
Acronyms	133
V Annexes	135
A Revue de presse du congrès ITS-WC 2015	137
A.1 Articles de presse évoquant l'expérimentation menée au congrès international des ITS à Bordeaux en 2015.	137
B Diagrammes de modules	139
B.1 Rôle des modules utilisés dans le contrôleur d'intersection et le contrôleur du véhicule	139
B.2 car-control	139
B.3 commsignia-v2x-client	140
B.4 communication-core	140
B.5 cooperative-autonomous-car	140
B.6 intersection-client	141
B.7 intersection-server	142
B.8 jssc	142
B.9 map-matching	142
B.10 pcan-usb-wrapper	143
B.11 positioning	143
B.12 pro-flex-500	144
B.13 renault-scenic-3	145
C Procédures de lancement et d'arrêt de la démonstration	147
D Étude de la commande latérale	151

I

CONTEXTE

INTRODUCTION GÉNÉRALE

L'amélioration du trafic routier est un défi majeur et actuel. Au cours du XX^{ème} siècle, les villes se sont étendues et le véhicule particulier s'est imposé comme le moyen de transport de référence. En conséquence le trafic routier a fortement augmenté, et de fait, la voirie s'est retrouvée partagée chaque jour par des millions de véhicules, engendrant la saturation du réseau, et provoquant accidents et congestions.

Pour pallier cette augmentation sans précédent des besoins en transport de personnes et de marchandises, des systèmes de régulation du trafic ont été mis en place. La signalisation routière et les feux de circulation se sont ainsi généralisés. Ceux-ci permettent d'organiser le partage de la route afin d'éviter les collisions et de fluidifier la circulation. En complément, l'évolution des modèles de l'ingénierie du trafic a permis d'optimiser l'organisation de l'infrastructure routière.

Néanmoins ces moyens de gestion du trafic apparaissent aujourd'hui limités. Le nombre de véhicules personnels augmente continuellement [Paquel et al., 2016], et ce, bien plus rapidement que la capacité d'accueil du réseau. Par conséquent, ce dernier sature. Cela a un impact direct sur l'économie et sur l'environnement. Le temps moyen passé dans les transports ne cesse d'augmenter, et la pollution provoquée par les véhicules thermiques (CO_2 , NO_x , particules fines) atteint un seuil critique [CITEPA/CORALIE, 2011].

Pour y remédier, les stratégies se multiplient : les villes développent leur réseau de transport en commun (bus, tramways), proposent des moyens de transports alternatifs (vélos en libre-service par exemple), promeuvent le covoiturage. Et lorsque c'est nécessaire, des restrictions de circulation sont appliquées (circulation alternée, interdiction de circulation des véhicules les plus anciens ou les plus polluants).

Cependant, depuis quelques années, le développement des systèmes de transport intelligent (STI) ouvre de nouvelles perspectives pour la régulation du trafic. Les véhicules s'automatisent, mais surtout deviennent capables de communiquer, entre eux (Vehicle-to-Vehicle (V2V)), ou bien avec l'infrastructure (Vehicle-to-Infrastructure (V2I)).

Parmi les nombreuses initiatives existantes, on peut citer GLOSA (indication de la vitesse optimale à l'approche d'un feu et conseil d'arrêt moteur), du projet COMPASS4D (<http://www.compass4d.eu>), qui vise à diminuer la consommation d'énergie de tout type de véhicule.

Cette émergence de la communication inter-véhiculaire donne naissance à de nouvelles approches de gestion du trafic, rassemblées sous le nom de « régulation coopérative ». La communication inter-véhiculaire y est utilisée afin d'introduire une forme de collaboration entre les véhicules, dont l'aboutissement est l'optimisation du trafic : moins de temps passé dans les transports, des économies de carburant, et une diminution de la pollution.

1.1/ CONTEXTE

L'intersection est le principal espace conflictuel de la circulation routière. Elle constitue un cas de partage spatial et temporel d'une ressource critique. C'est pourquoi la régulation du trafic s'est d'abord concentrée sur celle-ci, en définissant des règles de priorités. La signalisation de ces règles se fait via des panneaux ou des feux tricolores assurant la priorité d'un flux de véhicules sur un autre. Mais malgré les développements théoriques autour de ces dispositifs, la gestion par flux manque de flexibilité.

La régulation coopérative du trafic aux intersections se présente comme une solution à

ce problème. Permises par la généralisation prochaine des véhicules communicants, son principe fondateur est l'individualisation du droit de passage. En lieu et place des feux tricolores, qui régulent le trafic par flux, un ordinateur (ou Roadside Unit (RSU)) distribue le droit de passage à chaque véhicule.

Des travaux récents [Perronnet et al., 2013b], dont la présente thèse s'inscrit dans la continuité, ont montré que des stratégies coopératives s'avéraient plus performantes que des feux de signalisation pour gérer une intersection. En remplaçant la gestion par flux par une individualisation du droit de passage, les séquences de passages dans l'intersection peuvent être construites de manière quasi optimale. Les temps d'attente s'en trouvent alors réduits.

De plus, l'application des stratégies coopératives aux véhicules autonomes permet d'augmenter la vitesse des véhicules dans l'intersection en toute sécurité [Perronnet et al., 2013a], réduisant le temps passé dans la zone conflictuelle, améliorant encore la fluidité du trafic et réduisant les émissions de gaz polluants.

Néanmoins, indexer les commandes d'un véhicule (accélération, freinage, direction) sur des informations transmises par une communication sans fil impose plusieurs contraintes. En premier lieu, la commande du système physique continu (le véhicule) doit être capable d'intégrer l'aspect discret de la communication réseau qui sert à le piloter. En second lieu, le système doit être capable de se montrer robuste face aux problèmes de communication (pertes de messages, latences) et doit être en mesure de prévenir, sinon corriger, les situations d'inter-blocage.

1.2/ OBJECTIFS DE LA THÈSE

L'application de la régulation coopérative aux véhicules autonomes forme un système que nous appelons « intersection autonome ».

Les travaux existant sur l'intersection autonome et sur le trafic autonome sont à ce jour essentiellement théoriques. Des expériences conçues pour valider les protocoles existants ont été réalisées, mais elles restent rares. En particulier, avant que ne commencent les travaux de cette thèse, l'intersection autonome en conditions réelles avec au moins trois véhicules restait inédite.

La validation expérimentale étant essentielle à la vérification des modèles théoriques, il est nécessaire de concevoir un cadre permettant de modéliser, concevoir et prototyper l'intersection autonome, et plus globalement le trafic autonome par la généralisation des principes préalablement établis.

L'ingénierie dirigée par les modèles offre un cadre méthodologique pour atteindre cet objectif. Pour cela un modèle de l'intersection autonome doit être réalisé. À partir de celui-ci une démarche apparaît, allant de la conception de l'architecture pour l'intégration du véhicule autonome dans les stratégies de régulation coopérative existantes, jusqu'à la validation expérimentale par la réalisation d'un prototype réel d'intersection autonome.

Ce cadre posé, il devient ensuite possible d'étudier les problématiques de l'intersection autonome précitée, composant par composant, d'y apporter des solutions, et de les valider expérimentalement. En effet, des problèmes subsistent dans l'état de l'art sur le contrôle des véhicules et les algorithmes de régulation coopérative. Ce mémoire propose alors des solutions, requises en vue de la réalisation d'un prototype.

Par ailleurs, les cas de partage de ressources dans le trafic ne se limitant pas qu'aux seules intersections, le modèle préalablement posé peut être généralisé, notamment au cas du changement de voie. L'objectif est alors de profiter des mêmes bénéfices (en termes d'amélioration de la fluidité du trafic et de possibilités d'optimisations) mais dans un contexte différent. Les protocoles existants ne peuvent donc pas être appliqués tels quels et des travaux sont nécessaires pour, d'une part définir un protocole tenant compte des spécificités du changement de voie par rapport à l'intersection, d'autre part modéliser et simuler le système nouvellement défini pour en évaluer la pertinence. Enfin, de par leur nature l'optimisation d'une intersection et du dépassement diffèrent, il est donc intéressant d'étudier les possibilités d'optimisations du dépassement induites par l'utilisation de la communication inter-véhiculaire.

1.3/ PLAN DE LA THÈSE

Ainsi, dans le chapitre 2, cette thèse commence par étudier et proposer une architecture globale du système « intersection autonome » en s'appuyant sur les travaux récents réalisés dans ce domaine. Cette architecture se décompose alors en différents composants qu'il convient d'adapter pour les intégrer à l'ensemble.

En divisant ainsi le système en sous-modules et en adoptant une méthodologie de développement reposant sur la simulation logicielle avec Hardware-in-the-loop (HIL), les limites des modèles existants se manifestent. Afin d'y remédier, des modifications sont proposées, spécifiquement en ce qui concerne le contrôle du véhicule (chapitre 3) et la prévention de l'inter-blocage (chapitre 4).

Dans le chapitre 5, une synthèse de ces travaux est produite, se concrétisant par l'implémentation d'une intersection autonome réelle. Les conditions expérimentales de cette réalisation et les résultats obtenus y sont détaillés.

Finalement, les concepts de l'intersection autonome sont étendus au cas du changement de voie (dépassement, insertion) dans le chapitre 6, afin de généraliser l'utilisation de la coopération inter-véhiculaire à l'ensemble du trafic. Un cadre est alors construit sur ces bases pour optimiser le système résultant par une approche évolutionniste.

INTERSECTION AUTONOME ET COOPÉRATIVE

La nécessité de réguler le trafic est apparue très vite avec le développement de la circulation routière. Les feux tricolores ont été mis en place pour ordonner le droit de passage aux intersections, permettant de réduire les risques de collision et d'augmenter le flux de véhicules. La régulation du trafic par feux tricolores a ensuite bénéficié de développements théoriques permettant d'améliorer encore la fluidité du trafic. Toutefois, la régulation par feux tricolores présente aujourd'hui des limites qui empêchent d'optimiser plus encore le trafic. Les évolutions technologiques récentes, au travers des véhicules autonomes et communicants, permettent d'envisager de nouveaux moyens de réguler le trafic aux intersections s'affranchissant des limites des feux. En effet, la communication inter-véhiculaire permet d'individualiser le droit de passage des véhicules, permettant la création de séquences de passages plus complexes et optimisées.

Appliquée aux véhicules autonomes, cette individualisation du droit de passage donne naissance à l'**intersection autonome et coopérative**. L'intersection autonome et coopérative est un système de régulation du trafic aux intersections dans lequel des véhicules autonomes coopèrent pour éviter les collisions. Dans ce système, l'information échangée par les véhicules permet la gestion du droit de passage, mais également la synchronisation des vitesses entre véhicules. Grâce à cette synchronisation, les temps d'attentes peuvent être réduits et le trafic aux intersections peut encore être amélioré [Dresner et al., 2004], [Perronnet et al., 2013a].

L'intersection autonome est un système complexe et innovant, synthèse de productions théoriques et technologiques hétérogènes. Sa conception nécessite notamment d'intégrer les travaux existants sur la commande d'un véhicule automatisé avec la communication sans fil comme source d'information. Nous avons donc un système hybride, constitué d'un système physique continu (le véhicule) évoluant en utilisant des informations discrètes et présentant une latence.

Ainsi, dans ce chapitre nous commençons par présenter les principales définitions qui seront utilisées dans la suite de ce mémoire, tandis que nous proposons un état de l'art des travaux servant de fondation à la conception de l'intersection autonome. Dans un premier temps, nous nous focalisons sur les travaux relatifs à la gestion des intersections, et plus particulièrement à la gestion coopérative des intersections par utilisation de la communication intervéhiculaire. Puis, dans un second temps, nous portons notre attention sur l'automatisation des véhicules, en détaillant l'existant dans ce domaine.

Pour finir, nous introduisons brièvement quelques outils méthodologiques permettant de

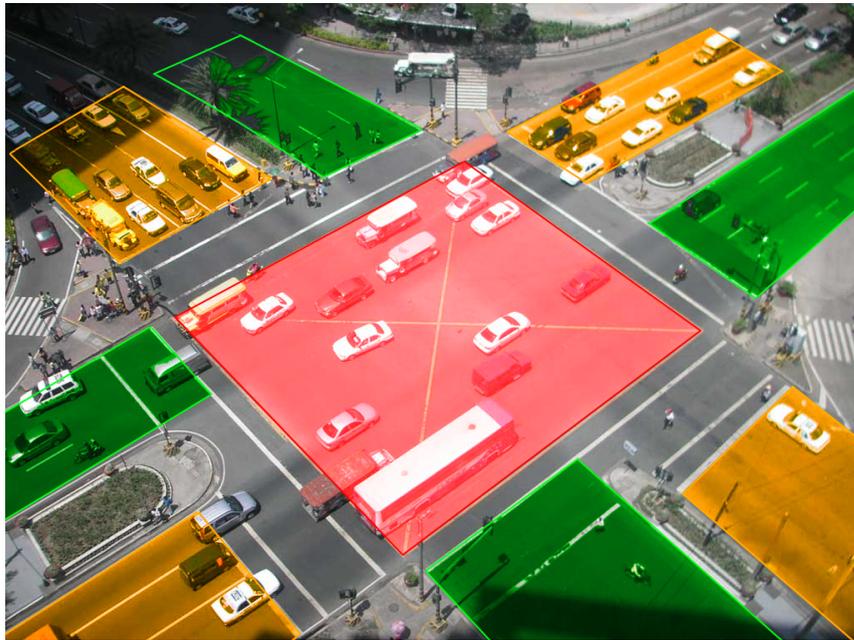


FIGURE 2.1 – Découpage d’une intersection en zones : en rouge la zone de conflit, en orange les zones de stockage, en vert les zones de sortie

modéliser et étudier l’intersection autonome. Puis nous concluons sur une synthèse présentant un modèle d’architecture du système « intersection autonome et coopérative », intégrant l’ensemble des différents composants nécessaires et présentés dans ce chapitre.

2.1/ DÉFINITIONS ET PRINCIPES GÉNÉRAUX

Cette section présente les principales définitions et les principes majeurs de l’ingénierie du trafic, et de la régulation classique du trafic aux intersections. De cette analyse du fonctionnement de la régulation du trafic, nous en faisons ressortir les limitations et leurs causes. La régulation coopérative du trafic est ensuite introduite, comme réponse à ces limites.

2.1.1/ INFRASTRUCTURE ROUTIÈRE ET INTERSECTION

Le **réseau routier** est l’espace pouvant être parcouru par les véhicules. Il peut être modélisé par un graphe dont les arêtes sont les routes ou voies de circulation (une route peut contenir plusieurs voies), et dont les nœuds sont les points de jonction de plusieurs voies de circulation.

Les intersections routières sont ces nœuds. Elles permettent aux véhicules de changer de direction, définissant ainsi un **mouvement** caractérisé par une voie d’origine et une voie de destination. Ainsi, c’est aux intersections que se décide l’affectation de l’infrastructure aux différents véhicules.

Une intersection peut être divisée en trois types de zones. La **zone de conflit** est le lieu où les mouvements peuvent se croiser, c'est la zone critique où les règles de priorité doivent être appliquées afin de donner la priorité à un mouvement sur un autre. Les **zones de stockage** sont les lieux où les véhicules attendent tant qu'ils n'ont pas l'autorisation (ou la priorité) pour s'engager dans la zone de conflit. Les **zones de sortie** sont la portion d'un mouvement hors de la zone de conflit et lui succédant. Le mouvement d'un véhicule traverse donc ces zones dans l'ordre suivant : zone de stockage, zone de conflit, zone de sortie. Dans de nombreux pays, le Code de la route interdit par défaut l'accès à la zone de conflit si la zone de sortie d'un véhicule n'est pas dégagée (afin de prévenir les situations d'inter-blocage).

La **signalisation routière**, qui comprend la signalisation verticale (panneaux, feux tricolores) et la signalisation horizontale (marquage au sol), permet de signaler aux conducteurs les règles de circulation, et notamment les règles de priorité aux intersections.

2.1.2/ MESURES DU TRAFIC

Le **temps inter-véhiculaire (TIV)** est l'intervalle en temps entre deux véhicules routiers consécutifs sur la même voie. Il est obtenu en mesurant le temps écoulé entre le passage de deux véhicules en un même point. Son équivalent spatial est la **distance inter-véhiculaire** qui est la distance séparant deux véhicules dans la même voie à un instant donné.

Le **débit** définit la répartition temporelle des véhicules, et correspond au nombre de véhicules étant passé par un point donné pour une période de temps donnée. Il est généralement exprimé en véhicules par heure. Son équivalent spatial (répartition des véhicules dans l'espace) est la **densité**, qui désigne à un instant donné le nombre de véhicules présents sur une section donnée.

Concernant les intersections, des mesures spécifiques existent pour en évaluer la performance. Le **temps d'arrêt** est le temps total passé par un véhicule à l'arrêt dans la zone de stockage. Le **retard** est la différence entre le temps que devrait mettre un véhicule pour rallier la zone de sortie depuis la zone de stockage en l'absence d'intersection (et en considérant sa vitesse constante) et le temps réellement mis sur ce trajet. Ainsi le retard inclut le temps d'arrêt, mais aussi le temps perdu dans les phases d'accélération et de décélération. Le **retard moyen par véhicule** permet d'évaluer la performance d'une intersection, et notamment de mesurer si une modification de l'intersection (infrastructure ou signalisation) a un impact positif ou négatif.

2.2/ RÉGULATION DU TRAFIC AUX INTERSECTIONS

Une même section de route (la ressource) est toujours naturellement traversée par les véhicules selon leur ordre d'arrivée, il n'est donc pas nécessaire de définir de règle de priorité. Aux intersections, cependant, deux véhicules peuvent requérir de traverser un espace (ou ressource) partagé simultanément, ce qui résulterait en un accident. Le rôle de la régulation du trafic est d'éviter ce genre de situation en organisant la séquence de passage des véhicules.

2.2.1/ APPROCHE CLASSIQUE

L'approche traditionnelle de la régulation du trafic aux intersections repose sur la priorisation d'un flux de véhicules sur un autre. Ainsi, un panneau « stop » ou « cédez le passage » indique aux conducteurs qu'ils n'ont pas la priorité à une intersection, et qu'ils ne pourront la traverser qu'à condition de ne pas faire obstacle à un véhicule venant d'une voie prioritaire. En l'absence de signalisation explicite, une règle de priorité « par défaut » est appliquée : en France, c'est la priorité à droite, dans d'autres pays (États-Unis notamment), c'est la règle « premier arrivé, premier sorti ».

Cependant, ces approches ne permettent pas de partager équitablement l'intersection entre les véhicules provenant des différentes voies (lors de la priorisation d'un flux) ou génèrent trop de phases de décélération/accélération (lors de l'application de la logique premier arrivé, premier servi), notamment en cas de fort trafic, et peuvent notamment conduire à la congestion des voies non prioritaires. Pour y remédier, la régulation des intersections par feux tricolores est généralement employée en cas de trafic important.

2.2.1.1/ FEUX TRICOLORES À CYCLE FIXE

Dans un carrefour à feux tricolores, le feu de signalisation, placé avant ou après l'intersection, permet d'indiquer au conducteur s'il a le droit de traverser l'intersection (feu vert) ou s'il doit s'arrêter (feu rouge). Le feu jaune ou orange sert à indiquer le passage du vert au rouge, et parfois le passage du rouge au vert. Ainsi, les feux tricolores distribuent successivement le droit de passage aux différentes voies. À un instant donné, le droit de passage est donné à certains mouvements pour une période de temps déterminée, cela constitue une **phase**. La succession des différentes phases se répète de manière cyclique, ainsi on parle de **cycle** pour décrire l'ensemble de ces phases.

La définition de ces phases impacte directement l'efficacité de l'intersection, et le calcul de ces durées a fait l'objet de nombreux travaux de recherche [Council, 2000]. La formule de Webster [Webster, 1958] permet de calculer le cycle optimal pour minimiser les temps d'attente des véhicules, et a été largement utilisée pour les feux à cycle fixe, c'est-à-dire les feux ayant un cycle fixe avec des durées de phases constantes.

Les feux à cycle fixe, cependant, sont limités par l'aspect statique des phases. Bien que différentes programmations puissent exister en fonction de l'horaire, notamment pour gérer les heures de pointe, en cas d'afflux soudain de véhicules en provenance d'une voie tandis que les autres voies sont dégagées, la priorité ne sera pas donnée à la voie congestionnée. Dans ce cas, les feux à cycle fixe ne permettent pas d'évacuer la voie congestionnée, alors que lui donner le droit de passage permettrait de l'évacuer sans pénaliser d'autres véhicules. Pour améliorer cet aspect, les feux adaptatifs ont été introduits : en fonction de l'état du trafic, les différentes phases peuvent être adaptées, soit par la modification de la durée d'une phase, soit par l'insertion d'une phase supplémentaire.

2.2.1.2/ FEUX TRICOLORES ADAPTATIFS

Les feux tricolores adaptatifs permettent l'ajustement dynamique des phases et de leur durée en fonction de l'état du trafic. Cela nécessite un dispositif de mesure du trafic, et un régulateur en charge de redéfinir les phases en fonction de la mesure.

De nombreux capteurs peuvent être utilisés pour mesurer l'état du trafic, les plus notables étant les boucles d'induction magnétique et les caméras vidéos. Le rôle de ces capteurs est de mesurer la présence ou le passage d'un véhicule en un point donné, et par extension le débit d'une voie. Ces informations sont ensuite transmises au régulateur.

Plusieurs stratégies de régulation peuvent être adoptées par ce dernier :

- Escamotage : lorsqu'au cours d'un cycle, aucun véhicule n'est détecté dans une voie, la phase donnant le droit de passage à cette voie est supprimée pour ce cycle.
- Correction du temps de cycle : la transition du droit de passage d'une voie à l'autre nécessite un temps de rouge intégral, ce qui ralentit le trafic. En cas de débit fort il est intéressant de limiter ces transitions, donc d'augmenter la durée des phases, et inversement en cas de débit faible.
- Véhicule actionneur : dans le cas d'une intersection avec des voies présentant un faible débit et des voies présentant un débit plus important, le principe du véhicule actionneur permet de laisser la voie libre en continu aux voies ayant un fort débit. Le droit de passage est donné aux autres voies uniquement quand un véhicule y est détecté.

Ces différentes stratégies permettent d'améliorer les performances des intersections à feux, mais présentent néanmoins certains inconvénients. Concernant le paramétrage de ces intersections, une intervention humaine est nécessaire pour définir la stratégie à appliquer pour un contexte donné, et tout changement notable du trafic peut nécessiter la redéfinition de la stratégie. Concernant la performance, ces stratégies gardent un aspect statique, et ne permettent pas la composition de mouvements et de séquences complexes : le système de mesure ne donne qu'une indication de l'occupation des voies (et non le nombre exact de véhicules), le système de signalisation donne toujours le droit de passage à un flux (et non à un véhicule).

2.2.1.3/ DISCUSSION ET PERSPECTIVES

L'utilisation des feux tricolores s'est rapidement imposée comme la solution de référence pour réguler le trafic aux intersections. Afin de faire face à un trafic toujours croissant, ces derniers ont bénéficié d'améliorations considérables. Toutefois cette approche présente une limite intrinsèque : le trafic est divisé en flux, et la mesure du trafic ou le droit de passage concernent des flux de véhicules.

Dans certaines situations, considérer chaque véhicule individuellement permettrait d'améliorer plus encore la performance de l'intersection, mais la nature même de la signalisation par feux ne le permet pas. Par exemple, dans la figure 2.2, la solution optimale consiste à donner le droit de passage aux véhicules gris puis au rouge (pour minimiser le temps d'attente moyen), mais définir cette stratégie requiert de connaître exactement les véhicules présents et leurs trajectoires.

Pour pallier ces limitations, des approches innovantes de régulation sans feux sont élaborées. Ces stratégies reposent sur la communication inter-véhiculaire et la coopération entre véhicules, et sont justement nommées « approches coopératives ».

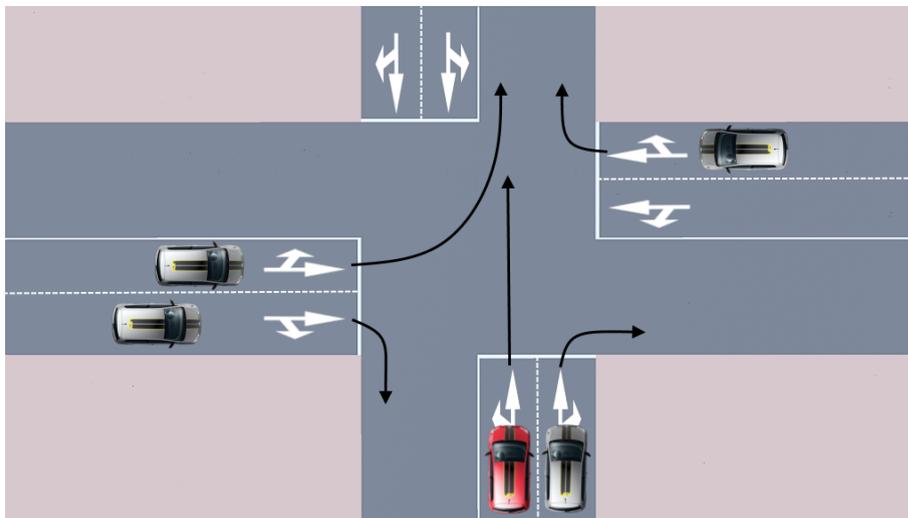


FIGURE 2.2 – Intersection complexe : intuitivement la meilleure séquence consiste à donner le droit de passage aux véhicules gris puis au rouge

2.2.2/ APPROCHE COOPÉRATIVE

Depuis le début des années 2000, de nouvelles approches sont proposées pour réguler le trafic aux intersections. Ces approches sont permises par le développement de la technologie, et plus particulièrement le développement de la communication véhiculaire et des systèmes d'aide à la conduite. Grâce à des unités de communication sans-fil, les véhicules peuvent signaler leur présence et leur trajectoire au niveau de l'intersection, permettant de distribuer un droit de passage individualisé et réactualisé à chaque changement d'état de l'intersection. En individualisant le droit de passage, ces approches permettent de créer des combinaisons plus complexes de mouvements autorisés simultanément aux intersections, et permettent l'optimisation de la vitesse des véhicules dans l'intersection. Elles permettent ainsi de répondre aux limitations des feux, et la qualité du trafic s'en trouve améliorée [Fok et al., 2012], [Lee et al., 2012].

La régulation coopérative est un domaine émergent, et différentes solutions existent dans la littérature pour l'implémentation de la régulation coopérative, avec des architectures matérielles différentes (centralisée, décentralisée), des protocoles de négociation différents (par synchronisation de vitesse, par réservation, par séquence, etc.), et des politiques de régulation différentes (premier arrivé, premier servi, par stratégie d'optimisation, ou par « paquets »).

Concernant l'architecture matérielle, les implémentations diffèrent selon que la communication est centralisée, décentralisée ou semi centralisée et selon que les véhicules sont conduits ou autonomes. Le choix d'une architecture matérielle au profit d'une autre conditionne les possibilités en termes de protocole de négociation (types de messages, données transmises, séquence des messages), et le choix d'un protocole conditionne les possibilités en termes de politiques de régulation (par exemple premier arrivé, premier servi, ou par groupe).

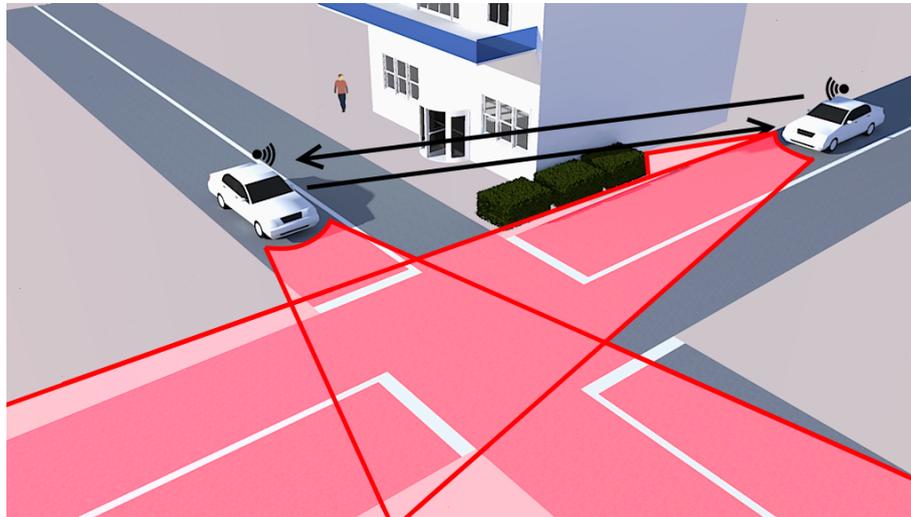


FIGURE 2.3 – Représentation schématique de la communication V2V : les véhicules communiquent directement entre eux, le champ de vision des capteurs frontaux des véhicules est affiché en rouge

2.2.2.1/ ARCHITECTURE DÉCENTRALISÉE

L'approche décentralisée repose sur la communication directe entre véhicules (V2V). A l'approche d'une intersection, les véhicules en conflit négocient directement entre eux, sans intermédiaire, pour définir leur ordre de passage dans l'intersection. Cette approche a été initialement proposée par [Naumann et al., 1997]. De nombreux travaux ont été réalisés par la suite pour définir des protocoles et politiques de régulation autour de l'architecture décentralisée [Makarem et al., 2011], [Adams et al., 2015], [Medina et al., 2017].

Le principal avantage de l'architecture décentralisée est le faible coût matériel de la solution : il n'est nécessaire d'équiper que les véhicules de dispositifs de communication. De plus, en supposant que les véhicules disposent d'une connaissance parfaite de leur environnement et que la communication sans-fil est infaillible, ces approches fonctionnent et permettent de prévenir les risques de collision aux intersections.

En conditions réelles, la communication sans-fil peut échouer (signal faible, défaillance matérielle) et un véhicule peut être incapable de communiquer avec les autres véhicules, et donc de recevoir l'information qu'il y a d'autres véhicules et qu'il n'est pas prioritaire à l'intersection. Ce qui pourrait résulter en une collision. On peut envisager de rendre la main au conducteur dans cette situation, mais ce n'est pas compatible avec l'idée d'un véhicule complètement autonome. On peut aussi équiper le véhicule de capteurs permettant de détecter les obstacles, mais les véhicules conflictuels ne se trouvent pas dans la même voie de circulation et des obstacles (bâtiments, végétation) peuvent empêcher les véhicules de se percevoir, les obligeant alors à s'arrêter à l'intersection. Des capteurs pourraient être disposés à l'intersection pour contrôler la présence des véhicules, mais on perd alors l'intérêt de la négociation décentralisée, et on revient sur une solution proche des feux tricolores adaptatifs. L'architecture centralisée propose d'apporter une réponse à ces problèmes.

2.2.2.2/ ARCHITECTURE CENTRALISÉE

L'approche centralisée est pensée pour répondre aux limites de l'architecture décentralisée : les véhicules ne communiquent plus directement entre eux, mais par le biais d'un contrôleur d'intersection (ou serveur) dont le rôle est de distribuer le droit de passage aux différents véhicules qui en font la requête (V2I). Cela permet d'introduire directement le principe d'arrêt par défaut : un véhicule défaillant incapable de négocier son droit de passage ne peut pas l'obtenir, il doit par défaut s'arrêter à l'intersection. Au contraire de l'approche décentralisée, l'absence de communication ne signifie plus l'absence d'autres véhicules (et donc un droit de passage tacite). À noter, des protocoles centralisés reposant sur le droit de passage par défaut sont envisageables, mais présenteraient un risque de collision en cas de défaut de communication.

Enfin, de même que l'architecture décentralisée, l'approche centralisée sert de support à différents protocoles et politiques de négociation.

Protocole par réservation Le protocole par réservation a été présenté pour la première fois dans [Dresner et al., 2004]. Il repose sur le principe de la réservation d'un espace-temps dans l'intersection : le véhicule approche de la zone de conflit, estime à quel moment et où il sera présent dans l'intersection, puis réserve cet espace-temps auprès du contrôleur d'intersection. Le contrôleur d'intersection peut alors accepter ou refuser cette réservation, en fonction des autres réservations déjà en cours : si dans la requête un espace donné est déjà occupé pour un temps donné, alors la réservation sera refusée. Une fois la réservation acceptée, le véhicule est tenu de la respecter (il ne peut pas être en retard ou en avance).

L'implémentation de ce principe implique quelques considérations pratiques :

- La requête de passage peut être refusée : en ce cas le véhicule doit réémettre des requêtes jusqu'à ce qu'une soit acceptée
- En cas de changement (par exemple si un véhicule est incapable de respecter sa requête initiale), le véhicule doit réémettre une nouvelle requête (avec un risque de refus)
- En cas de retard de communication, un véhicule peut tarder à obtenir une réservation et être contraint de s'arrêter

Ces difficultés ont pu être mises en évidence lors de tests réels [Quinlan et al., 2010], [Fok et al., 2012]. Le nombre important d'échanges de messages peut alors entraîner une saturation du serveur et de la communication réseau aggravant plus encore ces difficultés.

Intersection-based Cooperative Adaptive Cruise Control Dans [Zohdy et al., 2012] et [Zohdy et al., 2016], une implémentation de la négociation centralisée est proposée, nommée Intersection-based Cooperative Adaptive Cruise Control (iCACC). Les véhicules envoient une requête de passage au contrôleur d'intersection, qui en retour envoie une consigne de vitesse aux véhicules de manière à optimiser l'évacuation de l'intersection. Ce principe sert notamment de base à [Shahriari et al., 2016].

Ce protocole suppose que l'application de la consigne de vitesse sera exécutée parfaitement et sans délai. Cependant, ces conditions sont difficiles à respecter en réalité, les

véhicules n'étant pas suffisamment précis dans l'application de leur commande, et les délais de communication étant difficiles à prévoir. De plus, le contrôleur pilote indirectement les accélérations et les décélérations des véhicules : si une décélération est requise (car, par exemple, un véhicule conflictuel n'a pas pu ajuster sa vitesse comme prévu), mais que le message n'est pas transmis à temps, un risque de collision existe.

Protocole par séquence de passage Les travaux les plus récents se basent sur la construction d'une séquence de passage (ou attribution d'un niveau de priorité) au niveau de l'intersection ou au niveau de chacune des zones de conflit à l'intérieur de l'intersection [Wu et al., 2012], [Perronnet et al., 2012], [Gregoire et al., 2014], [Qian et al., 2014], [Tachet et al., 2016]. Dans ces protocoles, les véhicules émettent une requête auprès du contrôleur d'intersection pour obtenir le droit de passage, le contrôleur d'intersection compose alors une séquence de passage des véhicules : les véhicules doivent franchir l'intersection dans l'ordre défini par la séquence.

Par exemple, dans le protocole Cooperative Vehicle-Actuator System (CVAS) défini dans [Perronnet et al., 2012], les véhicules envoient une requête de passage contenant leur mouvement auprès du contrôleur d'intersection. Ce dernier construit alors une séquence de passage en accord avec une politique de régulation (par exemple premier arrivé, premier servi), et transmet la liste des véhicules autorisés. Quand un véhicule quitte l'intersection, il est supprimé de la séquence. Un véhicule est autorisé à traverser l'intersection dès qu'il n'y a plus de véhicules le précédant et ayant un mouvement conflictuel. L'avantage de ce protocole par rapport aux précédents (iCACC et réservation) est l'absence de contraintes difficiles à appliquer (respect d'un temps précis, ou d'une vitesse précise), puisque seul le droit de passage est communiqué aux véhicules. Ainsi, ce protocole a pu être expérimenté avec des conducteurs humains bénéficiant d'un affichage à bord (voir fig. 2.4). Dans [Perronnet et al., 2013b] il est montré que ce protocole est robuste face aux problèmes de latence de la communication, et approche la solution optimale quand associé à une politique Distributed Clearing Policy (DCP) (voir section 2.2.3.3).



FIGURE 2.4 – Expérimentation du protocole CVAS en conditions réelles : un smartphone communique en WiFi avec le contrôleur d'intersection pour obtenir le droit de passage individualisé ; l'affichage rouge indique au conducteur qu'il n'est pas autorisé à franchir l'intersection

Le protocole Transparent Intersection Manager (TIM) défini dans [Perronnet et al., 2013a] est une extension de CVAS au cas des véhicules autonomes (voir figure 2.5. Les véhicules transmettent leur mouvement, leur position et leur vitesse au contrôleur, qui en retour construit et transmet la séquence de passage des véhicules. Chaque véhicule synchronise sa vitesse avec le véhicule conflictuel qui le précède dans la séquence. S'il y a une latence dans la transmission de la position émise par un véhicule, cela n'engendre pas de risque de collision, car chaque véhicule se base sur la dernière position connue des autres véhicules.

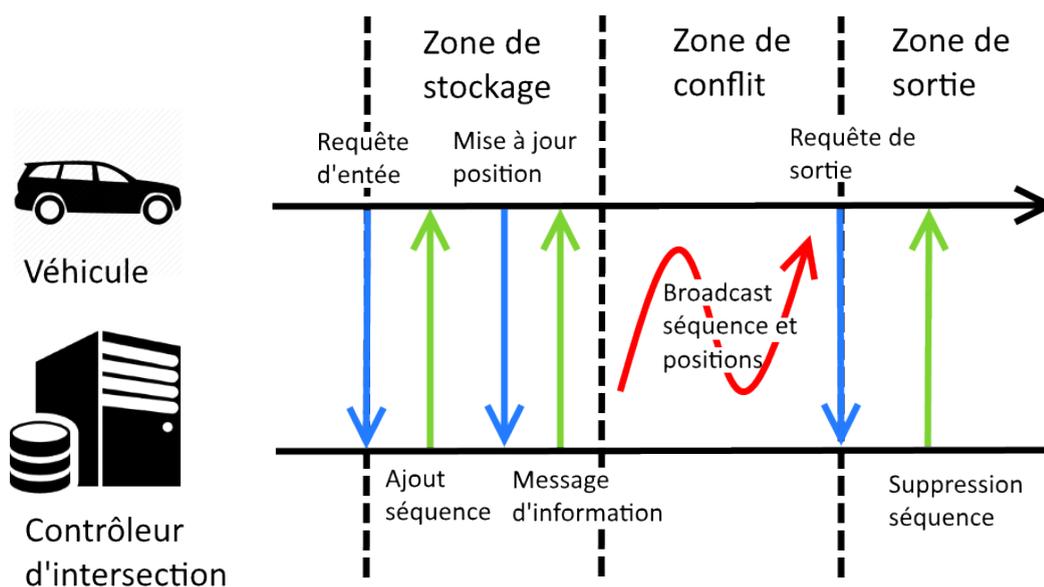


FIGURE 2.5 – Echanges de messages entre le contrôleur d'intersection et le véhicule dans le cadre du protocole TIM

La figure 2.7 illustre le fonctionnement de ce protocole : en entrant dans la zone de stockage un véhicule envoie une requête d'entrée et est ajouté à la séquence de passage (stockée au niveau du contrôleur et retransmise aux véhicules), en sortant une requête de sortie indique au contrôleur de supprimer le véhicule de la séquence. Ici, 1, 2 et 3 ont le droit de passage. 4 doit attendre après 3 pour pouvoir passer, car 3 est avant 4 dans la séquence et que les mouvements de 3 et 4 sont conflictuels.

La figure 2.6 illustre la façon dont une séquence de passage se construit en fonction des messages reçus. Trois types de messages coexistent : « entrée », « actualisation », « sortie ». Ces messages contiennent également l'identifiant, la position, la vitesse et la trajectoire des véhicules. On observe notamment sur cet exemple que les véhicules sont insérés de façon ordonnée dans la séquence de passage lors des requêtes « entrée », que les données sont mises à jour lors des requêtes « actualisation » et que les véhicules sont retirés de la séquence lors des requêtes « sortie » (permettant ici au véhicule B d'acquiescer le droit de passage après la sortie du véhicule A).

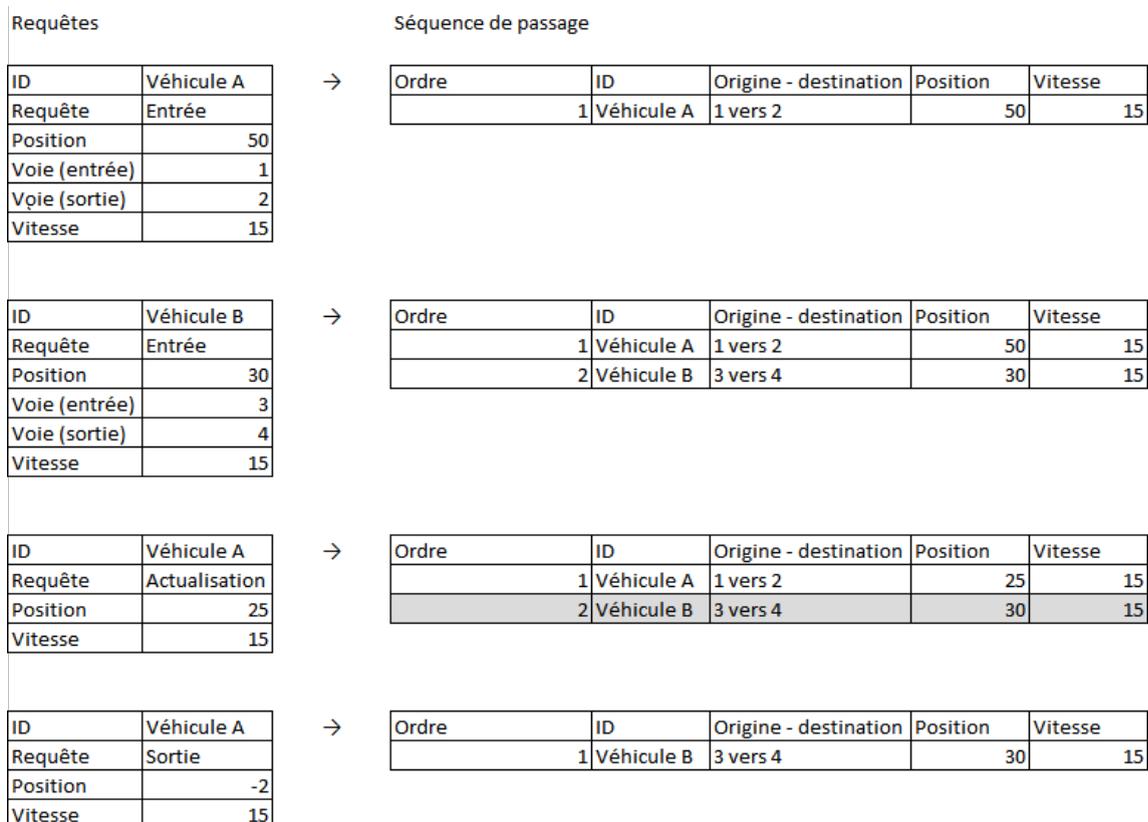


FIGURE 2.6 – Construction de la séquence de passage en fonction des messages reçus - le mouvement 1 vers 2 est supposé en conflit avec le mouvement 3 vers 4

2.2.2.3/ ARCHITECTURE SEMI-CENTRALISÉE

Les architectures semi-centralisées fonctionnent sur les mêmes principes que les architectures centralisées, à savoir qu'un contrôleur d'intersection doit assurer la distribution du droit de passage aux véhicules. La différence est que les véhicules n'ont pas nécessairement à communiquer directement avec le contrôleur d'intersection, les autres véhicules pouvant faire office de relai de communication lors de la requête du droit de passage (Vehicular Ad-Hoc Network (VANET)), et peuvent utiliser les informations communiquées directement par les autres véhicules pour adapter leurs vitesses (en V2V).

L'architecture semi-centralisée peut donc servir de support aux mêmes protocoles que l'architecture centralisée, mais possède des avantages en termes de communication : une portée étendue du contrôleur d'intersection par le principe du relai, et une diminution de la charge au niveau du contrôleur puisque certaines informations sont diffusées directement entre véhicules.

2.2.2.4/ DISCUSSION

À partir de cette analyse des différentes solutions de régulation coopérative des intersections, nous pouvons extraire des principes généraux qui doivent être respectés pour qu'un protocole soit applicable en conditions réelles. Ces principes répondent à deux

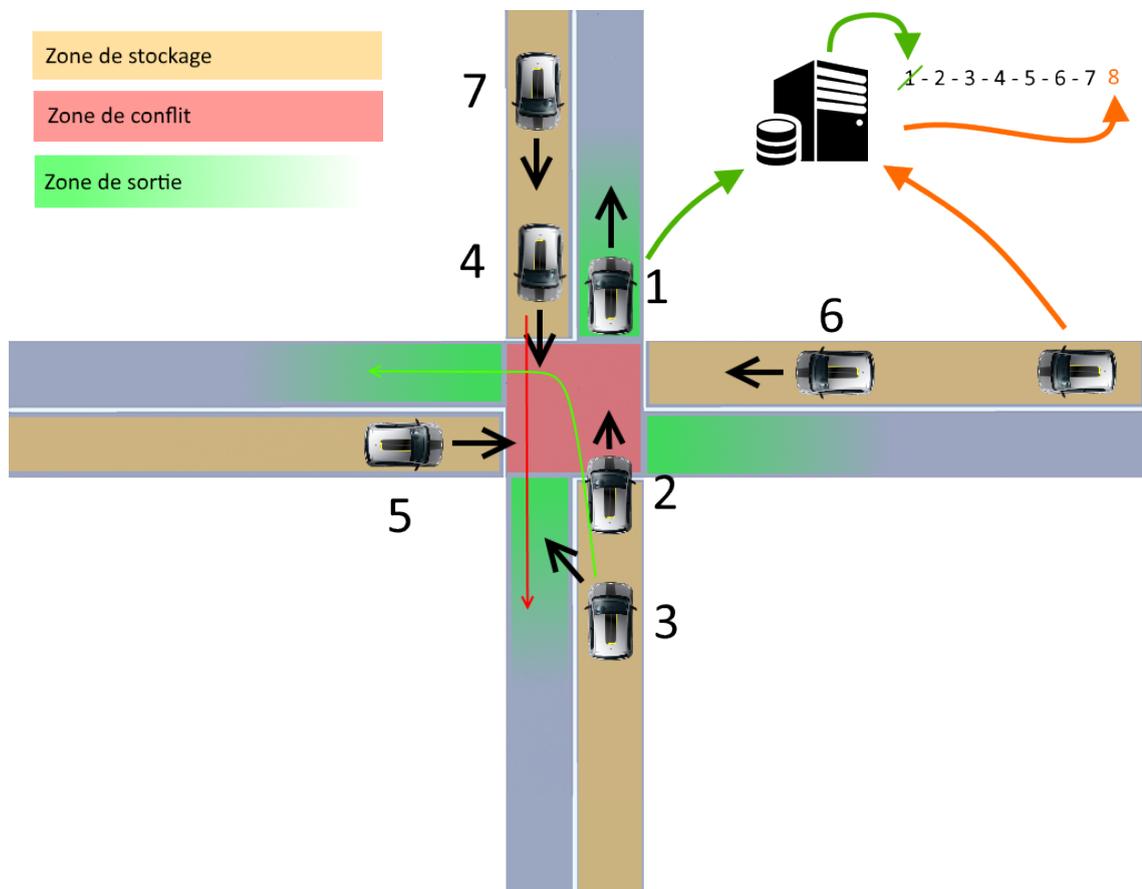


FIGURE 2.7 – Illustration du fonctionnement du protocole TIM

contraintes :

1. la communication inter-véhiculaire peut échouer ou présenter une latence
2. une consigne de commande (accélération, freinage, direction) peut ne pas pouvoir être respectée (défaillance matérielle, obstacle imprévu) ou être appliquée de façon imprécise

Du fait de la première contrainte, il doit y avoir **arrêt par défaut** : l'absence de communication ne peut pas donner un droit de passage tacite. A cause de cette même contrainte, le **droit de passage doit être définitif** : il est possible que le message informant un véhicule que son droit de passage lui a été retiré ne soit pas transmis, ou soit transmis trop tard ; de la même façon, un message ne peut pas être destiné à réduire la vitesse d'un véhicule. Enfin, à cause la deuxième contrainte, il ne peut pas être fait de supposition sur la position ou la vitesse courante d'un véhicule, ainsi le **droit de passage pour une zone conflictuelle est donné à un véhicule si, et seulement si, tout véhicule qui devait occuper cette zone avant a explicitement libéré la zone.**

Afin de respecter ces règles, les architectures centralisée ou semi-centralisée sont préférables, ainsi que les protocoles par séquence.

2.2.3/ POLITIQUES DE RÉGULATION COOPÉRATIVE

Indépendamment de l'architecture utilisée pour concevoir un système de régulation de trafic aux intersections coopératif, plusieurs politiques de régulation peuvent être appliquées.

L'objectif d'une politique de régulation est d'ordonner la distribution du droit de passage à l'intersection. L'utilisation de la régulation coopérative et l'individualisation du droit de passage ouvrent de nouvelles possibilités en termes de politiques. Il devient notamment possible de chercher à « optimiser » l'intersection, c'est-à-dire à distribuer le droit de passage de façon à minimiser ou maximiser certaines mesures (par exemple, le retard moyen ou le temps d'évacuation).

La littérature présente plusieurs politiques de régulation coopérative possible.

2.2.3.1/ PREMIER ARRIVÉ, PREMIER SERVI

Cette politique est l'une des plus utilisées, notamment dans les travaux s'appuyant sur le protocole de réservation [Dresner et al., 2004], [Dresner et al., 2008]. Les véhicules obtiennent le droit de passage dans l'ordre dans lequel ils contactent le contrôleur d'intersection, peu importe les conflits entre les mouvements souhaités. Cette politique est efficace en cas de trafic faible à modéré [Perronnet et al., 2013b] mais d'autres politiques prennent l'avantage en cas de trafic important. De plus, elle ne permet pas d'optimiser l'efficacité de l'intersection.

2.2.3.2/ PROGRAMMATION DYNAMIQUE, META-HEURISTIQUES

En choisissant précautionneusement l'ordre de passage des véhicules dans l'intersection, il est possible d'essayer de limiter les accélérations, les freinages, les temps d'arrêt et de maximiser l'évacuation de l'intersection. Pour cela le contrôleur d'intersection compose des groupes de véhicules pouvant traverser l'intersection simultanément. Ce problème peut être vu comme un problème combinatoire classique pouvant être représenté grâce à la théorie des graphes. Ainsi, il est possible de résoudre ce problème via des méthodes exactes, comme la programmation dynamique [Wu et al., 2009], ou des méthodes approchées comme les algorithmes de colonies de fourmis [Wu et al., 2012], les algorithmes évolutionnistes, la recherche taboue ou le recuit simulé [Shahriari et al., 2016].

Bien que pouvant fournir une solution exacte au problème de l'évacuation d'une intersection, ces solutions sont difficilement applicables en temps réel : d'une part elles nécessitent un temps de calcul important, d'autre part elles s'appuient sur une connaissance parfaite des arrivées de véhicules à l'intersection. Néanmoins ces solutions permettent d'obtenir, en simulation, la solution optimale et fournissent une référence pour l'évaluation d'autres politiques.

2.2.3.3/ POLITIQUE D'ÉVACUATION DISTRIBUÉE

Pour pallier les limitations des stratégies d'optimisation, la politique d'évacuation distribuée [Perronnet et al., 2013b] a été développée. Il s'agit d'une heuristique consistant à

former des groupes de véhicules situés sur une même voie et proches temporellement ou spatialement. Ainsi, quand un véhicule émet une requête de passage et s'il est proche temporellement (temps intervéhiculaire inférieur à un seuil prédéfini) ou spatialement (distance intervéhiculaire inférieure à un seuil prédéfini) d'un véhicule ayant le droit de passage et se trouvant dans la même voie, alors le nouveau véhicule obtient le droit de passage.

Cette stratégie permet de former des convois de véhicules et a montré des performances s'approchant des solutions optimales en simulation, notamment en cas de trafic fort. La figure 2.8 illustre le principe de la politique d'évacuation distribuée : le véhicule gris est accepté avant le rouge, car il est proche du véhicule vert déjà autorisé. Une politique premier arrivé, premier servi aurait fait passer le véhicule rouge avant le gris, imposant à celui-ci de s'arrêter aussi et augmentant le retard moyen.

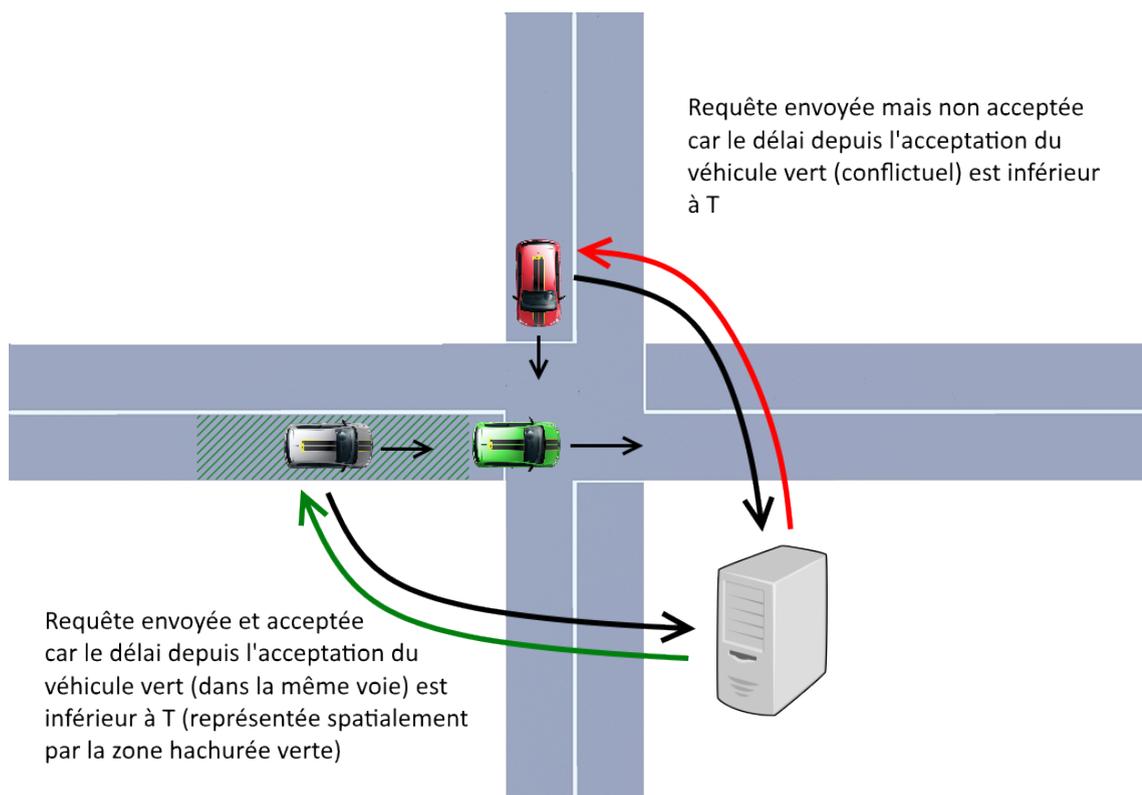


FIGURE 2.8 – Illustration de la politique d'évacuation distribuée : le véhicule gris étant proche du véhicule autorisé (vert), il est autorisé avant le véhicule rouge

2.2.4/ DISCUSSION

Nous venons de passer en revue l'évolution des systèmes de gestion du trafic aux intersections au cours du XX^{ème} siècle, et nous en avons montré les limites. Cependant, l'émergence des technologies de communication inter-véhiculaire ouvre de nouveaux horizons quant à la gestion du trafic, et ainsi apparaît la régulation coopérative des intersections.

La gestion coopérative des intersections exploite un certain nombre de principes (individualisation du droit de passage, connaissance du mouvement des véhicules) afin de

répondre aux limitations de la régulation par feux tricolores, et d'optimiser le trafic aux intersections.

Le concept d'intersection autonome et coopérative prolonge la régulation coopérative des intersections en usant des particularités des véhicules autonomes pour améliorer plus encore la qualité du trafic. Il devient notamment possible d'ajuster automatiquement la vitesse des véhicules pour diminuer le temps d'occupation de la zone de conflit, limiter les phases d'accélération et de décélération et ainsi réduire la consommation d'énergie.

Pour ce faire, il faut s'attacher à la conception de véhicules pilotables en accélération, en freinage et en direction, et intégrer le fait que, pour l'intersection autonome et coopérative, la communication Vehicle-to-Anything (V2X) devient l'une des principales sources d'information pour la commande du véhicule.

2.3/ VÉHICULE AUTONOME

Un véhicule autonome est un véhicule capable d'évoluer dans le trafic sans la moindre intervention humaine. Plus précisément, la Society of Automotive Engineers (SAE) définit dans son standard SAE J3016 six niveaux d'automatisation [Committee, 2014] :

- Niveau 0 - Aucune automatisation : le conducteur contrôle entièrement son véhicule
- Niveau 1 - Assistance à la conduite : l'essentiel des fonctions est toujours piloté par le conducteur, mais certaines fonctions spécifiques concernant le contrôle de l'accélération ou de la direction peuvent être réalisées automatiquement par la voiture (freinage automatique d'urgence par exemple)
- Niveau 2 - Automatisation partielle : les fonctions d'accélération et de direction sont entièrement pilotées par le système, mais le conducteur est toujours en charge des décisions, de plus il doit reprendre le contrôle du véhicule en cas de nécessité
- Niveau 3 - Automatisation conditionnelle : un système de pilotage automatique contrôle entièrement le véhicule, mais le pilote doit pouvoir reprendre la main à tout moment en cas de besoin
- Niveau 4 - Automatisation avancée : le système de pilotage automatique est capable de gérer entièrement le véhicule, même si le pilote ne réagit pas correctement lorsque le système lui demande d'intervenir
- Niveau 5 - Automatisation complète : le système de pilotage automatique est capable de remplir sans condition toutes les fonctions d'un pilote humain (le véhicule peut alors être éventuellement inoccupé)

Un tel véhicule doit répondre à certaines caractéristiques, notamment être équipé de mécanismes pilotables de contrôle de l'accélération, du freinage et de la direction, de capteurs, et de logiciels permettant de faire le lien entre l'information provenant des capteurs et le contrôle du véhicule.

2.3.1/ ARCHITECTURE MATÉRIELLE ET LOGICIELLE

Afin d'y parvenir, l'architecture du véhicule doit être adaptée pour que celui-ci puisse percevoir son environnement et réagir à ces perceptions. Pour cela des capteurs doivent équiper le véhicule, afin de fournir les informations requises aux systèmes d'aides à

la conduite. Les travaux sur la conception de véhicules autonomes sont anciens et nombreux [Payton, 1986], [Chen et al., 2004], mais l'évolution récente des équipements d'aide à la conduite, et la commercialisation prochaine de ces véhicules (niveau 4 en 2018 d'après Tesla) permet de définir précisément les éléments nécessaires [Guizzo, 2011].

En sus, un véhicule autonome et communicant doit disposer d'un système de communication inter-véhiculaire lui permettant d'envoyer et de recevoir des informations de la part des autres véhicules, et de l'infrastructure.

2.3.1.1/ CAPTEURS

Les capteurs sont les instruments permettant à un véhicule de mesurer des grandeurs physiques qui lui sont propres. En robotique mobile, les capteurs se divisent en deux catégories : les capteurs proprioceptifs qui effectuent leurs mesures par rapport à ce qu'ils perçoivent localement du déplacement du robot, et les capteurs extéroceptifs qui se basent sur des mesures relatives à l'environnement global.

Capteurs proprioceptifs Les premiers capteurs à avoir équipé les véhicules sont les **odomètres** et **tachymètres**, dont les fonctions sont respectivement de mesurer la distance parcourue et la vitesse instantanée. Ceux-ci fonctionnent grâce à la mesure du déplacement des roues (en supposant qu'il n'y ait ni patinage ni dérapage). Par l'adjonction d'un **capteur d'angle volant** permettant de connaître l'angle des roues, il est également possible d'estimer le lacet du véhicule (angle de rotation autour de l'axe vertical, aussi connu sous l'appellation anglaise « yaw »).

En complément les véhicules modernes sont équipés d'un **accéléromètre** et d'un **gyromètre**, voire d'une **centrale inertielle** permettant de mesurer les accélérations latérales et longitudinales, et la vitesse de lacet du véhicule.

Systèmes de positionnement global Les systèmes de positionnement global permettent à un véhicule de connaître sa position par rapport à un repère global de l'environnement. Le système de positionnement par satellites appelé **Global Navigation Satellite System (GNSS)** est constitué de plusieurs satellites émettant un signal codé qui peut être exploité par un récepteur GNSS afin de se géolocaliser.

Le premier GNSS est le **Global Positioning System (GPS)** développé par les États-Unis à partir des années 1960. Par la suite, d'autres systèmes GNSS ont été développés, notamment en Russie avec **GLobal NAVigation Satellite System (GLONASS)** et en Europe avec Galileo.

La précision des récepteurs GPS étant limitée à 5 à 15 mètres, des solutions d'amélioration de la précision ont été développées. Le **Differential Global Positioning System (D-GPS)** ou **Satellite Based Augmentation System (SBAS)** utilise un réseau de stations de référence dont la position réelle est connue, ces dernières calculent leur position en utilisant les signaux satellites pour déterminer l'erreur de la mesure. Cette erreur est ensuite retransmise aux autres récepteurs GNSS qui l'utilisent pour corriger la position mesurée. La précision des systèmes D-GPS est de l'ordre du mètre. Les erreurs déterminées par les systèmes SBAS ne sont valides qu'à une échelle régionale, c'est pourquoi chaque région développe son propre système. Les systèmes SBAS existants sont Wide

Area Augmentation System (WAAS) pour l'Amérique du Nord, European Geostationary Navigation Overlay Service (EGNOS) pour l'Europe, GPS Aided Geo Augmented Navigation (GAGAN) pour l'Inde et Multi-functional Satellite Augmentation System (MSAS) pour le Japon.

Pour améliorer encore la précision du D-GPS, la cinématique temps réel ou **Real Time Kinematic (RTK)** mesure la phase des ondes porteuses des signaux émis par les satellites. La précision atteinte par les récepteurs RTK est de l'ordre du centimètre.

En supplément des approches traditionnelles de positionnement par satellite, des techniques basées sur l'utilisation de capteurs optiques existent (odométrie visuelle, Simultaneous Localization And Mapping (SLAM)). Grâce à l'analyse d'image, un véhicule peut se localiser. L'avantage de ces méthodes est l'indépendance vis-à-vis du signal satellite qui peut parfois être perdu, et vis-à-vis des stations de références des SBAS qui ne couvrent pas toute la surface terrestre.

Étant donné que les systèmes de positionnement relativement précis (c'est-à-dire avec une précision de l'ordre du mètre) sont amenés à se généraliser, la résolution du système de positionnement sera régulièrement supposée de l'ordre du mètre dans le reste de ce mémoire.

Détection d'obstacles et perception de l'environnement De plus en plus de véhicules sont équipés de systèmes de détection d'obstacles. Ceux-ci permettent de déterminer la position et la vitesse relative des objets situés dans l'environnement proche du véhicule. Certains d'entre eux reposent sur des capteurs à ultrason ou **Sound Navigation And Ranging (SONAR)**, bon marchés, mais à la portée (quelques mètres) et à la précision limitées. D'autres utilisent des émetteur/récepteurs d'ondes électromagnétiques tels que le **Radio Detection And Ranging (RADAR)** ou le **Laser Detection And Ranging (LIDAR)**, la portée est alors de quelques centaines de mètres.

Les **caméras vidéos stéréo** équipent également la plupart des véhicules haut de gamme, ainsi que les véhicules autonomes. Ces dernières servent aussi à l'identification et à la détection des obstacles, mais peuvent également analyser la signalisation routière (lecture des panneaux ou des feux tricolores, et détection des marquages au sol).

2.3.1.2/ SYSTÈMES D'AIDE À LA CONDUITE

La réalisation du véhicule autonome n'est pas un bond technologique soudain, mais le fruit d'un processus itératif. Il a été développé progressivement en concevant, testant et déployant des systèmes isolés destinés à automatiser certaines tâches de la conduite : ce sont les systèmes d'aide à la conduite ou **Advanced Driver Assistance Systems (ADAS)**. Le développement de l'ensemble de ces sous-systèmes et leur intégration permet alors d'automatiser l'ensemble de la conduite.

Les premiers ADAS avaient vocation à augmenter la sécurité en améliorant le contrôle du conducteur sur son véhicule, mais ne remplaçaient pas le contrôle humain. Les plus courants sont le système anti-blocage des roues ou **Antiblockiersystem (ABS)**, l'Assistance au Freinage d'Urgence (AFU) et le correcteur électronique de trajectoire ou Electronic Stability Program (ESP).

Par la suite des ADAS permettant d'automatiser le contrôle longitudinal (gestion de l'accélération et du freinage) ont été développés. Ainsi, le régulateur de vitesse permet au conducteur de ne plus avoir à réguler lui-même la vitesse de son véhicule par une action sur la pédale d'accélération. Il sélectionne sa vitesse cible, puis le régulateur applique la consigne d'accélération nécessaire pour respecter cette vitesse. Utilisé en combinaison avec un RADAR frontal capable de détecter le véhicule précédent, le régulateur devient l'Adaptive Cruise Control (ACC), et la vitesse cible s'adapte automatiquement pour respecter un temps inter-véhiculaire prédéfini. Enfin, l'utilisation du RADAR permet également de freiner le véhicule (freinage automatique d'urgence) pour réduire l'impact de la collision avec un obstacle, voire l'éviter, sans contrôle du conducteur. Enfin, via l'utilisation d'une caméra, les véhicules peuvent détecter panneaux et feux tricolores pour définir la vitesse du véhicule en fonction de la signalisation. A noter toutefois que le cas des intersections n'est pas encore géré par les ADAS commerciaux, en particulier en l'absence de feux.

Après avoir automatisé le contrôle de l'accélération et du freinage en fonction des informations perçues par les capteurs, les ADAS les plus récents automatisent en partie le contrôle de la direction. Les premiers développements autour du contrôle de direction ont visé à automatiser les manœuvres de stationnement, en utilisant le SONAR afin de détecter les obstacles. Plus récemment, les systèmes d'alerte de franchissement de ligne (via caméra) ont été développés, pour détecter la sortie de voie du véhicule, puis améliorés pour maintenir le conducteur dans la voie sans qu'il n'agisse sur le volant. Enfin la dernière innovation dans le contrôle de direction est l'assistance au changement de voie : le conducteur actionne son clignotant pour indiquer qu'il désire changer de voie, puis en utilisant ses capteurs SONAR et RADAR le véhicule peut déterminer si le changement de voie peut être effectué, et si oui l'exécute.

2.3.1.3/ COMMUNICATION SANS FIL

La communication sans fil est une technologie émergente dans le domaine du véhicule, et en 2017, relativement peu de véhicules sont équipés d'unités de communication. Néanmoins les prochaines années verront l'essor de la communication véhiculaire, étant donné les nombreuses possibilités offertes [Meraihi et al., 2008]. Ainsi, les premières applications prévues permettront au véhicule de payer (péage, stationnement) et d'être informé en temps réel d'éventuels événements du trafic.

De nombreux protocoles de communication sans-fil existent tels que le GPRS, GSM, Wi-Fi, 3G, 4G, Bluetooth, etc. Ceux-ci, bien qu'utilisables par les véhicules à des fins d'assistance ou de service, ne remplissent pas les critères nécessaires pour les applications Intelligent Transportation Systems (ITS), et notamment les applications critiques. C'est pourquoi des protocoles spécifiquement dédiés à la communication inter-véhiculaire ont été développés.

Au cours des dernières années, le standard IEEE 802.11p a été développé et imposé comme référence. Ce protocole de communication sert de base au protocole d'échange de messages Wave Short Message Protocol (WSMP) [Ahmed et al., 2013], et au standard européen ETSI ITS-G5 pour les applications ITS. Ces communications sont regroupées sous l'appellation Dedicated Short-Range Communications (DSRC). De plus, ces protocoles prennent également en charge la création de réseaux véhiculaire ad-hoc ou VANET afin d'étendre la portée de la communication.

Le protocole ITS-G5 divise la bande de fréquence attribuée par IEEE 802.11p en 8 canaux en fonction des applications potentielles. En particulier, les canaux G5-CCH (control channel), G5-SCH1 (service channel 1) et G5-SCH2 sont réservés aux applications de sécurité routière et pourraient être exploités par les applications de régulation coopérative des intersections.

2.3.2/ MODÉLISATION DE LA CONDUITE

L'utilisation de capteurs et d'actionneurs ne suffit pas à rendre des véhicules autonomes. Il faut concevoir des algorithmes qui en fonction des perceptions des capteurs vont appliquer les commandes appropriées, de façon à reproduire, puis perfectionner, la conduite humaine.

Pour cela, il faut modéliser la conduite humaine. Différents modèles existent afin de décrire le comportement humain dans des situations spécifiques (planification de trajet, suivi de trajectoire, changement de voie, suivi de véhicules et respect des distances intervéhiculaires).

Ces modèles se répartissent sur trois niveaux [Michon, 1985] :

- Niveau stratégique (planification)
- Niveau tactique (manoeuvre)
- Niveau contrôle

L'intégration de ces différents modèles en un tout permet de modéliser dans son ensemble le comportement d'un conducteur.

Par la suite, dans ce mémoire, nous présentons en détail les modèles et algorithmes liés au suivi de trajectoire et au respect des distances inter-véhiculaires (chapitre 3), et au changement de voie (chapitre 5). Ainsi, nous ne présenterons que brièvement ces modèles ici.

Planification de trajet Le véhicule est un moyen de transport permettant à une ou plusieurs personnes de se déplacer entre une **origine** et une **destination**. Afin de sélectionner l'ensemble des noeuds du réseau routier à traverser pour rallier la destination depuis l'origine, des algorithmes de planification de trajet sont utilisés.

L'objectif est généralement de minimiser le temps, la distance, ou le coût pour faire le trajet origine/destination, c'est pourquoi la planification de trajet repose sur des algorithmes de recherche du plus court chemin dans un graphe, tels que Dijkstra ou A*. Plus récemment, la généralisation des systèmes de guidage routier et l'émergence des ITS ont permis de développer de nouvelles stratégies de recherche de chemin en utilisant les informations de trafic en temps réel et en prédisant le trafic [Fu et al., 2006], [Dong, 2011].

Adaptation de la vitesse et respect des distances intervéhiculaires : contrôle longitudinal Le conducteur doit régulièrement adapter sa vitesse pour :

- Respecter les limitations de vitesse
- Adapter sa vitesse à la trajectoire
- Éviter les collisions avec les obstacles, y compris les véhicules qui le précèdent (respect de la distance intervéhiculaire)

Le rôle d'un modèle de régulation de vitesse (ou Cruise Control (CC)) est de reproduire la façon dont un conducteur humain ajuste sa pression sur la pédale d'accélération afin de respecter une consigne de vitesse. En effet, la relation entre évolution de la vitesse et pression sur les pédales d'accélération et de freinage dépend de nombreux facteurs (masse, vitesse actuelle, frottement de l'air, adhérence, pente, etc.) et nécessite un ajustement permanent.

Ce modèle peut alors servir de base à un modèle d'adaptation de la vitesse pour une trajectoire, dont le rôle est de reproduire l'ajustement de la vitesse réalisé par un être humain en fonction de la trajectoire (par exemple, réduction de la vitesse cible en fonction de la courbure de la route).

Et la régulation de vitesse sert également de base à la régulation de vitesse adaptative (ou ACC) dont le rôle est de reproduire l'ajustement de la vitesse cible en fonction de la présence d'obstacles frontaux, généralement de façon à maintenir un temps intervéhiculaire stable et une distance minimale de sécurité.

Planification et suivi de trajectoire Le conducteur doit rester sur la route et suivre la voie dans laquelle il se trouve (sauf en cas de changement de voie). Pour cela il ajuste la trajectoire du véhicule en agissant sur le volant (contrôle latéral).

Cet ajustement se fait généralement en deux temps :

- Définition d'une trajectoire cible : le conducteur définit la trajectoire à suivre en fonction de son observation et de sa connaissance de la route (détection de voie et informations cartographiques) et d'éventuels obstacles
- Respect de la trajectoire cible : par action sur la direction du véhicule, il s'efforce de respecter au mieux la trajectoire à suivre

Le chapitre 3 détaille précisément les stratégies existantes pour le respect de la trajectoire cible.

Changement de voie Dans la modélisation de la conduite, le rôle d'un modèle de changement de voie est de définir dans quelles conditions un changement de voie peut ou doit être réalisé, et sous quelles conditions il peut être réalisé. Ces modèles reposent généralement sur l'utilisation de modèles de contrôle longitudinal afin de réaliser les adaptations nécessaires en termes de vitesse, et les modèles de contrôle latéral pour définir et respecter la trajectoire propre à l'opération de dépassement. Les modèles existant sont détaillés plus précisément dans le chapitre 6.

Actuellement de tels modèles ne sont pas déployés dans les véhicules commerciaux : même lorsqu'une assistance au changement de voie est présente, le choix de dépasser ou de rester dans la voie est laissé au conducteur.

Stationnement et manœuvre Au départ de son trajet, le conducteur doit généralement quitter un emplacement de stationnement, et à l'arrivée il doit à nouveau se stationner.

Depuis quelques années, des algorithmes ont été conçus et déployés afin d'automatiser ces opérations. Ceux-ci reposent généralement sur des stratégies de contrôle longitudinal et latéral simplifiées et limitées au seul cas du stationnement automatisé. De plus la seule source d'information utilisée est généralement le sonar.

Gestion des intersections et des priorités Un modèle ayant vocation à décrire le comportement d'un être humain doit intégrer la gestion des intersections et des priorités : en fonction des informations provenant des capteurs, le modèle définit la priorité du véhicule, qui peut ensuite être utilisée pour définir trajectoire et vitesse.

Définir le comportement d'un conducteur humain face à une intersection est simple, ces dernières étant généralement explicitement réglementées. Néanmoins il est difficile d'identifier la situation de l'intersection à partir des seuls capteurs. Par exemple, lorsqu'un conducteur humain entre sur un circuit giratoire ou entre sur une intersection aveugle, il doit prendre de l'information, analyser et anticiper le comportement des autres usagers, et réagir rapidement en conséquence. Ainsi l'utilisation des modèles de gestion des priorités et des intersections est souvent limité au cas des intersections par feux tricolores, ou au contexte de la simulation (dans laquelle on peut introduire un biais en donnant à un agent une connaissance complète de son environnement et des intentions des autres agents).

Les protocoles de régulation coopérative apportent une solution simple en permettant de décrire formellement l'ordre de priorité de chaque véhicule dans une intersection.

Modèle général Enfin, différents travaux ont été réalisés afin de définir un modèle comportemental général du conducteur, intégrant les différents sous-modèles précités. On peut citer entre autres les modèles :

- COgnitive SIMulation MOdel of the DRIVEr (COSMODRIVE) [Tattegrain-Veste et al., 1996]
- PAtH DRIVer COgnitive (PADRIC) [Delorme, 2001]
- Le modèle de conducteur intégrateur ACT-R (Adaptive Control of Thought - Rational) [Salvucci, 2006]

Les modèles du comportement du conducteur servent à prédire le comportement humain permettant ainsi d'anticiper des situations dangereuses, mais aussi à le reproduire afin d'automatiser la conduite [AbuAli et al., 2016].

2.4/ MODÉLISATION ET SIMULATION DU SYSTÈME

Les différents constituants de l'intersection autonome sont maintenant présentés. Afin de réaliser l'intersection autonome, il est nécessaire de construire le modèle faisant le lien entre ces éléments, c'est-à-dire entre les perceptions du véhicule et les commandes à appliquer, en utilisant les modèles et algorithmes prédéfinis. Un tel système est par essence :

- hybride : les véhicules se déplacent dans un espace continu tandis que leur droit de passage est discret
- distribué : le système est composé de nombreux véhicules interagissant en communiquant entre eux, et chaque véhicule est lui-même un système composé de sous-systèmes en interaction
- imparfait : la connaissance de l'ensemble du système à un instant donné ne peut être parfaite à cause de l'imprécision des dispositifs de mesure et des défauts de la communication sans-fil

Le grand nombre de composants impliqués et leur hétérogénéité en fait un système complexe à analyser, concevoir et tester.

Un système complexe tel que celui-ci ne saurait être décrit par de simples formules mathématiques ou une structure de donnée élémentaire sans faire d'hypothèses très simplificatrices. Dans un tel cas, afin d'évaluer la faisabilité et l'efficacité du système, on recourt à la modélisation et à la simulation informatique.

Toutefois, les simulations nécessitent une validation expérimentale des résultats. Ceci permettra de confronter les modèles théoriques sous-jacents aux conditions réelles.

L'ingénierie logicielle, et plus généralement l'ingénierie des systèmes, offre des méthodologies et outils donnant un cadre permettant la réalisation d'un tel système, de la modélisation à l'implémentation réelle, en passant par la simulation. Ces outils incluent entre autres la conception orientée composant, l'ingénierie dirigée par les modèles, le développement dirigé par le comportement et la simulation multi-agent.

En vue d'introduire un modèle architectural du système intersection autonome, nous présentons brièvement ces méthodologies et outils.

2.4.1/ INGÉNIERIE DIRIGÉE PAR LES MODÈLES

L'ingénierie dirigée par les modèles (ou Model Driven Engineering (MDE) en anglais) [Kent, 2002] est une pratique d'ingénierie des systèmes utilisant les capacités des technologies informatiques pour décrire au travers de modèles, à la fois le problème posé et sa solution.

La phase d'analyse dans ce processus d'ingénierie consiste donc à réaliser un modèle du système à concevoir, puis à l'exprimer dans un langage de modélisation (Unified Modeling Language (UML) ou Systems Modeling Language (SysML) par exemple). L'objectif de la réalisation d'un modèle du logiciel est théoriquement de pouvoir automatiser la production du code en formalisant la définition du problème.

Néanmoins, même en l'absence de cette automatisation, dans le cadre de l'étude du système, la création d'un modèle permet de définir un cadre formel à la conception de ce système. Cela permet ainsi de réduire les temps de développement et d'améliorer la maintenabilité et l'évolutivité du produit final.

2.4.2/ CONCEPTION ORIENTÉE COMPOSANT

L'une des stratégies de modélisation du système consiste à le penser sous la forme de **composants** interconnectés. Ainsi, la programmation orientée composant (POC) (ou Component-Based Software Engineering (CBSE) en anglais) [Aoyama et al., 1998] consiste à utiliser une approche modulaire de l'architecture d'un projet informatique.

Un composant est, par définition :

- Spécialisé
- Indépendant
- Substituable
- Réutilisable

Ces modules peuvent être testés unitairement (tests unitaires) ou ensemble (tests d'intégration).

En assemblant des composants logiciels reproduisant les composants réels de l'intersection autonome, il est possible de simuler entièrement l'intersection autonome. Cela permet de valider la conception du système, et d'en estimer les performances.

De plus, un composant se définit par son interface, autrement dit par la façon dont il se connecte aux autres éléments du système. En s'assurant que deux composants remplissant la même fonction présentent la même interface, ils deviennent interchangeables. Et si un composant logiciel présente la même interface qu'un composant réel, et se comporte de façon similaire à ce composant réel (produit les mêmes sorties pour les mêmes entrées), il peut se substituer au composant réel, on parle alors de simulation Software-in-the-loop (SIL). Dans l'autre sens, c'est-à-dire en remplaçant un composant logiciel par un composant réel, on parle de simulation HIL.

Plus récemment, des travaux considèrent le système entier comme un composant interagissant avec un environnement, et c'est à un système entier, ou même à l'environnement, qu'on peut substituer un autre système. On parle alors de simulation System-in-the-loop (SITL) [Day et al., 2012] ou encore Vehicle-in-the-loop (VIL) [Bokc et al., 2007] pour les véhicules.

2.4.3/ BEHAVIOR-DRIVEN DEVELOPMENT

Le Behavior-Driven Development (BDD) [North et al., 2006] [Solis et al., 2011] est une méthode de développement logiciel faisant partie de la famille des méthodes Agile [Beck et al., 2001]. Cette méthode définit un certain nombre de pratiques et de conventions dont l'objectif est d'améliorer rapidité et qualité dans le développement d'un logiciel.

Les aspects spécifiques de cette méthode les plus intéressants dans le cadre de l'intersection autonome sont :

- La définition du système (ou de ses composants) sous forme de « comportements » attendus du véhicule ou du serveur d'intersection ; ces comportements sont généralement spécifiés sous la forme « *given* pré-conditions *when* action *then* post-conditions »
- La création de tests unitaires et de tests de non-régression à partir des comportements préalablement définis
- L'utilisation de *mocks*, c'est-à-dire des composants logiciels simulant les composants non encore écrits et répondant aux comportements préalablement définis

2.4.4/ SYSTÈMES MULTI-AGENT

Un système multi-agent est un système composé d'un ensemble d'agents, placés dans un environnement et interagissant entre eux [Weiss, 1999]. Les agents sont au moins partiellement autonomes.

Dans le système « intersection autonome », chaque véhicule peut être vu comme un agent autonome interagissant avec les autres véhicules, au sein de l'environnement qu'est l'infrastructure routière. Les systèmes multi-agents sont donc parfaitement adaptés pour modéliser et simuler le trafic à l'échelle microscopique dans le cadre de la régulation coopérative du trafic [Dresner et al., 2008]. Il est également possible de considérer les différents composants du véhicule comme des agents interagissant entre eux, étant donné

que les composants sont relativement autonomes dans la façon dont ils interprètent les données et les ordres provenant des autres composants.

Le principal bénéfice apporté par les systèmes multi-agent dans l'étude du système « intersection autonome et coopérative » est d'offrir un cadre pour la modélisation et l'étude de l'interaction entre les différents agents. La simulation multi-agent permet ainsi d'étudier les propriétés qui émergent de l'interaction des différents agents. Elle peut ainsi servir à évaluer la distribution des ressources que se partagent les agents, et donc d'estimer la performance de différentes stratégies de régulation, ou éventuellement de mettre au jour des problèmes d'inter-blocage (voir chapitre 4).

2.4.5/ DISCUSSION : APPLICATION À L'INTERSECTION AUTONOME

Dans le cadre de ce mémoire, ces outils de l'ingénierie logicielle et de l'ingénierie système sont utilisés pour concevoir intégralement l'intersection autonome.

Premièrement, un modèle de composant, permettant de spécifier l'architecture globale du système est réalisé. Ce modèle permet de définir précisément les composants matériels et logiciels requis par la conception du système, ainsi que les relations entre ces composants.

Dans un second temps, nous implémentons les composants logiciels préalablement définis en utilisant une stratégie de développement de type BDD. Les composants sont spécifiés par le comportement attendu, puis sont implémentés et intégrés au fur et à mesure dans le système global. En l'absence de certains composants réels ou logiciels, des *mocks* sont utilisés pour permettre de réaliser des tests d'intégration, mais également de réaliser des simulations du système. Nous pouvons alors mettre en évidence d'éventuels défauts de conception, ou bien contribuer à l'amélioration d'un module spécifique tout en nous assurant que les modifications apportées ne dégradent pas le fonctionnement du système (via des tests de non-régression).

Enfin, les principes de la simulation multi-agent sont appliqués afin de simuler l'intégration de plusieurs systèmes isolés et communicants. En effet, dans le système intersection autonome, chaque véhicule autonome est un agent, ainsi que le contrôleur d'intersection. Ces agents évoluent dans un environnement physique continu, dynamique, incertain et complexe, et coopèrent pour atteindre leur objectif personnel (atteindre leur destination) en toute sécurité (en évitant les collisions).

2.5/ MODÈLE D'ARCHITECTURE D'UNE INTERSECTION AUTONOME

Ce chapitre a présenté l'intersection autonome comme un système distribué et complexe, composé de sous-systèmes interagissant entre eux en relative autonomie que sont les véhicules autonomes et l'infrastructure routière (contrôleurs d'intersection, dispositifs de communication et de signalisation). Ces véhicules autonomes ont également été découpés en sous-systèmes : capteurs, actionneurs et logiciels de contrôle. Enfin, ces logiciels permettant de piloter le véhicule en fonction des données provenant des capteurs et de la communication sont eux-mêmes divisibles en sous-composants logiciels (algorithmes de contrôle longitudinal et latéral, modèles de conduite, etc.).

Nous avons donc présenté des outils permettant l'analyse, la modélisation, la conception

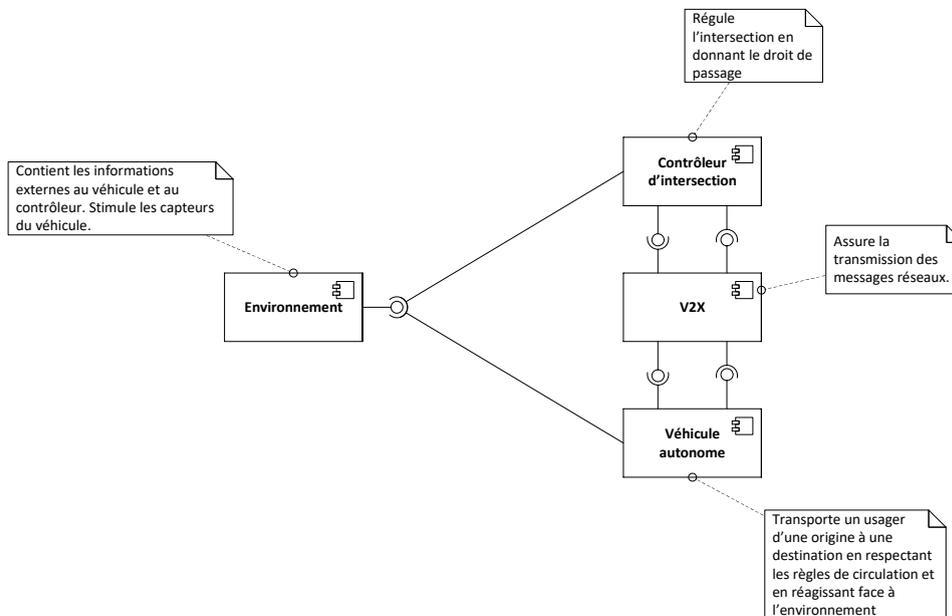


FIGURE 2.9 – Diagramme de composants global

et l'expérimentation du système dans sa globalité. En exploitant ces outils, nous pouvons synthétiser l'ensemble du système par la réalisation d'un modèle de l'intersection autonome intégrant les différents composants préalablement mentionnés.

2.5.1/ DIAGRAMME DE COMPOSANTS

Le système « intersection autonome » dans son ensemble peut être divisé en trois types de composants principaux :

- Le véhicule autonome et coopératif, évoluant sur le réseau routier en fonction des données provenant des capteurs
- Le système de régulation de l'intersection, ordonnant le droit de passage des véhicules à l'intersection
- L'environnement, stimulant les différents capteurs

En définissant l'environnement comme un composant, nous le définissons donc comme étant un élément substituable. Cela signifie qu'à l'environnement réel nous pouvons substituer un environnement virtuel. L'utilisation d'un *mock* pour l'environnement permet de tester des scénarios complexes sans nécessiter de tests réels.

En affinant notre analyse, nous pouvons séparer le système de régulation en différents composants (voir figure 2.10) :

- Le protocole de communication : en charge de la définition et de l'interprétation des messages servant à la négociation
- La politique de régulation : en charge de gérer l'attribution du droit de passage de la façon la plus efficace possible
- Le système de communication : en charge de la transmission et de la réception des messages

A noter, les modèles des systèmes centralisés et décentralisés différant, nous ne nous focalisons ici que sur le système centralisé tel que défini dans [Perronnet et al., 2012]. Dans un système décentralisé, les éléments du système de régulation apparaîtraient comme sous-composants du véhicule.

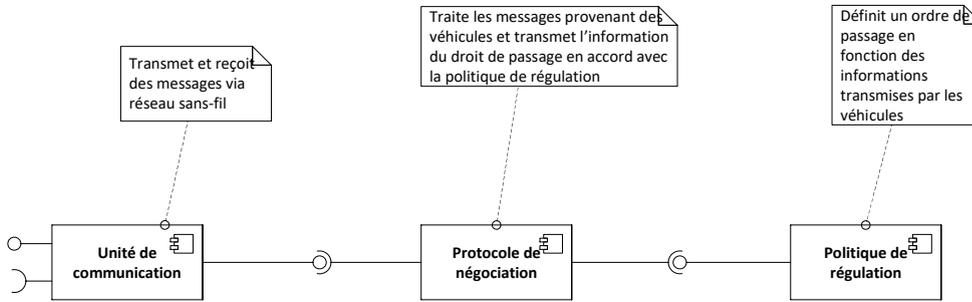


FIGURE 2.10 – Diagramme de composants du contrôleur d'intersection

En procédant de la même façon pour le véhicule autonome et coopératif, nous pouvons définir les différents sous-composants du système (voir figure 2.11).

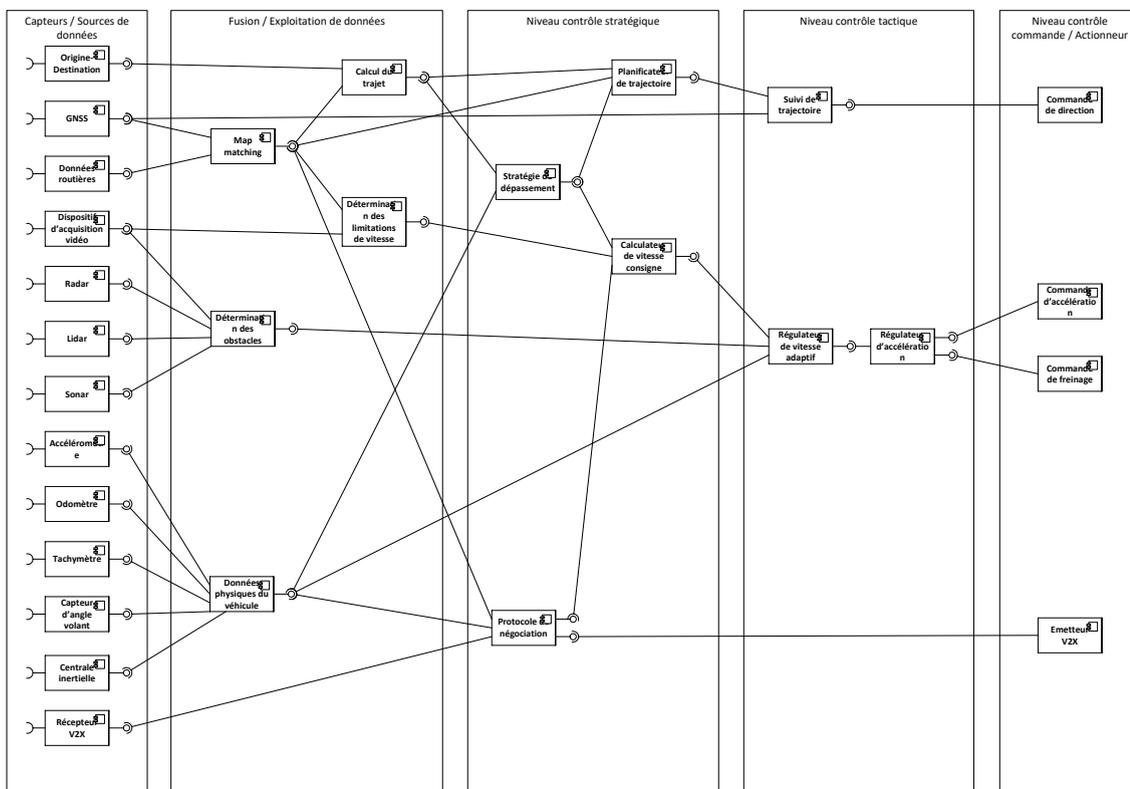


FIGURE 2.11 – Diagramme de composants du véhicule autonome et coopératif

Nous divisons le modèle architectural en cinq niveaux allant du capteur (et de la donnée) jusqu'à la commande :

1. Le niveau « capteurs / données » s'interface avec l'environnement et produit des données brutes
2. Le niveau « fusion / exploitation de données » agrège les données provenant des différents capteurs et y attache une sémantique
3. Le niveau « contrôle stratégique » récupère ces données afin de définir les orientations générales du système de conduite
4. Le niveau « contrôle tactique » opère de façon à transformer les décisions stratégiques en commandes directement applicables
5. Le niveau « contrôle commande / actionneurs » représente les commandes physiques du véhicule

Les contributions présentées dans ce mémoire se concentrent sur les niveaux stratégiques et tactiques.

A noter, le diagramme de composant présenté dans la figure 2.11 n'est pas exhaustif. D'une part, d'autres composants peuvent toujours être intégrés au système, et les composants présentés sont eux-mêmes composés d'autres sous-composants. D'autre part, par souci de clarté certaines dépendances, notamment entre capteurs, ne sont pas affichées. Enfin, il ne constitue pas une référence de l'architecture d'un véhicule autonome et coopératif, mais représente néanmoins le modèle utilisé pour les développements réalisés dans le cadre de cette thèse.

Ainsi, il permet de situer la position des différentes contributions présentées subséquemment dans un schéma global du système.

2.6/ CONCLUSION

Ce chapitre d'introduction présente le contexte dans lequel le concept d'intersection autonome prend racine. Nous montrons ainsi qu'il apparaît comme une évolution logique des systèmes de régulation du trafic existant, s'appuyant sur l'émergence des technologies liées au véhicule autonome, aux systèmes de transports intelligents et à la communication inter-véhiculaire. Nous présentons ensuite une étude des travaux existants liés à la régulation coopérative des intersections, avec une attention portée sur l'intersection autonome, c'est-à-dire l'intersection de véhicules autonomes par communication sans-fil et sans intervention humaine.

L'intersection autonome étant intrinsèquement liée au développement du véhicule autonome et de la communication inter-véhiculaire, nous nous attachons à présenter un état de l'art de ce domaine. De cette analyse émerge une liste de composants et de modèles devant s'intégrer afin de composer l'intersection autonome. Il en ressort qu'il s'agit d'un système complexe à concevoir et valider. C'est pourquoi nous présentons une série d'outils permettant de modéliser ce système, mis en application par la réalisation d'un modèle de composants du système faisant la synthèse des éléments présentés dans ce chapitre.

À l'instant de la rédaction de ce mémoire, le système « intersection autonome et coopérative » demeure essentiellement théorique. Ainsi, de par notre volonté d'augmenter la

maturité technologique du système, et de faire un pas en direction des phases de standardisation, puis de déploiement, nous travaillons à l'amélioration de ses composants.

Ainsi, dans les chapitres suivants, nous présentons différentes contributions à l'intersection autonome, concernant la commande du véhicule (chapitre 3), le protocole de régulation (chapitre 4), et la réalisation d'un prototype réel (chapitre 5). Bien qu'apparaissant disparates, car concernant des domaines différents, ces contributions s'intègrent dans un même ensemble que nous venons de définir dans le présent chapitre. Ainsi, la conception, et par conséquent la validation, du concept d'intersection autonome repose sur chacune de ces contributions et leur intégration.



CONTRIBUTIONS À LA CONCEPTION DE L'INTERSECTION AUTONOME

COMMANDE DU VÉHICULE

La conception d'une intersection autonome repose sur le véhicule autonome. Ce dernier doit être capable d'adapter sa vitesse à partir des informations provenant des différents capteurs, et également de suivre une trajectoire prédéfinie. Il est donc nécessaire d'appliquer une stratégie de contrôle du véhicule autonome permettant de piloter le véhicule en accélération et en direction.

Ainsi, le contrôle du véhicule se fait de deux façons :

- Le contrôle longitudinal : en fonction des données renvoyées par les différents capteurs (radar, position, etc.) et d'une stratégie de suivi, les commandes d'accélération et de freinage sont actionnées.
- Le contrôle latéral : en fonction des données provenant des différents capteurs, une action est appliquée sur le volant pour diriger le véhicule.

Dans le cadre spécifique de la régulation coopérative du trafic appliquée aux véhicules autonomes, c'est-à-dire de l'intersection autonome, l'une des sources de données influençant le comportement du véhicule est la communication sans-fil.

En effet, nous avons vu que dans le cas du protocole TIM, les véhicules synchronisent leur vitesse à l'intersection en utilisant la position et la vitesse des véhicules, transmises par le contrôleur d'intersection. Ces données sont actualisées à chaque réception de message par le véhicule. La problématique qui en découle est qu'il faut choisir une solution de contrôle prenant en compte cet aspect, évitant ainsi les oscillations et les risques de collision.

Dans le cas de l'intersection autonome, le contrôle longitudinal ou ACC est coopératif (on parle alors de Cooperative Adaptive Cruise Control (CACC)) car l'information sur le véhicule à suivre provient de la communication réseau. La littérature propose différentes stratégies de suivi, ainsi nous réalisons une comparaison des principales solutions existantes avec vérification en simulation et en conditions réelles. L'objectif est de définir et valider une stratégie répondant aux contraintes de l'intersection autonome.

En ce qui concerne le contrôle latéral, la littérature propose également différentes solutions afin d'ajuster l'angle du volant en fonction d'un trajet à suivre. Dans ce chapitre nous produisons un état de l'art des solutions les plus utilisées, et proposons une contribution à la commande latérale en fournissant une stratégie de contrôle adaptée à un positionnement GPS, capable de résister à la fois à l'imprécision et à la latence du système de positionnement, et au délai d'application de la commande.

3.1/ CONTRÔLE LONGITUDINAL

Le principe du contrôle longitudinal d'un véhicule est d'adapter la vitesse de ce dernier en fonction de différentes contraintes. Un régulateur de vitesse simple, par exemple, se charge de maintenir la vitesse du véhicule constante par rapport à une programmation donnée. Un régulateur de vitesse adaptatif ajuste la vitesse du véhicule en fonction des obstacles situés devant le véhicule, de manière à éviter tout risque de collision (en cas d'absence d'obstacle, le régulateur applique la vitesse consigne). Enfin, un régulateur de vitesse coopératif ou CACC ajuste la vitesse du véhicule en fonction d'informations transmises via V2X.

Dans un ACC, la régulation se fait généralement sur deux niveaux [Liang et al., 1999] :

- haut-niveau : une accélération consigne à respecter est calculée pour éviter la collision avec un obstacle ; elle est donc calculée en fonction de la distance et de la vitesse de cet éventuel obstacle
- bas-niveau : un système de régulation se charge d'appliquer l'accélération consigne

Dans cette section, nous nous focaliserons principalement sur la partie « haut-niveau », c'est-à-dire le calcul de l'accélération consigne. Dans l'intersection autonome, un véhicule est amené à ajuster sa vitesse en fonction des obstacles perçus via ses capteurs (radar, lidar), mais aussi en fonction des informations fournies par le contrôleur d'intersection. Un véhicule prioritaire à l'intersection constitue un obstacle virtuel, et un véhicule désirent franchir une intersection doit réguler sa vitesse en fonction de celui-ci. De ce fait, nous pouvons dire que les véhicules non prioritaires suivent ceux qui le sont.

Afin de permettre à un véhicule de réguler sa vitesse par rapport aux véhicules prioritaires (et donc de réaliser une synchronisation de vitesse), nous définissons par l'équation 3.1 la distance inter-véhiculaire s_{int} entre deux véhicules en mouvement conflictuel à une intersection, avec x_l la distance du véhicule leader à la sortie de la zone de conflit, l_l la longueur du véhicule leader et x_f la distance du véhicule suiveur à la zone de conflit (voir figure 3.1). Cette distance s_{int} donne la consigne d'accélération a_{int} par l'application d'une stratégie d'ACC.

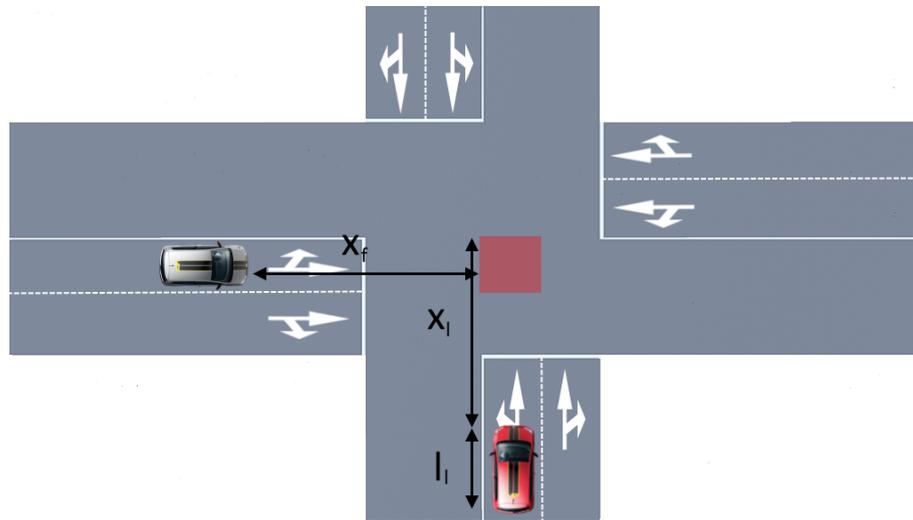
$$s_{int} = x_f - (x_l + l_l) \quad (3.1)$$

D'après cette définition, $x_f < x_l + l_l \Rightarrow s_{int} < 0$, autrement dit le véhicule suiveur doit se placer derrière le véhicule leader (une distance négative entraînant nécessairement un freinage de la part de la commande d'ACC). Néanmoins, pour éviter un freinage trop brusque, la consigne de freinage a_{int} résultant de l'utilisation de cette distance s_{int} est bornée à la valeur b_f (valeur de décélération confortable). D'autre part, pour éviter la création d'un espace vide entre le véhicule et l'intersection, l'accélération relative à la distance à l'intersection a_s est utilisée quand $s_{int} < 0$.

La commande d'accélération résultante est donnée par 3.2 où s_0 représente une distance minimale de sécurité.

$$a_v = \begin{cases} a_s & \text{si } s_{int} - s_0 < 0 \\ \max(a_{int}, b_f) & \text{avec } b_f < 0 \text{ sinon} \end{cases} \quad (3.2)$$

En accord avec cette stratégie de synchronisation des vitesses à l'intersection, nous no-

FIGURE 3.1 – Composantes du calcul de la distance s_{int} entre deux véhicules

tons qu'ajuster la vitesse d'un véhicule en fonction d'un obstacle virtuel demande de faire face à certaines contraintes :

- La position du véhicule à suivre est déterminée par positionnement satellite, donc de façon discrète et avec une précision limitée
- Cette position est transmise à l'intersection de façon discrète, avec un risque de perte de message
- Cette position est elle-même retransmise aux véhicules suiveurs de façon discrète, et ils la comparent avec l'estimation de leur position, déterminée également par satellite

Ainsi, au contraire d'un suivi par radar classique, le suivi par positionnement satellite et communication sans-fil doit intégrer une notion de temps de latence, la position relative des véhicules pouvant dater.

3.1.1/ SOLUTIONS CLASSIQUES

Différents modèles de contrôle longitudinal existent dans la littérature pour la régulation de vitesse adaptative. Ces modèles définissent une consigne d'accélération en fonction de paramètres propre au véhicule (tels que la vitesse actuelle) et de paramètres relatifs aux autres véhicules (tels que la distance et la vitesse relative de ceux-ci).

3.1.1.1/ RÉGULATEUR PROPORTIONNEL, INTÉGRATEUR, DÉRIVATEUR (PID)

Le régulateur Proportionnel, Intégrateur, Dérivateur (PID) est le régulateur le plus utilisé dans l'industrie. Son objectif est de faire correspondre une grandeur mesurée à une grandeur de référence en agissant sur un actionneur. La consigne d'accélération $u(t)$ est calculée par l'équation 3.3 où $e(t)$ est la différence entre la grandeur mesurée et la grandeur de référence, et (K_p, K_i, K_d) sont des coefficients.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (3.3)$$

Dans le cadre d'un ACC, la grandeur à réguler est le temps inter-véhiculaire en agissant sur la consigne d'accélération. Le calcul de la consigne d'accélération peut être calculé par un régulateur PID [Ioannou et al., 1993] [Kreuzen, 2012]. Ainsi $e(t)$ est la différence entre le temps inter-véhiculaire désiré (généralement 2 secondes) et celui qui est mesuré.

De manière générale, ce régulateur est particulièrement bien adapté lorsque la grandeur physique à réguler évolue de manière relativement continue. Cependant, dans le cas de la régulation de vitesse aux intersections, la connaissance de la distance relative à l'obstacle virtuel dépend de la communication : les messages sont envoyés et reçus de manière discrète, avec des temps relativement longs entre deux messages. Dans le cas d'un freinage d'urgence du véhicule suivi, le régulateur PID n'offre aucune garantie d'éviter la collision si le véhicule suiveur ne reçoit pas immédiatement l'information.

3.1.1.2/ MODEL PREDICTIVE CONTROLLER (MPC)

Le régulateur Model Predictive Control (MPC), ou commande prédictive, utilise un modèle mathématique du système à contrôler afin de prédire l'évolution de la grandeur à réguler en fonction d'une séquence d'actions. Dans le cadre d'un système CACC ce régulateur donne de bons résultats [Kreuzen, 2012].

Toutefois, contrairement au cas du CACC, dans une intersection autonome lorsque le véhicule à suivre se trouve dans une autre voie, il est difficile de modéliser son évolution car elle dépend de paramètres non connus (notamment les caractéristiques physiques du véhicule suivi, et la présence ou non d'obstacles sur la trajectoire du véhicule suivi).

3.1.1.3/ LE MODÈLE DE GIPPS

Au contraire de PID et MPC, le modèle de Gipps [Gipps, 1981] donné par 3.4 (et dont les paramètres sont donnés dans la table 3.1) n'est pas un régulateur mais est un modèle de suivi de véhicule. Son objectif est de déterminer l'accélération qu'un conducteur humain aurait en fonction de ses perceptions, qui sont ici la position et la vitesse relatives du véhicule précédent.

$$v_n(t + \tau) = \min\left\{v_n(t) + 2.5a_n\tau\left(1 - \frac{v_n(t)}{V_n}\right)\left(0.025 + \frac{v_n(t)}{V_n}\right)^{1/2}, \right. \\ \left. b_n\tau + \sqrt{b_n^2\tau^2 - b_n\left[2[x_{n-1}(t) - s_{n-1} - x_n(t)] - v_n(t)\tau - \frac{v_{n-1}(t)^2}{b}\right]}\right\} \quad (3.4)$$

a_n	Accélération maximale du véhicule
b_n	Freinage maximal du véhicule $b_n < 0$
s_n	Taille du véhicule n augmentée d'une marge de sécurité
V_n	Vitesse désirée du véhicule n
$x_n(t)$	Position de l'avant du véhicule n à l'instant t
$v_n(t)$	Vitesse du véhicule n à l'instant t
τ	Temps de réaction du véhicule

TABLE 3.1 – Paramètres du modèle de Gipps

L'une des limitations de ce modèle est l'absence de considération d'une décélération désirée : ce modèle permet de garantir l'absence de collision mais ne permet pas de garantir un freinage confortable [Treiber et al., 2013]. Tandis que ce modèle convient pour la simulation du comportement de suivi d'un conducteur humain sur une voie, il n'est pas exploitable tel quel pour la synchronisation de vitesse aux intersections.

3.1.1.4/ INTELLIGENT DRIVER MODEL (IDM) ET COOPERATIVE IDM

Tout comme le modèle de Gipps, les modèles Intelligent Driver Model (IDM) [Treiber et al., 2000] et *enhanced IDM* [Kesting et al., 2010] sont des modèles de suivi de véhicule. Connaissant la vitesse du véhicule suivi et la distance inter-véhiculaire, IDM estime l'accélération qu'un conducteur humain aurait dans des conditions similaires. Cette estimation peut alors être utilisée par un ACC.

$$\dot{v}_a = a^{(\alpha)} \left[1 - \left(\frac{v_a}{v_0^{(\alpha)}} \right)^\delta - \left(\frac{s^*(v_a, \Delta v_a)}{s_a} \right)^2 \right] \quad (3.5)$$

$$s^*(v, \Delta v) = s_0^{(\alpha)} + s_1^{(\alpha)} \sqrt{\frac{v}{v_0^{(\alpha)}}} + T^{(\alpha)} v + \frac{v \Delta v}{2 \sqrt{a^{(\alpha)} b^{(\alpha)}}} \quad (3.6)$$

v_0	Vitesse désirée
s_0	Espace inter-véhiculaire minimal
T	Temps inter-véhiculaire désiré
V_n	Accélération maximale
a	Décélération maximale désirée (ou décélération confortable)
b	Vitesse du véhicule n à l'instant t
δ	Coefficient d'« agressivité » (généralement défini à 4)

TABLE 3.2 – Paramètres d'IDM

Au contraire de PID et MPC cette solution ne requiert pas de paramétrage complet et est relativement simple à mettre en place. Cependant il s'agit d'une fonction continue. L'application de cette formule de façon discrète avec un échantillonnage discret des mesures physiques (vitesse, espace inter-véhiculaire) ne permet pas de garantir que la valeur de décélération confortable sera respectée, ni même qu'il n'y aura pas de collision avec le véhicule suivi [Perronnet, 2015].

Dans [Perronnet et al., 2013a], une adaptation d'IDM nommée Cooperative Intelligent Driver Model (CIDM) a été proposée pour le cas de l'intersection coopérative.

$$a_f = a \left[1 - \left(\frac{v_f}{v_{max}} \right)^\delta - \left(\frac{s^*}{d_{af}} \right)^2 \right] \quad (3.7)$$

$$s^* = s_0 + v_f \left(\frac{d_{el}}{v_f'} + T \right) + \frac{v_f \Delta v'}{2 \sqrt{ab}} \quad (3.8)$$

Dans CIDM, la distance inter-véhiculaire s est remplacée par d_{af} distance à la zone de conflit. Au contraire de la version classique d'IDM, les données sont ici supposées provenir de la communication réseau, afin d'intégrer la latence de la communication et pour prévenir tout risque de collision, CIDM suppose que le véhicule suivi a réalisé un freinage d'urgence immédiatement après avoir transmis les dernières informations (pire cas possible). Ainsi, dans la relation, la vitesse du véhicule suivi devient $v'_l = v_l - b_{max} \times t_s$ où b_{max} est la décélération maximale et t_s l'âge de la dernière information concernant le véhicule. $\frac{d_{af}}{v'_l}$ représente le temps estimé pour que le véhicule suivi ait évacué complètement la zone de conflit.

3.1.1.5/ AUTRES MODÈLES DE SUIVI

Nous venons de présenter brièvement les deux modèles de suivi les plus répandus dans la littérature, à savoir le modèle de Gipps et IDM. Néanmoins, il en existe beaucoup d'autres. Nous pouvons citer entre autres [Krauβ, 1998], [Leclercq, 2007], [Das et al., 2012], une revue plus complète des modèles de suivi existants étant donnée par [van Wageningen-Kessels et al., 2015].

Toutefois, ces modèles sont conçus pour répondre au même objectif que celui de tous les modèles de suivi : simuler le comportement humain le plus précisément possible. La qualité de ces modèles se mesure à leur aptitude à reproduire des observations statistiques concernant la densité, le débit, les accélérations moyennes, etc. obtenues à partir de conducteurs humains. Cependant un modèle reproduisant le comportement humain n'est pas nécessairement le meilleur modèle pour la synchronisation de vitesse aux intersections car il n'a pas vocation à intégrer pas les problématiques purement informatiques que cela pose, à savoir la discrétisation et la latence de l'information de positionnement relatif.

3.1.1.6/ REACTION TIME - ACC (RT-ACC)

Comme IDM, le modèle Reaction Time - Adaptive Cruise Control (RT-ACC) [Hao et al., 2015] [Lombard et al., 2015] n'est pas un régulateur, mais il n'est pas non plus un modèle de suivi traditionnel en ce qu'il ne se destine pas à modéliser le comportement humain. La relation 3.9 définit une consigne d'accélération en fonction de différents paramètres (vitesse, espace inter-véhiculaire, etc.) de façon à éviter tout risque de collision, et en intégrant notamment l'aspect discret du système par l'ajout d'un paramètre « temps de réaction » τ .

$$a_r = \frac{b_f \tau - 2v_f - 2b_f \sqrt{\frac{b_f b_l \tau^2 + 4b_l v_f \tau + 4v_f^2 - 8b_l s}{4b_f b_l}}}{2\tau} \quad (3.9)$$

Ce modèle n'intègre pas directement la notion de vitesse cible, ainsi il doit être utilisé en conjonction avec un régulateur de vitesse. L'accélération consigne finale est le minimum entre la consigne d'accélération a_r et la consigne du régulateur.

Dans le cadre d'un système CACC ou d'intersection autonome, le paramètre τ permet de prendre en compte le fait que la dernière information relative au véhicule suivi date d'au plus τ secondes. Ainsi, si le véhicule suivi effectue un freinage d'urgence (décélération

b_f	Décélération maximale désirée du véhicule suiveur
b_l	Décélération maximale du véhicule suivi
τ	Temps de réaction maximal
v_f	Vitesse du véhicule suiveur
v_l	Vitesse du véhicule suivi

TABLE 3.3 – Paramètres de RT-ACC

valant b_l), et que le véhicule suiveur reçoit cette information avant τ secondes, il n'y a pas de risque de collision.

3.1.2/ RT-ACC : VALIDATION EXPÉRIMENTALE

Afin de valider l'utilisation de RT-ACC pour le cas de l'intersection autonome, des tests ont été réalisés, en simulation et en condition réelle. Les objectifs de ces tests sont :

- Tester la faisabilité et l'efficacité de la solution proposée
- Evaluer l'impact des délais dû à la fréquence d'acquisition des informations et à la latence de transmission de celles-ci
- Evaluer l'impact du délai d'application de la commande du véhicule

3.1.2.1/ VALIDATION EN SIMULATION : COMPARAISON CIDM ET RT-ACC

Pour comparer CIDM et RT-ACC dans le cadre de l'intersection autonome, une simulation est réalisée (voir figure 3.2). Les véhicules utilisent TIM afin de négocier leur droit de passage selon la politique First-come, first-served (FCFS). L'accélération du véhicule est pilotée en utilisant soit CIDM, soit RT-ACC. Les accélérations effectives des véhicules, leur vitesse et leur position sont ensuite mesurées.

Pour ces tests, les valeurs de b et b_f définissent une décélération confortable valant $-2m.s^{-2}$, et les valeurs de T et τ sont de $2s$. De plus, des erreurs sont introduites pour simuler le comportement de capteurs et d'actionneurs réels :

- Délai de communication e_{tc} : temps moyen de transmission d'une information d'un véhicule vers le contrôleur d'intersection ou du contrôleur vers le véhicule
- Erreur de position e_p : la position fournie par le GPS simulé du véhicule se trouve dans un cercle de diamètre e_p autour de la position réelle
- Erreur du radar frontal e_{rf} : la distance retournée par le radar frontal est égale à la distance réelle multipliée par k avec k pris uniformément dans $[1 - e_{rf}; 1 + e_{rf}]$
- Temps de réaction tr : le temps entre deux prises de décision du véhicule, pris uniformément dans $[tr_{min}; tr_{max}]$

Dans le scénario de simulation présenté ces valeurs sont fixées à $e_{tc} = 0.5s$, $e_p = 2m$, $e_{rf} = 0.1$, $tr_{min} = 0.2$, $tr_{max} = 1.8s$.

Nous pouvons observer sur la figure 3.3 que la décélération confortable b n'est pas respectée en utilisant CIDM ($-7m.s^{-2}$ pour le véhicule 3, $-4m.s^{-2}$ pour le véhicule 6), à l'inverse de RT-ACC (figure 3.4).

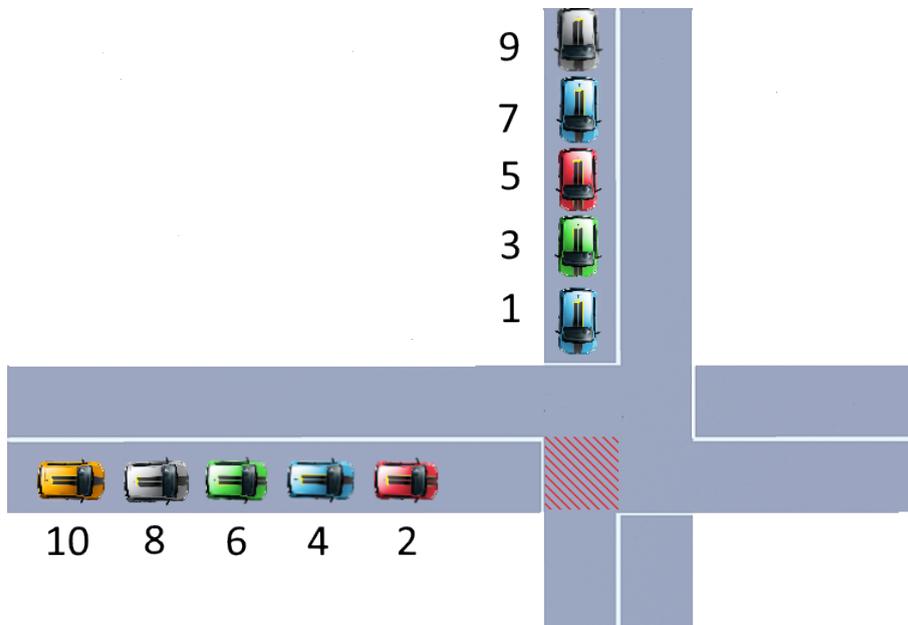


FIGURE 3.2 – Scénario de simulation de comparaison entre CIDM et RT-ACC

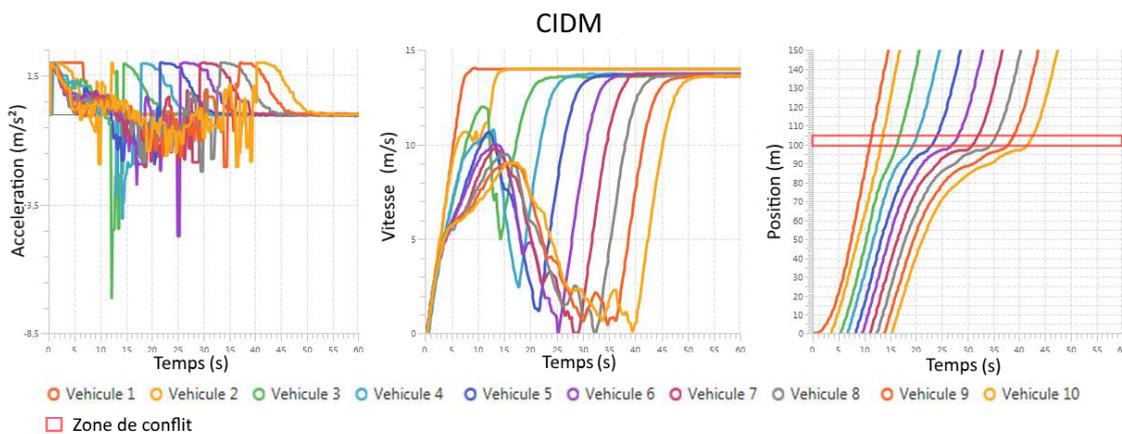


FIGURE 3.3 – Mesures obtenues en pilotant l'accélération des véhicules via CIDM

3.1.2.2/ VALIDATION EN CONDITIONS RÉELLES

Pour valider expérimentalement la commande longitudinale RT-ACC en conditions réelles, il faut disposer d'un véhicule dont les commandes d'accélération et de freinage sont pilotables.

Le test effectué a consisté à laisser un véhicule Renault Scenic pilotable en accélération et freinage, et équipé d'un radar frontal, avancer sur une voie plane. Un obstacle est placé au bout de la voie, le véhicule doit alors décélérer en fonction de la distance à l'obstacle renvoyée par le radar, en utilisant la consigne d'accélération donnée par RT-ACC. La conversion entre consigne d'accélération et pression pédale est réalisée par un régulateur à modèle interne.

Dans le cadre du test τ a été fixé à 2s, et b_f à $-2m.s^{-2}$. Afin de convertir l'accélération désirée (retournée par RT-ACC) en une pression pédale (pourcentage de 0 à 100% pour la

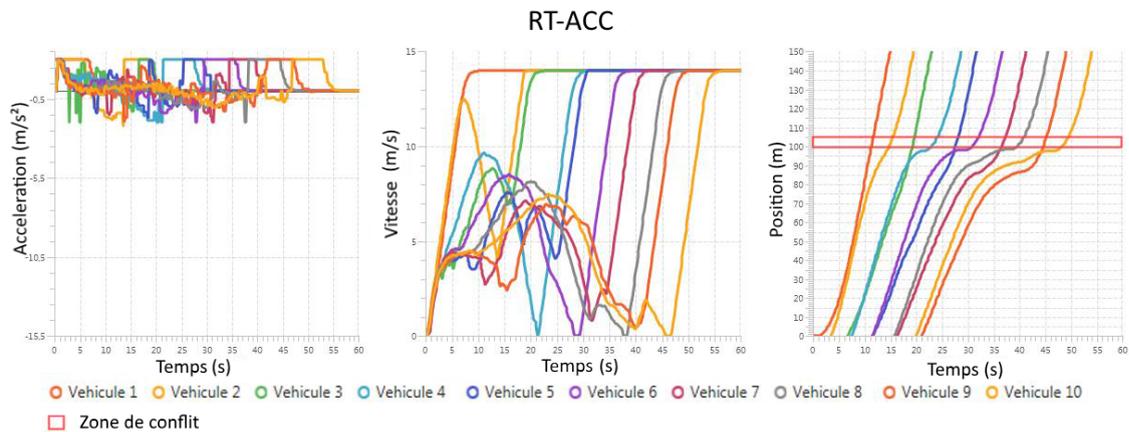


FIGURE 3.4 – Mesures obtenues en pilotant l'accélération des véhicules via RT-ACC

pédale d'accélération ou la pédale de freinage), nous avons réalisé des mesures permettant d'établir une table donnant la valeur d'accélération (ou de décélération) en fonction d'une pression pédale et de la vitesse actuelle du véhicule.

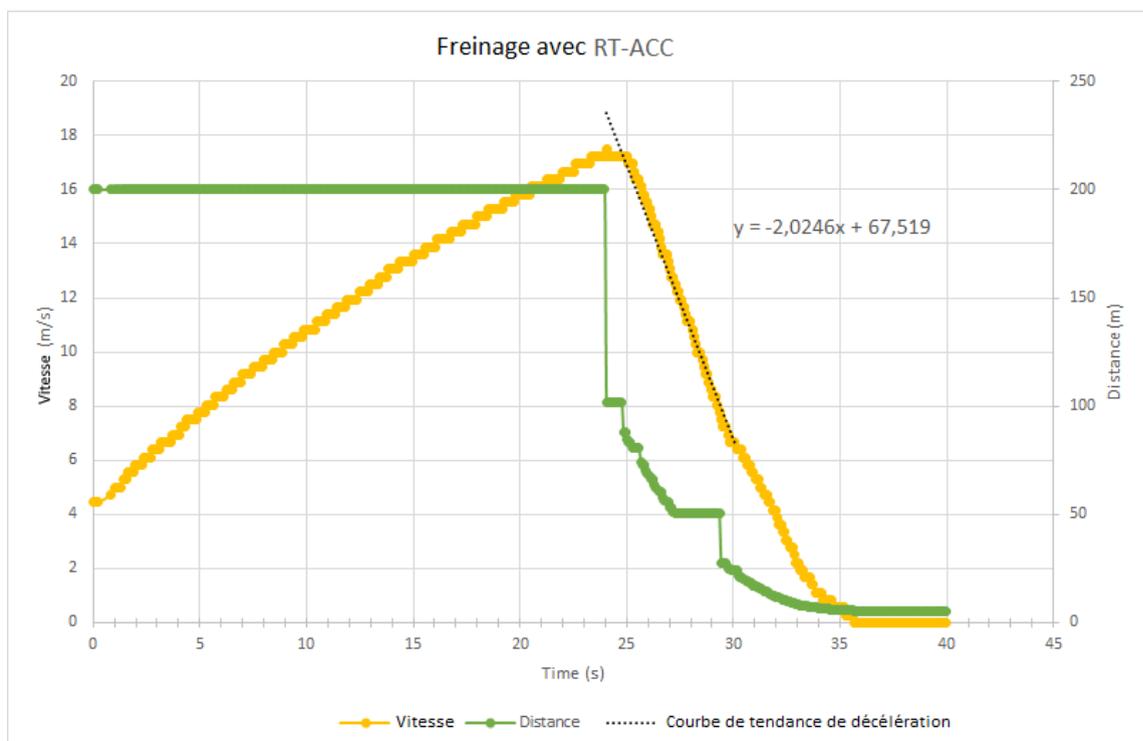


FIGURE 3.5 – Mesure de la vitesse du véhicule et de la distance à l'obstacle avec application de la commande RT-ACC

La figure 3.5 montre des mesures de vitesses et de position (relative) obtenues lors du test. Nous pouvons constater que malgré les problèmes de détection du radar (allant ici jusqu'à 2 secondes sans détection de l'obstacle entre $t = 27s$ et $t = 29s$), la décélération a été bornée à la valeur cible $-2m.s^{-2}$ (voir la courbe de tendance). Les résultats attendus à partir de l'analyse théorique et des simulations sont donc confirmés, de plus la faisabilité et la possibilité de mettre en place cette stratégie de contrôle longitudinal dans un

véhicule réel ont été validées.

3.1.3/ CONCLUSION (CONTRÔLE LONGITUDINAL)

Nous avons analysé les différentes solutions existantes pour le contrôle longitudinal du véhicule. Parmi le choix à disposition, nous avons retenu RT-ACC dont les caractéristiques permettent de contrôler avec précision la décélération du véhicule, même lorsque la source d'information servant au contrôle longitudinal présente une forte latence. Cela rend cette solution bien adaptée au cas de l'intersection autonome.

Nous avons ensuite validé expérimentalement cette solution en effectuant des simulations et des tests en conditions réelles. Les simulations montrent l'intérêt de la solution RT-ACC par rapport aux solutions basées sur IDM, notamment CIDM. Le test réel d'arrêt sur obstacle montre l'efficacité du respect de la commande consigne, même en cas de perte de la détection de l'obstacle pendant un temps $t < \tau$.

3.2/ CONTRÔLE LATÉRAL

Le contrôle latéral d'un véhicule routier consiste à définir l'angle du volant à appliquer en fonction d'informations de positionnement. Deux aspects de la commande latérale peuvent être distingués :

- La planification de chemin : en fonction de données sur la position du véhicule, la position relative d'obstacles, et un chemin à suivre, une trajectoire de référence à suivre est construite.
- Le suivi de trajectoire : la trajectoire de référence à suivre étant préalablement définie, la commande à appliquer au volant est calculée de façon à minimiser l'écart entre la position du véhicule et la trajectoire de référence.

Dans le cadre de l'intersection autonome, les véhicules doivent se situer par rapport à un tracé de référence qu'ils doivent suivre. Afin d'obtenir une grande précision sur la position du véhicule, la solution de positionnement utilisée est le GPS RTK. Ce système de positionnement est discret (la position n'est connue qu'à une fréquence fixe) et comporte des erreurs (imprécision et perte de signal). Pour suivre le tracé de référence, une solution de contrôle latéral doit être implémentée, ajustant la position du volant (et donc l'angle des roues du véhicule et sa direction) en fonction de la position transmise par le GPS.

3.2.1/ SOLUTIONS CLASSIQUES

Différentes solutions de contrôle latéral de véhicules existent dans la littérature afin de définir l'angle $\delta(t)$ à appliquer aux roues du véhicule à l'instant t , en fonction entre autres de l'empattement E du véhicule, la distance à la trajectoire de référence et de l'angle entre la tangente à la trajectoire de référence et la direction du véhicule. Dans cette section nous présentons brièvement les solutions les plus courantes en accord avec [Snider et al., 2009].

Les solutions de contrôle latéral ne sont évidemment pas limitées à ce bref aperçu. On peut notamment citer [Normey-Rico et al., 2001] et [Zhao et al., 2012] qui utilisent un

contrôleur PID adaptatif pour suivre le tracé, [Antonelli et al., 2007] et [Huang et al., 2010] utilisent un contrôleur flou, et [Falcone, 2007], [Abbas, 2011] utilisent un MPC. Cependant, ces stratégies souffrent de défauts ou ne satisfont pas les prérequis que nous avons préalablement posés : les paramètres du PID sont difficiles à configurer et cette stratégie tend à sortir de la trajectoire en fin de courbe (phénomène d'*overshoot*), les contrôleurs flous nécessitent également un paramétrage précis, et les MPC requièrent une forte puissance de calcul et une faible latence (dans l'acquisition des informations et l'application de la commande).

3.2.1.1/ PURE POURSUITE

La pure poursuite a été initialement développée pour diriger un missile vers une cible [Scharf et al., 1969], puis a été employée dans le domaine de la robotique par [Wallace et al., 1985], avant d'être utilisée pour le contrôle de véhicules [Coulter, 1992]. C'est une solution très utilisée pour le contrôle latéral de robots ou de véhicules [Samuel et al., 2016]. Son principe est simple et son implémentation l'est autant : la position du véhicule est projetée sur une trajectoire de référence, puis un point est défini en amont de cette projection sur la trajectoire, le véhicule détermine l'angle du volant de façon à ce que la trajectoire du véhicule atteigne ce point. Etant donné le rayon R du cercle passant par l'essieu arrière du véhicule et le point cible, et admettant la droite dirigée par la direction véhicule comme tangente, l'angle δ est donné par 3.10.

$$\delta(t) = \tan^{-1} \left(\frac{E}{R} \right) \quad (3.10)$$

La position du point cible influence grandement la qualité du suivi. Il est courant de placer ce point à une distance $d = k \times v(t)$ en amont du véhicule, avec $v(t)$ la vitesse du véhicule, et k un coefficient multiplicateur.

3.2.1.2/ MÉTHODE DE STANLEY

La méthode de Stanley [Thrun et al., 2006] a été utilisée lors du *DARPA challenge*. L'angle des roues γ est défini par 3.11, avec θ_e la différence entre la direction du véhicule et la direction de la trajectoire, e_{fa} la distance entre l'essieu avant et la trajectoire, $v_x(t)$ la vitesse du véhicule et k un paramètre de gain.

$$\delta(t) = \theta_e(t) + \tan^{-1} \left(\frac{ke_{fa}(t)}{v_x(t)} \right) \quad (3.11)$$

3.2.1.3/ COMMANDE LINÉAIRE QUADRATIQUE

[Cho et al., 1995] décrit la dynamique du système par un ensemble d'équations différentielles linéaires avec un coût décrit par une fonction quadratique. Ensuite, une commande linéaire quadratique (LQ) est utilisée pour déterminer l'angle δ .

La conception d'une solution de contrôle latérale en utilisant une commande LQ repose sur trois étapes :

1. La conception de la commande LQ
2. L'analyse de l'erreur en régime permanent pour la corriger
3. La définition d'une distance d'anticipation pour éviter les oscillations

Par la suite, la commande résultante de ces étapes est nommée LQR-FF (pour Linear Quadratic Regulator with Feedforward).

3.2.1.4/ MÉTHODE DE LINDEROTH

La méthode de Linderoth [Linderoth et al., 2008] exprime l'angle $\delta(t)$ avec la relation 3.12.

$$\delta(t) = \tan^{-1} \left(\frac{-\cos(e_\theta(t))e^\perp(t) - (l_1 + l_2)\sin(e_\theta(t))}{l_1 - (l_1 + l_2)\cos(e_\theta(t)) + \sin(e_\theta(t))e^\perp(t)} \right) \quad (3.12)$$

Avec e_θ l'erreur d'angle, e^\perp l'erreur latérale, l_1 l'empattement du véhicule, l_2 une distance cible.

3.2.2/ CONTRIBUTION À LA COMMANDE LATÉRALE

Dans le cadre de la réalisation d'une intersection autonome avec positionnement satellite uniquement, la stratégie de contrôle latéral doit être robuste et supporter une certaine latence et une certaine imprécision quant à la position. Des tests en simulation nous ont montré que sous ces contraintes la plupart des solutions existantes ne donnaient pas des résultats satisfaisants.

Lors de ces tests, la stratégie ayant donné les meilleurs résultats fut la pure poursuite. Cela s'explique par une tolérance plus forte face aux perturbations et des prérequis moins forts quant à la précision et au temps de réaction du système. Néanmoins, la pure poursuite tend à couper les courbes par l'intérieur, et ce, d'autant plus que la vitesse augmente. Pour y remédier, la stratégie de suivi peut intégrer la courbure de la trajectoire dans l'équation. Une première approche a été proposée dans [Lombard et al., 2016a] puis nous l'avons raffinée avec 3.13.

$$\delta_t = \zeta \tan^{-1} \left(\frac{2E \cos(e_t - \tan^{-1}(\frac{d}{e_d}))}{\sqrt{e_d^2 + d^2}} \right) + \tan^{-1} \left(\frac{2E \sin^{-1} \frac{\kappa}{2}}{L} \right) \quad (3.13)$$

Nous avons nommé "Improved CF (Curvature Following)" la commande issue de l'équation 3.13. Les paramètres de 3.13 sont (voir figure 3.6) :

- e_t l'angle la direction du véhicule et la tangente de la trajectoire
- d la distance du point cible avec $d = 2\tau \times v(t)$, τ étant le temps de réaction du système (en pratique le temps entre deux acquisitions de position) et $v(t)$ la vitesse du véhicule
- e_d la distance du véhicule à la trajectoire
- κ la norme de la différence entre la tangente au point cible et la tangente à la trajectoire au point issu de la projection du véhicule sur la trajectoire (c'est-à-dire une courbure)

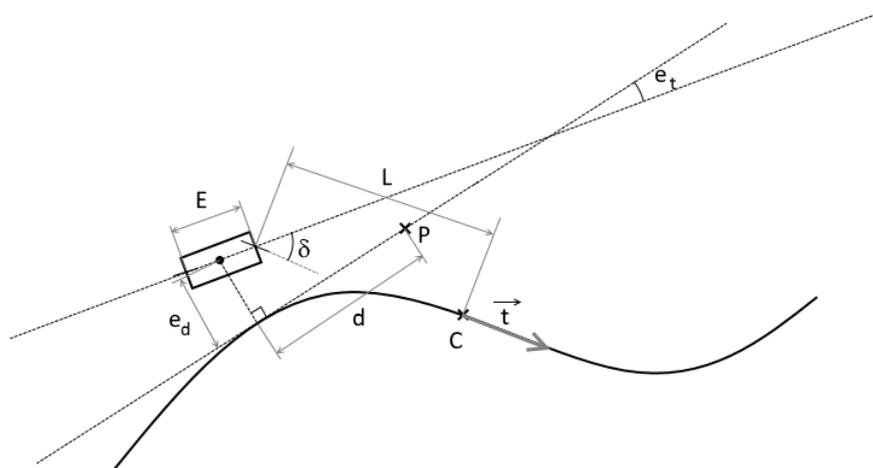


FIGURE 3.6 – Illustration des paramètres de 3.13

- L la distance (curviligne) prévisionnelle correspondant au suivi de courbure
- ζ vaut -1 lorsque le véhicule est à gauche de la trajectoire, 1 lorsqu'il est à droite

Cette équation est formée de l'addition de deux membres. D'un point de vue géométrique, le premier membre correspond à l'application de la pure poursuite à destination du point P. Le second membre correspond à l'application de la courbure de la trajectoire (prise en amont, pour tenir compte du temps de réaction du système).

3.2.3/ EXPÉRIMENTATIONS ET MESURES

3.2.3.1/ SIMULATIONS

Des simulations ont été réalisées, afin de pouvoir comparer la solution proposée aux stratégies existantes. Ces simulations consistent à faire suivre une trajectoire prédéfinie à un véhicule en utilisant les différentes stratégies précitées. Pour reproduire les contraintes du système réel deux perturbations sont introduites :

- Une imprécision sur la position du véhicule
- Une latence sur l'application de la commande calculée

Trois scénarios sont établis pour évaluer la performance du véhicule selon les conditions :

- *No latency* : Sans imprécision et avec une latence faible
- *Latency* : Avec latence et sans imprécision
- *Latency, error* : Avec latence et imprécision

De nombreux tests ont été réalisés, pour différents tracés de référence, pour différentes valeurs d'imprécision sur la position et pour différentes valeurs de latence. Par souci de concision, seuls les résultats correspondant au scénario approchant le plus les conditions réelles d'expérimentation de l'intersection autonome sont présentés.

Ainsi, sur le tracé défini par 3.7, nous avons obtenu les résultats donnés par les figures 3.8, 3.9 et 3.10, respectivement pour les scénarios 1, 2 et 3. Pour ces mesures, la latence (lorsque présente) était fixée à 400 ms, et l'imprécision du positionnement était fixée à 10cm pour la position (la position utilisée par le véhicule pour déterminer sa commande

était une valeur aléatoirement et uniformément choisie autour de la valeur réelle et dans un rayon de 10cm), et à ± 5 deg pour la direction (également uniformément autour de la valeur réelle).

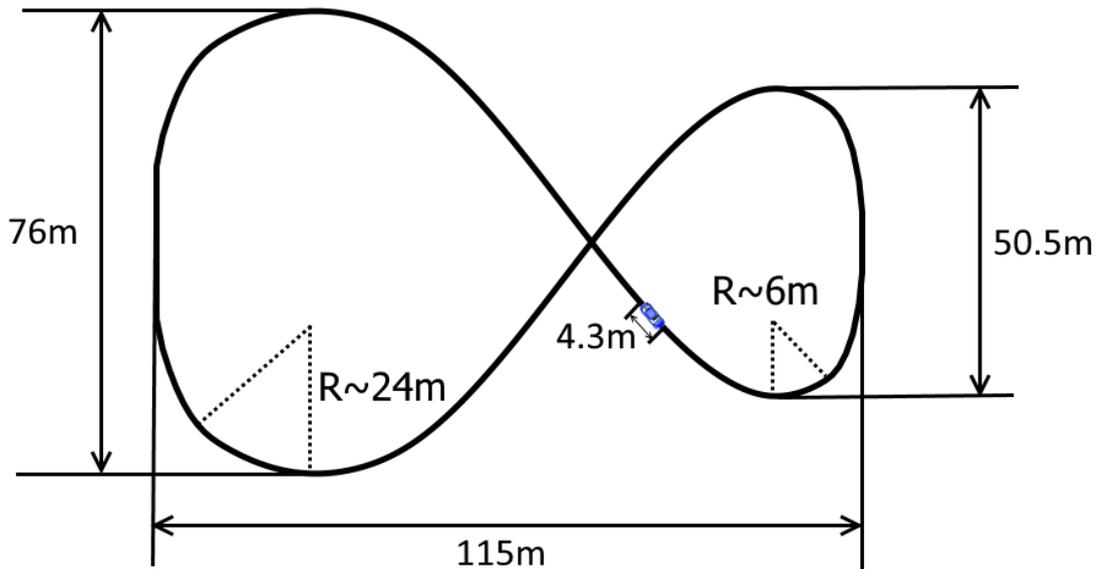


FIGURE 3.7 – Circuit utilisé en simulation pour les résultats présentés

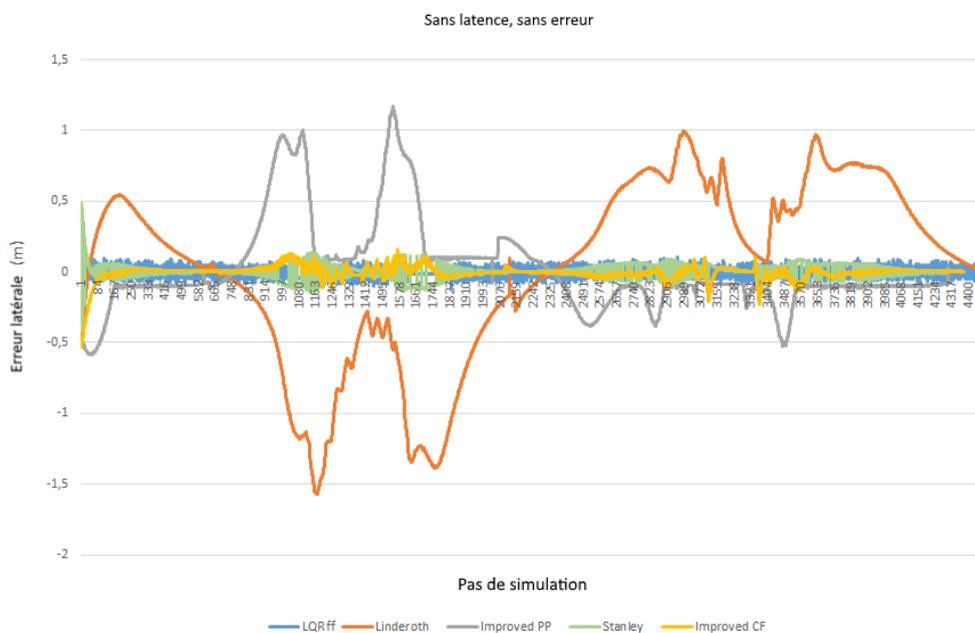


FIGURE 3.8 – Erreur latérale instantanée dans le cas d'une simulation sans imprécision de positionnement et sans latence

Les erreurs latérales moyennes données par les figures 3.11 et 3.12 montrent le gain sensible apporté par la solution proposée, notamment lorsqu'il y a latence et imprécision.

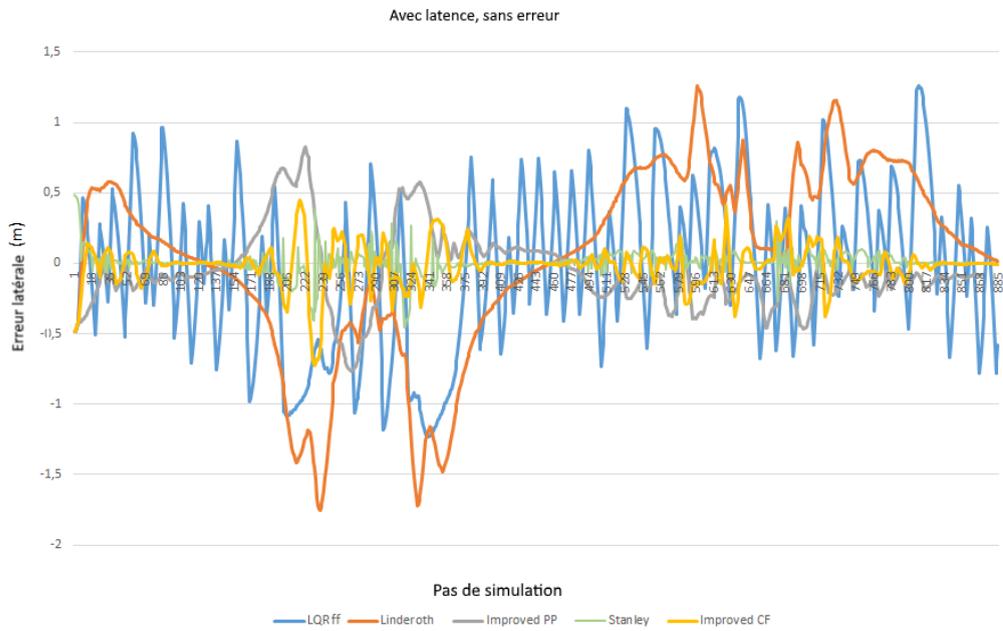


FIGURE 3.9 – Erreur latérale instantanée dans le cas d’une simulation sans imprécision de positionnement et avec latence sur l’application de la commande

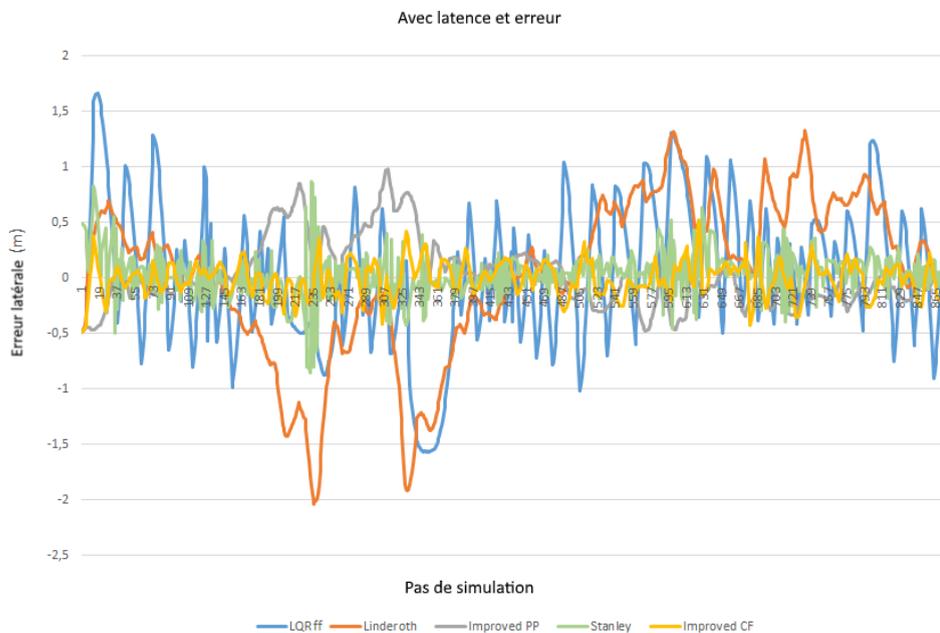


FIGURE 3.10 – Erreur latérale instantanée dans le cas d’une simulation avec imprécision de positionnement et avec latence sur l’application de la commande

	LQR	Linderoth	Improved PP	Improved CF	Stanley
Sans latence	3,06E-02	5,08E-01	2,13E-01	2,87E-02	4,01E-02
Avec latence	4,36E-01	4,89E-01	1,97E-01	8,67E-02	4,81E-02
Latence, erreur	4,79E-01	5,19E-01	2,02E-01	1,13E-01	1,47E-01

FIGURE 3.11 – Erreur latérale moyenne sur l'ensemble du tracé (exprimée en mètres)

	LQR	Linderoth	Improved PP	Improved CF	Stanley
Sans latence	-6,29%	-94,35%	-86,49%	0,00%	-28,42%
Avec latence	-80,11%	-82,27%	-56,04%	0,00%	80,18%
Latence, erreur	-76,43%	-78,21%	-44,16%	0,00%	-23,02%

FIGURE 3.12 – Erreur relative par rapport à la solution proposée

3.2.3.2/ TESTS RÉELS

Dans un second temps, la commande a été testée en conditions réelles sur un véhicule robotisé et équipé d'un GPS RTK (les détails concernant l'architecture matérielle et logicielle de ce véhicule sont donnés dans le chapitre 5).

La figure 3.13 montre le circuit de référence utilisé en rouge, et le circuit effectivement réalisé par le véhicule au cours de trois tours réalisés en utilisant la commande latérale autonome (seuls l'accélération et le freinage étaient pilotés manuellement). Le rayon de courbure minimal du tracé de référence est estimé à 8,2m d'après les données GPS.



FIGURE 3.13 – Tracé GPS du circuit de référence en rouge, et tracés GPS réalisés en utilisant le contrôle latéral autonome

La figure 3.14 indique l'erreur latérale (distance orthogonale au tracé de référence) mesurée en fonction du temps pour chacun des circuits. La vitesse étant pilotée manuellement, celle-ci varie tout au long du circuit et est comprise en moyenne entre 4,5m/s (dans les

virages) et $10m/s$ (dans les lignes droites).

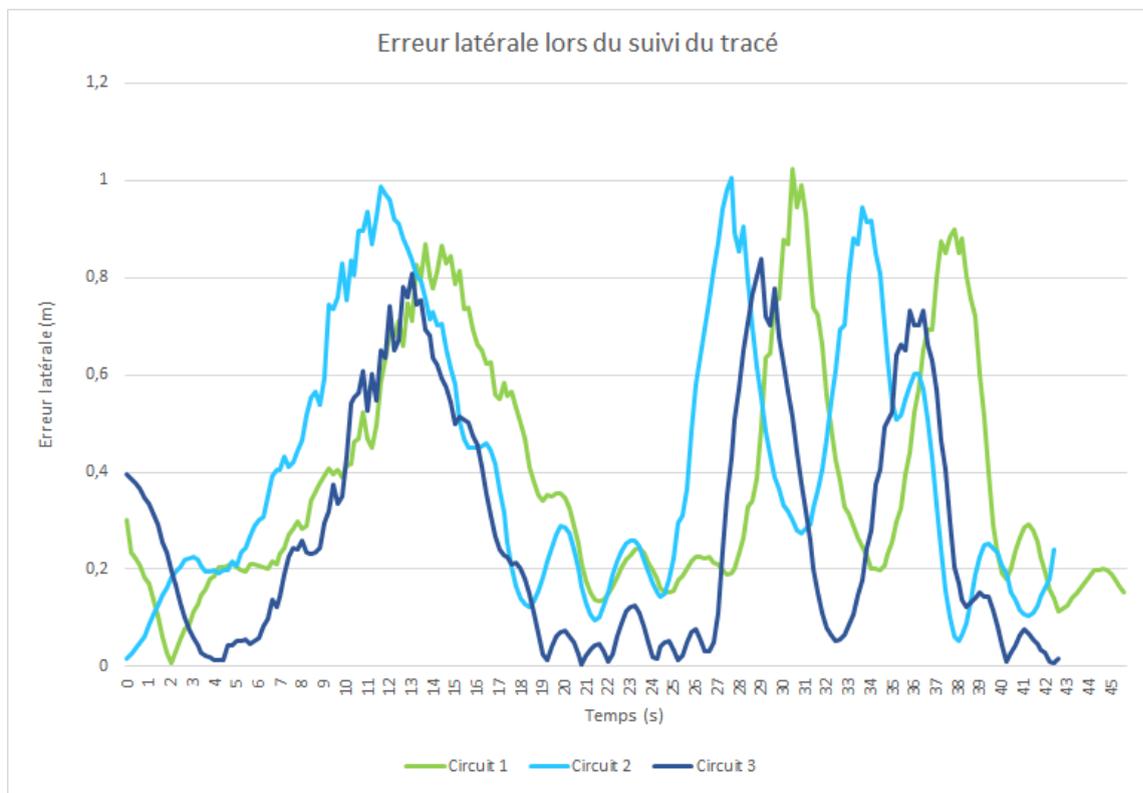


FIGURE 3.14 – Erreur latérale mesurée lors de l'expérimentation en conditions réelles de la commande latérale

Nous pouvons ainsi remarquer une erreur limitée à $1m$ se manifestant dans les virages, et une erreur limitée à $0,2m$ dans les lignes droites.

Les résultats sont globalement moins bons que ceux obtenus en simulation (pour lesquels l'erreur était limitée à $0,5m$). Cela s'explique notamment par un décalage de la commande du volant (la commande 0deg correspondait en réalité à un angle réel du volant d'environ 15deg). Ce problème a été observé lors des deux premiers tracés, puis compensé dans le dernier, ce qui a permis de borner l'erreur à $0,8m$ dans les virages et $0,1m$ dans les lignes droites. Pour le reste, l'erreur supplémentaire s'explique par une sous-estimation de l'erreur d'orientation du véhicule en simulation ($\pm 10\text{deg}$ en réel au lieu des $\pm 5\text{deg}$ considéré dans la simulation). Aussi, l'erreur GPS est mesurée avec elle-même une erreur, ce cumul induit une mesure d'erreur plus élevée que l'erreur réelle.

Dans l'ensemble l'erreur est contenue, et le comportement est relativement proche des estimations préalablement obtenues en simulation : erreur très faible dans les lignes droites, un peu plus importante dans les virages, et très peu d'oscillations.

3.2.4/ CONCLUSION (CONTRÔLE LATÉRAL)

De nombreuses solutions de contrôle latéral existent dans la littérature, avec leurs avantages et leurs inconvénients. Nous proposons ici une nouvelle stratégie de contrôle inspirée de la pure poursuite. L'objectif de cette commande est de limiter l'erreur de suivi de

trajectoire même en cas de défauts de positionnement ou d'application de la commande (latence et imprécision).

En ce sens, des tests en simulation sont réalisés pour comparer la solution proposée aux solutions existantes. Nous pouvons ainsi observer une nette réduction de l'erreur par rapport aux autres stratégies existantes, et ce plus notablement encore lorsqu'une imprécision dans le positionnement et une latence sur l'application de la commande existent. Ces résultats sont ensuite confirmés en conditions réelles en utilisant cette commande pour piloter un véhicule robotisé à partir du positionnement GPS RTK.

Enfin, un avantage supplémentaire de cette commande est sa rapidité d'implémentation : au contraire d'une stratégie basée sur une commande LQ ou une commande à modèle interne, il y a peu de paramétrage à réaliser. Les principaux paramètres étant τ et L dont la valeur est ajustable pour affiner le comportement du véhicule.

3.3/ ADAPTATION DE LA VITESSE PAR RAPPORT À LA TRAJECTOIRE

3.3.1/ PRINCIPE

Lorsque la courbure de la trajectoire augmente, il peut être requis de réduire la vitesse du véhicule afin de maintenir la condition de roulement sans glissement.

Dans ce cadre nous proposons une heuristique permettant de réduire la vitesse maximale du véhicule dans les virages et de l'augmenter dans les lignes droites. Pour ce faire, nous utilisons les commandes longitudinales 3.9 et latérales préalablement définies 3.13 dans le cadre de la méthode définie dans [Lombard et al., 2016a].

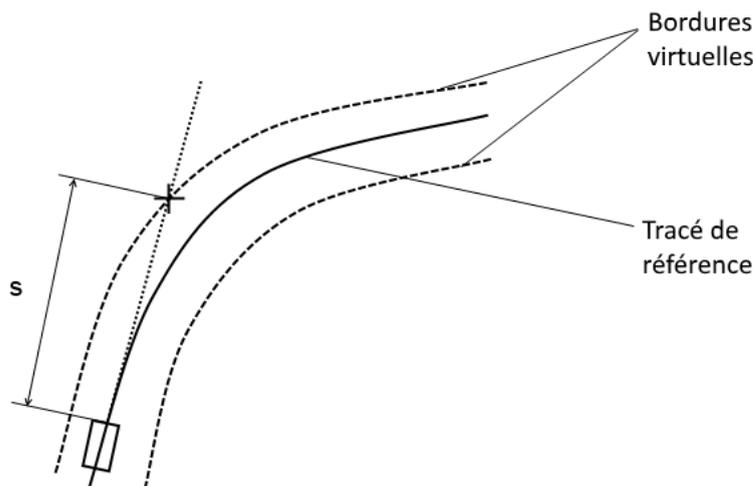


FIGURE 3.15 – Schéma représentant les « bordures » virtuelles

Dans cette méthode, nous définissons des « bordures » virtuelles autour de la trajectoire de référence (voir figure 3.15) dont l'entraxe est paramétrable, mais constant. Nous utilisons ensuite la fonction de contrôle de l'accélération en intégrant ces bordures comme

étant des obstacles. Ainsi, lorsqu'il se dirige vers une bordure (cas d'un virage) le véhicule réduit sa vitesse, et lorsqu'il est en ligne droite, il peut ré-augmenter sa vitesse.

Par exemple, en supposant que le véhicule est situé sur la trajectoire et dans la direction de la trajectoire, alors pour un virage de rayon de courbure constant R , et pour un entraxe des bordures virtuelles valant e , la relation 3.14 nous donne la distance s à l'obstacle virtuel.

$$s = \sqrt{\frac{e^2}{4} + eR} \quad (3.14)$$

Nous pouvons ensuite utiliser cette distance comme étant la distance à un obstacle dans une commande d'ACC, et ainsi estimer une vitesse optimale en fonction du rayon de courbure de la trajectoire. Dans ce contexte, nous définissons cette vitesse optimale comme étant celle pour laquelle la commande d'ACC donne une accélération ou un freinage nul.

De plus, un avantage secondaire de l'utilisation des bordures virtuelles, par rapport à une stratégie statique de définition de la vitesse optimale en fonction de la courbure, est qu'elle permet de réduire la vitesse du véhicule lorsque celui-ci s'éloigne de la trajectoire de référence, ou qu'il se dirige en dehors de celle-ci.

3.3.2/ EXEMPLE D'APPLICATION

Nous cherchons à déterminer la vitesse optimale dans le cadre de la stratégie d'adaptation de la vitesse en fonction de la courbure dans le cas de RT-ACC. Si nous résolvons l'équation $a_r = 0$ avec s donné par 3.14, alors la plus petite racine strictement positive du polynôme 3.15 (issu directement de l'équation 3.9 en posant $a_r = 0$) nous donne la vitesse optimale v_f .

$$4v_f^2 - 4b_f\tau v_f - b_f \frac{b_f b_l \tau^2 + 4b_f v_f \tau + 4v_l^2 - 8b_l s}{b_l} + b_f^2 \tau^2 = 0 \quad (3.15)$$

En posant $v_l = 0$, les racines de ce polynôme sont données par 3.16.

$$v_f = b_f \tau \pm \sqrt{-b_f(-b_f \tau^2 + 2s)} \quad (3.16)$$

Ainsi, par application numérique en utilisant les valeurs classiques $b_f = -2$, $\tau = 2$, $b_l = -8$ et la relation 3.14, les figures 3.16 et 3.17 nous donnent la relation entre vitesse et rayon de courbure respectivement pour $e = 5$ et $e = 10$.

3.3.3/ SIMULATION ET RÉSULTATS

Nous appliquons dans les conditions décrites par 3.2.3.1 la stratégie d'adaptation de la vitesse. Nous utilisons la stratégie de contrôle latérale définie par 3.13 avec la stratégie de contrôle longitudinale RT-ACC.

Dans le cadre de cette simulation, nous fixons $e = 5m$. Nous comparons ensuite l'erreur latérale et l'accélération normale sans et avec adaptation de la vitesse.

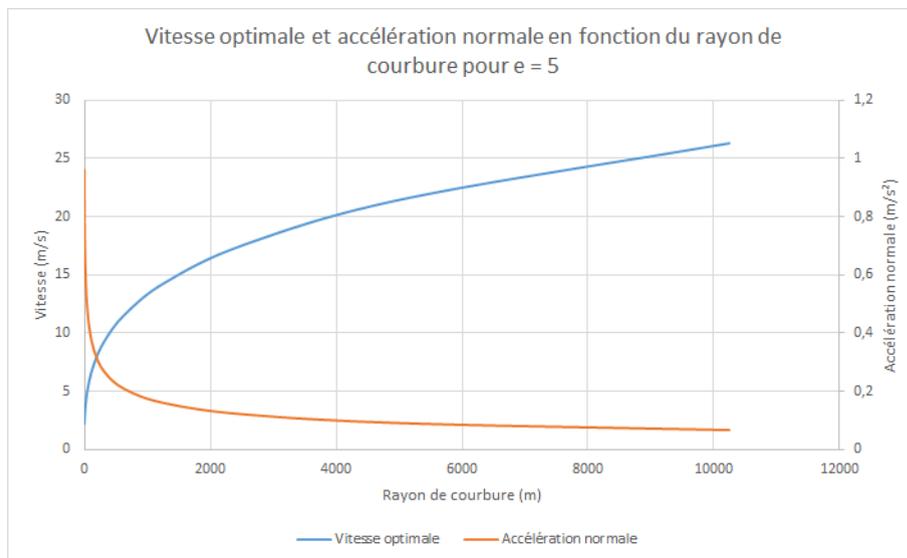


FIGURE 3.16 – Vitesse optimale et accélération normale associée en fonction du rayon de courbure pour $e = 5$ (virage constant)

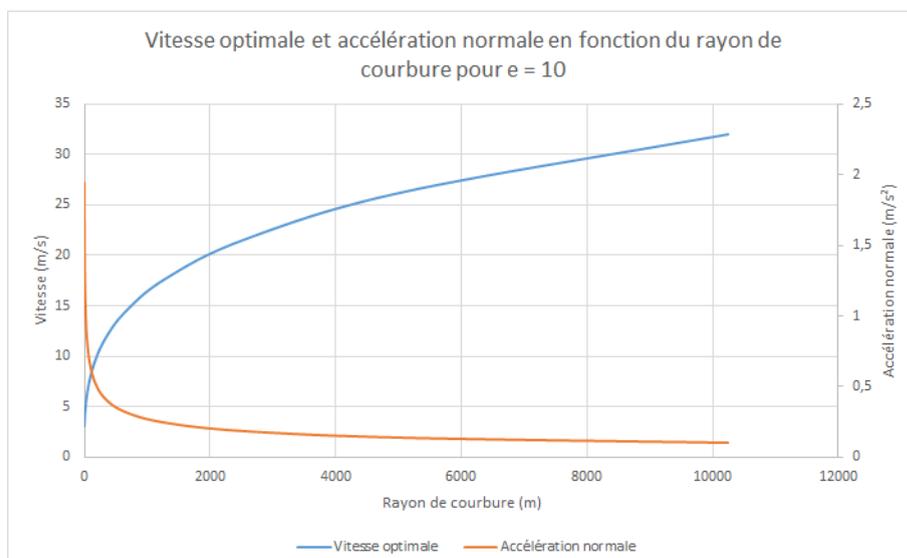


FIGURE 3.17 – Vitesse optimale et accélération normale associée en fonction du rayon de courbure pour $e = 10$ (virage constant)

La figure 3.18 nous montre des résultats similaires en termes d'erreur latérale, la mesure de l'erreur latérale moyenne donne 4,9% (ou 0,02m) d'écart en faveur de la simulation à vitesse constante. La différence d'erreur est faible et ne montre pas une perte significative des performances de la fonction de contrôle longitudinal, mais elle montre que cette stratégie d'adaptation de la vitesse ne peut amener un gain en termes d'erreur latérale.

Cependant, la figure 3.19 montre un net gain en termes d'accélération latérale avec une réduction de l'accélération latérale moyenne de 23,5% par l'utilisation de la stratégie d'adaptation de la vitesse. On peut également noter que les valeurs élevées d'accélération normale montre que la vitesse cible choisie n'est pas adaptée au circuit de simulation, mais la réduction de la valeur de e permet de palier à ce problème.

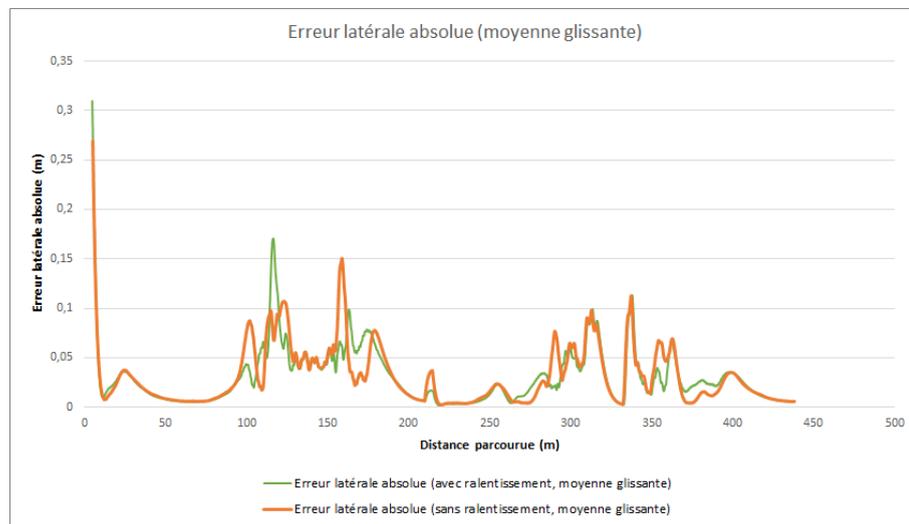


FIGURE 3.18 – Erreur latérale en valeur absolue (moyenne glissante) en fonction de la distance parcourue par le véhicule

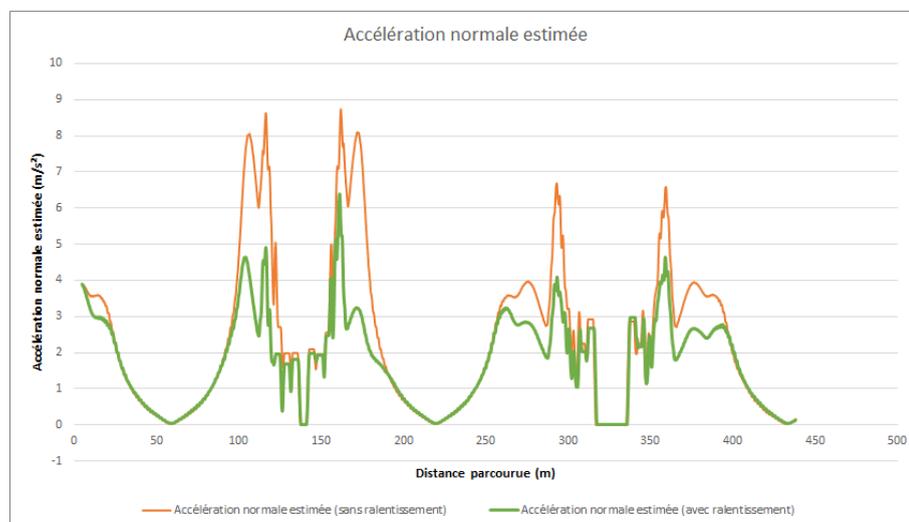


FIGURE 3.19 – Accélération normale (moyenne glissante) en fonction de la distance parcourue par le véhicule

3.4/ CONCLUSION

La réalisation d'une intersection autonome est conditionnée au développement du véhicule autonome. Ce dernier doit pouvoir déterminer la commande de direction à appliquer en fonction de sa position, et doit pouvoir synchroniser sa vitesse avec les véhicules qui le précèdent dans la séquence de passage. Toutefois, des contraintes supplémentaires s'appliquent, notamment lorsqu'il s'agit de synchroniser sa vitesse avec un véhicule dont la position et la vitesse sont connues avec un délai et avec une imprécision.

Ainsi, l'introduction de ce chapitre consiste à établir un bref état de l'art de différentes stratégies de contrôle, puis à évaluer celles qui conviennent le mieux compte tenu des contraintes spécifiques liées à la réalisation d'un prototype d'intersection autonome.

Dans ce cadre, nous analysons les différentes solutions existantes tant pour le contrôle longitudinal que pour le contrôle latéral. Concernant la commande longitudinale, il ressort que la commande RT-ACC répond aux besoins et aux contraintes de l'intersection autonome. Concernant la commande latérale, nous proposons une nouvelle stratégie de contrôle, et par la simulation nous vérifions que celle-ci se comporte mieux que les solutions existantes.

Enfin, nous proposons l'intégration des commandes latérales et longitudinales dans le cadre d'une stratégie d'adaptation de la vitesse en fonction de la courbure de la trajectoire. Puis, par analyse et simulation, nous montrons que cette technique permet de réduire et contrôler l'accélération normale du véhicule lors du suivi de trajectoire.

INTERSECTION AUTONOME ET INTER-BLOCAGE

Dans une intersection autonome, les véhicules arrivant à l'intersection négocient leur droit de passage en utilisant la communication sans-fil. Indépendamment du protocole de négociation utilisé ou de la stratégie de régulation associée, cette négociation conduit à l'établissement d'un ordre de priorité à l'intersection afin d'ordonner les véhicules et d'éviter les collisions entre véhicules ayant des mouvements conflictuels.

Chaque véhicule devant traverser l'intersection adapte donc sa vitesse de façon à franchir l'intersection après les véhicules qui ont la priorité sur lui. Ainsi un véhicule peut être amené à réguler sa vitesse par rapport à trois types d'« obstacles » différents :

- les obstacles réels, perçus par les capteurs de distance (radar, lidar), autrement dit les véhicules qui le précèdent dans la même voie
- les obstacles virtuels, renseignés par la communication sans fil, qui sont les véhicules prioritaires en provenance des voies conflictuelles
- la ligne d'arrêt, placée juste avant l'intersection et à laquelle doit systématiquement s'arrêter un véhicule n'ayant pas le droit de passage

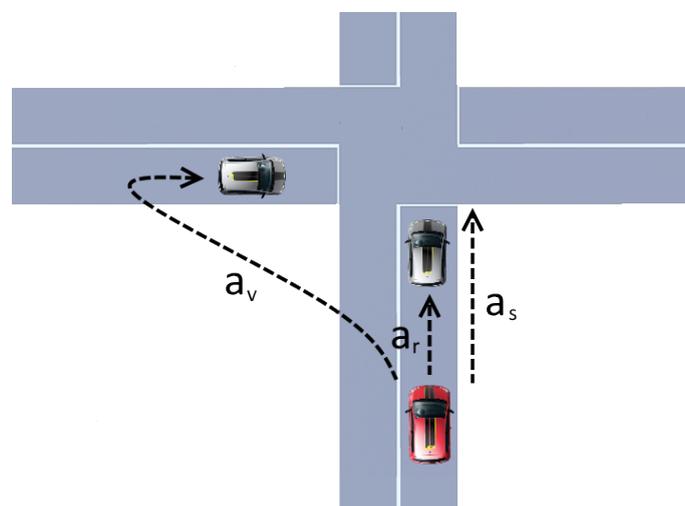


FIGURE 4.1 – Représentation des types obstacles pour un véhicule

La régulation de la vitesse relative à chacun de ces obstacles définit une consigne d'accélération spécifique : a_r relative aux obstacles réels, a_v relative aux obstacles virtuels

et a_s relative à la ligne d'arrêt si le véhicule n'a pas reçu l'autorisation de passer (voir 4.1). Par mesure de sécurité, nous bornons l'accélération effective du véhicule a par le minimum de ces trois consignes (voir équation 4.1).

$$a = \min(a_r, \min(a_s, a_v)) \quad (4.1)$$

Ainsi, chaque véhicule « suit » à la fois ses obstacles réels et ses obstacles virtuels, créant une relation de précedence entre les véhicules. De ce fait, une situation d'inter-blocage est possible si un véhicule adapte indirectement sa vitesse par rapport à lui-même (voir figure 4.2).

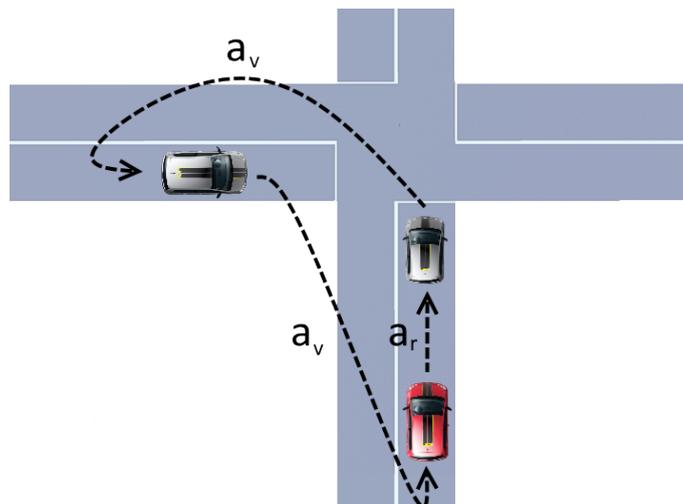


FIGURE 4.2 – Illustration d'un cas d'inter-blocage : un véhicule est indirectement son propre obstacle

Une telle situation d'inter-blocage est critique en ce qu'elle paralyse entièrement l'intersection. Il est donc nécessaire que les politiques d'ordonnancement des véhicules dans les intersections coopératives considèrent ce problème afin de le prévenir.

Dans ce chapitre, nous proposons une étude de ce problème et une contribution à sa résolution sous la forme d'un algorithme s'intégrant dans la politique d'ordonnancement. Nous commençons par décrire précisément les causes aboutissant à la création d'une situation d'inter-blocage, notamment via l'utilisation de la communication sans fil, puis montrons que les conditions d'apparition d'un inter-blocage sont réunies. Enfin, nous détaillons l'algorithme permettant de le résoudre.

4.1/ COMMUNICATION SANS-FIL ET ORIGINES DE L'INTER-BLOCAGE

Afin de concevoir une intersection coopérative de véhicules autonomes, nous commençons par réaliser un prototype en simulation en utilisant le cadre préalablement défini. Pour valider les aspects liés à la communication de l'intersection coopérative, nous implémentons un composant logiciel simulant la communication sans-fil entre véhicules, et l'utilisons dans le cadre d'un protocole de régulation coopérative existant.

Le protocole standard de la communication inter-véhiculaire est le standard IEEE 802.11p. Bien que ce protocole soit conçu pour limiter les pertes de messages, celles-ci ne peuvent être évitées [Demmel et al., 2012]. De plus, la présence d'obstacles ou d'interférences peut augmenter le risque de pertes de messages : l'augmentation du nombre d'unités de communication ITS aggrave ces problèmes [Eichler, 2007]. Via des tests réels avec des unités de communication IEEE 802.11p Commsignia, nous avons pu vérifier empiriquement ces problèmes.

Ainsi, pour reproduire au mieux le fonctionnement réel de la communication inter-véhiculaire, la simulation doit reproduire la latence et les pertes de messages observées en utilisant les protocoles standards.

Concernant la stratégie de régulation reposant sur cette communication sans fil, plusieurs protocoles sont disponibles (voir chapitre 2). Nous choisissons d'utiliser le protocole TIM [Perronnet et al., 2013b] en raison des bénéfices préalablement présentés. La politique de régulation que nous utilisons en association est la politique FCFS, efficace en cas de trafic faible à modéré.

4.1.1/ MODÉLISATION DE LA COMMUNICATION SANS-FIL : LATENCE ET PERTES

Par la simulation de la communication sans-fil avec latence et pertes de messages, notre objectif est de valider le principe de fonctionnement de l'intersection coopérative, même avec une communication instable. C'est pourquoi nous choisissons de modéliser les perturbations de la communication sans-fil de manière très dégradées (et non nécessairement réalistes). Si l'intersection autonome fonctionne en simulation avec une forte latence et des pertes de messages importantes, alors nous pouvons supposer que ces défauts ne poseront pas de problèmes en tests réels sous des conditions plus favorables (bien qu'une validation expérimentale reste nécessaire). A l'inverse, la simulation permettra de mettre en avant des éventuels défauts de conception à régler avant les tests réels.

Dans notre simulation multi-agent, la couche communication est ainsi simulée : lors qu'un agent envoie un message à destination d'un autre (*unicast*) ou plusieurs autres (*broadcast*), ce message est envoyé à un agent intermédiaire chargé de sa distribution. Cet agent possède une certaine probabilité $p_{reception}$ de remettre les messages à leurs destinataires : une valeur de 1 indique que tous les messages sont reçus par chaque destinataire, une valeur de 0.5 indique que chaque agent n'a que 50% de recevoir le message dont il est destinataire, et une probabilité de 0% occulte tous les messages). Chaque paire (*message, destinataire*) se voit également attribuer un temps $t_{latence}$ généré selon une distribution uniforme dans un intervalle $[t_{min}; t_{max}]$: le *destinataire* de *message* ne recevra le message que $t_{latence}$ secondes après son envoi. Cette latence reproduit le délai nécessaire pour qu'un message passe de la couche application de l'émetteur à la couche application du destinataire. Cette communication est illustrée par la figure 4.3.

De cette façon, les messages n'ont pas de garantie d'être livrés et leur ordre d'arrivée n'est pas non plus garanti. Cela simule le fonctionnement d'une communication réseau en mode non-connecté.

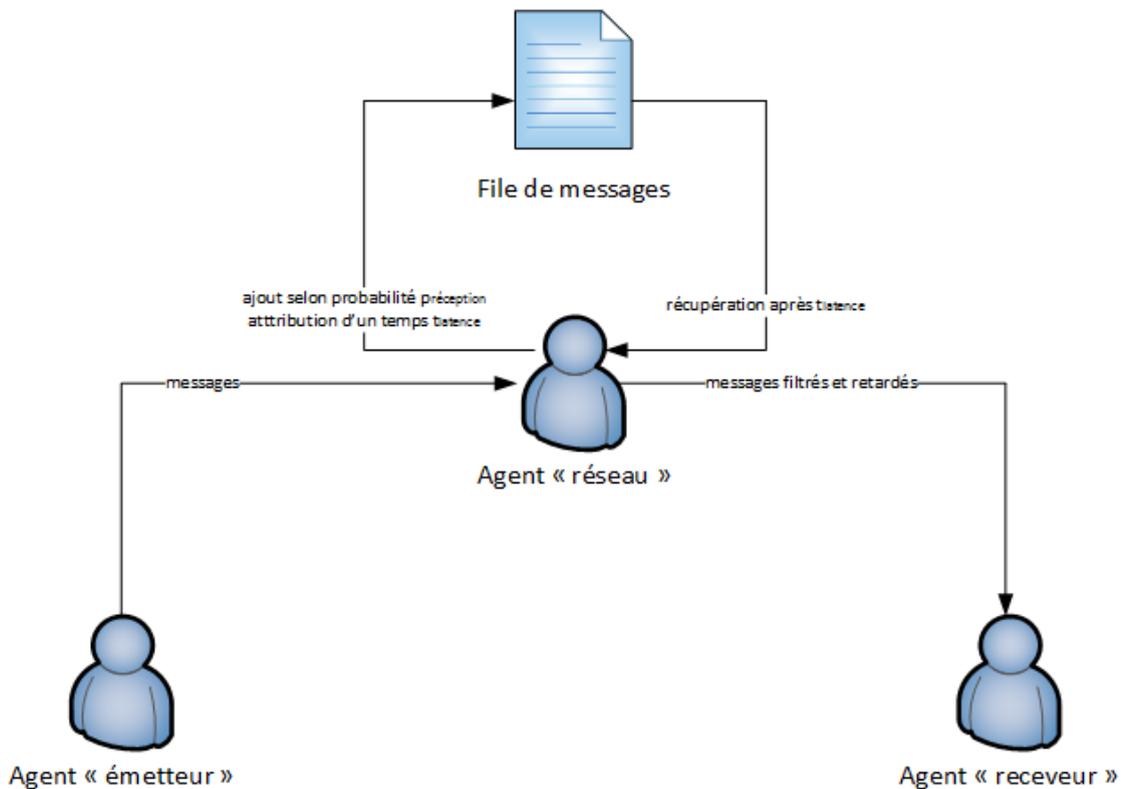


FIGURE 4.3 – Représentation schématique des agents intervenants dans la distribution des messages dans le cadre de la simulation de la communication sans-fil

4.1.2/ INTER-BLOCAGE D'UNE INTERSECTION AUTONOME

L'inter-blocage est un phénomène se produisant lorsque deux processus ou agents s'attendent mutuellement. Les processus impliqués restent bloqués indéfiniment. D'après [Coffman et al., 1971], quatre conditions sont requises pour conduire à un inter-blocage :

1. Les processus réclament l'accès exclusif à une ressource (condition d'exclusion mutuelle)
2. Les processus ont déjà l'accès exclusif à une ou des ressources et attendent des ressources supplémentaires (condition d'attente)
3. Les ressources ne peuvent pas être libérées de manière forcée par les processus qui les détiennent (condition de non-préemption)
4. Il existe une relation de dépendance circulaire entre les processus (condition d'attente circulaire)

Dans le cas de la régulation autonome d'une intersection, nous définissons chaque véhicule comme étant un processus, et un segment de route comme étant une ressource. Chaque segment de route ne peut être occupé que par un seul véhicule simultanément (exclusion mutuelle). Pour accéder au segment de route suivant, celui-ci doit être libre (condition d'attente). Un véhicule ne peut pas libérer la ressource retenue tant qu'il n'a pas accès à la ressource qu'il attend (condition de non-préemption). Nous définissons qu'un segment de route est réservé par un véhicule lorsqu'il se trouve dessus. Lorsque les véhicules circulent dans une même voie, les segments de route sont réservés dans

l'ordre des véhicules, il n'y a donc pas inter-blocage [Havender, 1968]. A l'intersection, en accord avec le protocole TIM, une réservation du segment de route constituant la zone de conflit est faite via la communication sans fil, cette réservation prend la forme d'un droit de passage, attribué aux véhicules selon l'ordre des demandes (d'après la politique de régulation FCFS).

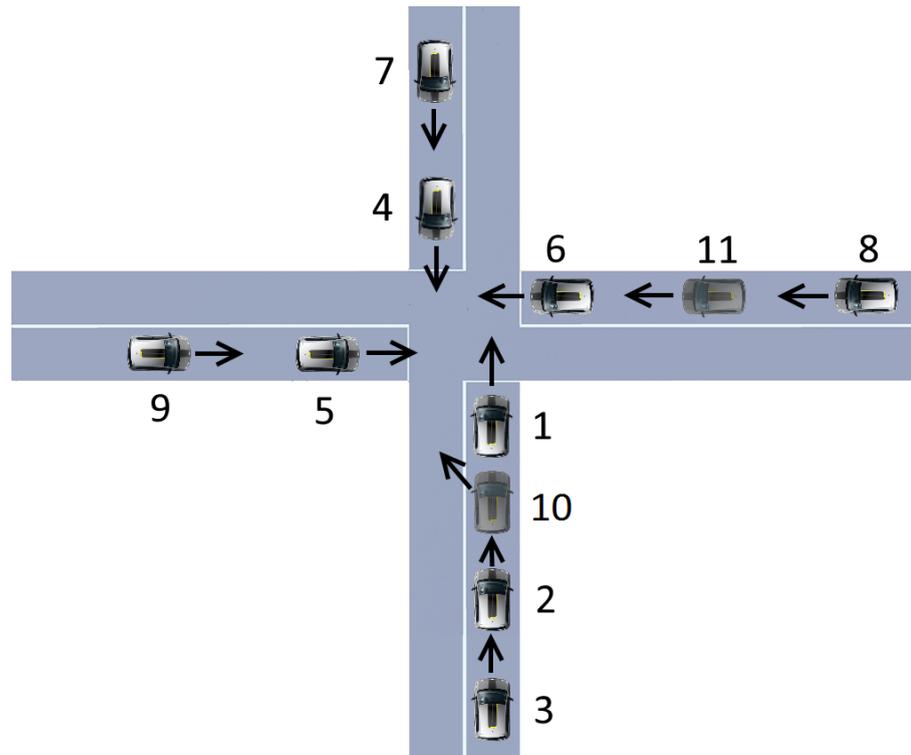


FIGURE 4.4 – Construction de la séquence de passage : l'ordre de passage dans la séquence (indiqué auprès de chaque véhicule) diffère de l'ordre réel des véhicules

Par la simulation un défaut conduisant à un cas d'inter-blocage a été mis en évidence. En effet, par l'utilisation de la communication sans-fil, il n'y a pas de garantie que l'ordre de réservation de la zone de conflit corresponde à l'ordre de réservation des segments de route (voir figure 4.4). Cela peut notamment se produire dans le cas où un véhicule demande le droit de passage (et l'obtient) avant le véhicule qui le précède. Ainsi, l'inversion entre l'ordre réel des véhicules dans leur voie et leur ordre de passage au sein de l'intersection constitue une condition d'apparition de l'inter-blocage (création de la condition d'attente circulaire), indépendamment du protocole de régulation utilisé.

Par exemple, dans la figure 4.4, nous avons $10 > 9 > 7 > (6, 5) > (3, 4) > 2 > 10$ (avec $i > j$ signifiant que i attend sur j pour franchir l'intersection).

Dans nos simulations, trois cas ont amené à l'apparition de cette situation :

- La requête émise par le second véhicule a été reçue par le contrôleur avant la requête émise par le premier véhicule du fait de la latence de la communication
- La requête émise par le premier véhicule de la voie a été perdue, la requête émise par le véhicule suiveur a été reçue par le contrôleur d'intersection avant que le premier véhicule ne réémette sa requête
- L'imprécision du positionnement a amené le second véhicule à émettre sa requête avant le premier véhicule (un véhicule émet une requête de positionnement dès

qu'il se trouve dans la zone d'entrée de l'intersection)

4.2/ RÉSOLUTION DE L'INTER-BLOPAGE

Dès lors que la situation d'inter-blocage apparaît, il est nécessaire de la résoudre. La résolution de l'inter-blocage est un problème largement étudié et de nombreuses solutions existent. Celles-ci comptent parmi elles la définition d'un ordre de réservation des ressources [Havender, 1968] et l'algorithme du banquier [Dijkstra, 1982]. Néanmoins, ces algorithmes ne s'appliquent pas ici car l'inter-blocage résulte d'un manque ou d'une erreur d'information concernant les processus existants et les ressources actuellement réservées. Un algorithme spécifique doit donc être conçu et mis en place.

Les trois premières conditions d'apparition de l'inter-blocage étant intrinsèques au trafic, nous ne pouvons agir que sur la dernière (attente circulaire). Pour supprimer le cycle d'inter-blocage il faut modifier l'ordre d'accès à la zone de conflit (l'ordre des véhicules dans la voie ne pouvant être changé), ce qui revient à supprimer le droit de passage d'un véhicule. Comme discuté dans le chapitre 2, supprimer le droit de passage d'un véhicule engendre un risque de collision. Par exemple, dans la figure 4.5, la priorité d'un véhicule mal positionné dans la séquence est modifiée : il est placé en position 0 pour supprimer l'attente sur le véhicule 1, l'inter-blocage est ainsi prévenu, mais le véhicule en position 2 (qui préalablement avait le droit de passage, car n'étant pas en conflit avec un véhicule prioritaire) peut ne pas recevoir l'information à temps, ce qui conduit à une situation de collision possible.

Si nous reprenons la situation décrite par la figure 4.4, les véhicules 1 et 4 ont initialement le droit de passage du fait qu'ils ne possèdent pas de véhicules conflictuels prioritaires dans la séquence de passage. Pour résoudre l'inter-blocage, la séquence peut être réorganisée de manière à insérer le véhicule 10 en accord avec sa position physique (c'est-à-dire derrière 1 dans la séquence) lorsqu'il est découvert (i.e. lorsque le contrôleur reçoit sa requête de passage). Néanmoins une telle stratégie crée un risque de collision en 4 et 10 : du fait de sa nouvelle position dans la séquence, 10 devient prioritaire sur 4, or il n'y a aucune garantie que 4 reçoive la nouvelle séquence, donc 4 et 10 peuvent considérer avoir le droit de passage au même moment pour un même segment, créant un risque de collision. L'algorithme de résolution de l'inter-blocage doit donc intégrer cette contrainte.

De plus, la prise en compte du positionnement des véhicules lors de leur requête ne permet de résoudre simplement le problème, car du fait du déplacement continu du véhicule la position reçue par le contrôleur ne correspond plus à la position physique du véhicule au moment où le message est traité (voir figure 4.6). Néanmoins si le véhicule est fixe (comme dans le cas d'inter-blocage) et qu'il réémet régulièrement sa position (en accord avec le protocole TIM), nous pouvons faire l'hypothèse que le contrôleur finira par recevoir la position réelle du véhicule. Pour prendre en compte cette modification de la position, la séquence doit être réadaptée en continu, c'est-à-dire à chaque réception d'un nouveau message.

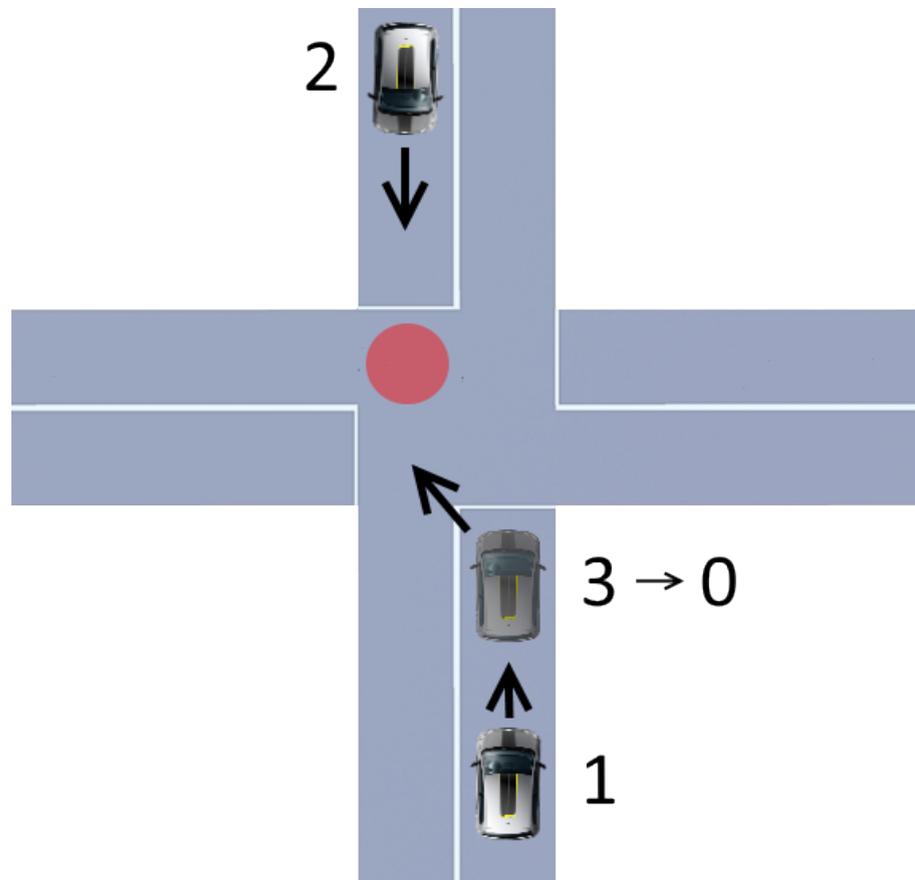


FIGURE 4.5 – Tentative de résolution de l'inter-blocage en changeant la priorité du véhicule mal situé dans la séquence résultant en une possible collision

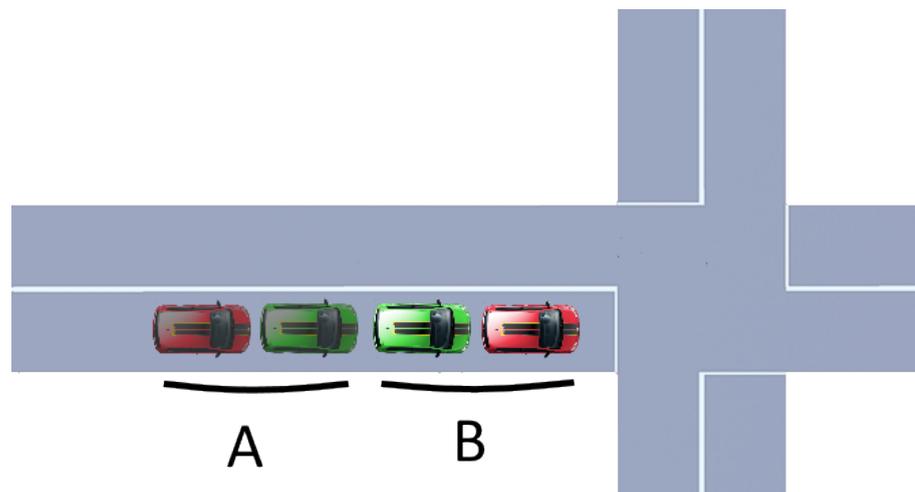


FIGURE 4.6 – A : position des véhicules au moment où ils ont émis leur requête d'accès à l'intersection ; B : position réelle des véhicules

4.2.1/ ALGORITHME COLLISION-DEADLOCK-FREE (CDF)

Pour résoudre la situation d'inter-blocage, nous proposons un algorithme nommé Collision and Deadlock Free (CDF) [Lombard et al., 2016b] (voir algorithme 1).

Algorithm 1 Algorithme Collision and Deadlock Free (CDF)

```

function SIZEOF(s) ▷ taille de l'ensemble s

function L(v) ▷ voie du véhicule v

function D(v) ▷ distance du véhicule v à l'intersection

function S(v) ▷ position du véhicule v dans la séquence

function CONFLICT(v1, v2) ▷ vrai si v1 et v2 sont en conflit, faux sinon

procedure MOVE(s, i, j) ▷ déplace l'élément à la position i dans la séquence s vers la position j

procedure INSERT(s, i, o) ▷ insère l'objet o à la position i dans s
procedure INSERTIONSEQUENCE(v)
  expectedPosition ← sizeof(sequence)
  for all v' in sequence do
    if  $l(v) = l(v')$  and  $d(v) < d(v')$  then
      expectedPosition ←  $\min(\text{expectedPosition}, s(v'))$ 
    end if
  end for
  insertionPosition ← expectedPosition
  if  $\text{expectedPosition} \neq \text{sizeof}(\text{sequence})$  then
    for i from expectedPosition to  $\text{sizeof}(\text{sequence}) - 1$  do
      if  $\text{conflict}(v, \text{sequence}[i])$  then
        insertionPosition ← i
      end if
    end for
    moved, i ← 0
    while  $i < \text{insertionPosition} - \text{moved}$  do
      if  $l(v) = l(\text{sequence}[i])$  then
        MOVE(sequence, i, insertionPosition)
        moved ← moved + 1
      else
        i ← i + 1
      end if
    end while
    if moved > 0 then
      insertionPosition ←  $\text{insertionPosition} - \text{moved} + 1$ 
    end if
  end if
  INSERT(sequence, insertionPosition, v)
end procedure

```

Cet algorithme fonctionne sur la base des hypothèses suivantes :

- Les positions des véhicules sont transmises régulièrement au contrôleur
- L'erreur de positionnement des véhicules est inférieure à 1 m, de sorte que les positions mesurées de deux véhicules fixes dans une même voie ne permettent

- pas de se tromper sur leur ordre réel
- Les véhicules ne reculent pas, les positions reçues par le contrôleur pour un véhicule sont toujours « croissantes »
- La position d'un véhicule reçue par le contrôleur d'intersection peut correspondre à une ancienne position du véhicule, mais pas à une position future : la position réelle du véhicule peut être en avant de la position reçue par le contrôleur mais jamais en arrière

De ce fait, le contrôleur finira toujours par connaître l'ordre réel des véhicules dans une voie.

Le principe de l'algorithme est d'insérer les véhicules dans la séquence de passage de façon à ce qu'ils ne se situent pas après un véhicule dont ils bloquent physiquement l'accès (ce qui créerait un cycle), ni avant un véhicule ayant un mouvement conflictuel (ce qui créerait une collision potentielle).

Ainsi, l'algorithme peut se résumer en deux étapes :

1. Le véhicule v à insérer est placé après le dernier véhicule conflictuel
2. Chaque véhicule physiquement placé derrière ce véhicule v est déplacé derrière dans la séquence

L'algorithme CDF est appelé à chaque réception d'un message. Les données relatives à tous les véhicules impliqués sont régulièrement mises à jour au travers de la requête d'actualisation des données du protocole TIM. Cette opération est effectuée en dehors de l'algorithme présenté.

Nous démontrons ensuite que cet algorithme permet de respecter la contrainte d'évitement de collision, tout en permettant de résoudre l'inter-blocage.

4.2.2/ ABSENCE DE COLLISION

Nous considérons qu'un risque de collision existe dès que deux véhicules en mouvement conflictuel obtiennent le droit de passage simultanément pour une zone de conflit donnée. Nous voulons démontrer que l'algorithme CDF n'induit pas un risque de collision.

Démonstration. L'attribution d'une priorité (ou d'un ordre passage) pour une zone de conflit donnée permet d'éviter le risque de collision [Qian et al., 2014]. Ainsi, tant que les priorités initialement attribuées, via la séquence, sont respectées, il ne peut y avoir de collision.

A l'inverse, si un véhicule gagne la priorité sur un autre, il peut y avoir collision.

Nous définissons pour toutes séquences possibles les relations d'ordre strict suivantes :

- Ordre réel : $\forall (v_i, v_j) \in V, v_i <_F v_j$ si et seulement si v_i précède v_j dans la même voie (relation non totale)
- Ordre virtuel : $\forall (v_i, v_j) \in V, v_i <_f v_j$ si et seulement si v_i précède v_j dans la séquence (relation totale)
- Ordre effectif : $\forall (v_i, v_j) \in V, v_i < v_j$ si et seulement si $v_i <_F v_j \vee (v_i <_f v_j \wedge \neg \exists v \in V/v <_F v_i \wedge v_j <_f v)$

La relation d'ordre $<_F$ désigne l'ordre réel des véhicules qui ne peut être modifié. La relation d'ordre $<_f$ désigne l'ordre virtuel des véhicules, c'est-à-dire leur ordre dans la séquence. Il peut être modifié.

Enfin, la relation d'ordre $<$ sur V exprime l'ordre de passage des véhicules pour une zone de conflit : un véhicule franchit nécessairement l'intersection après le véhicule qui le précède dans sa voie (d'où $v_i <_F v_j \Rightarrow v_i < v_j$), et un véhicule v_j franchit l'intersection après le véhicule v_i qui le précède dans la séquence sauf s'il existe un véhicule v tel que v_j est devant v dans la séquence et v est physiquement devant v_i (auquel cas nous avons $v_j < v_i$).

Soit un véhicule x à insérer dans la séquence par l'algorithme CDF. Par construction, nous avons $x <_f v \Leftrightarrow x <_F v$. De plus, toujours par construction, si $v_i <_f v_j$ avant CDF et si $v_j <_f v_i$ après CDF, cela signifie $x <_F v_i$ (avant et après CDF), donc $x < v_i$ d'où $v_j < v_i$ car $\exists v \in V/v <_F v_i \wedge v_j <_f v$. Ainsi, étant donné que l'ordre effectif de passage des véhicules n'est pas modifié par l'application de l'algorithme CDF, aucun risque de collision n'est introduit.

□

4.2.3/ RÉOLUTION DE L'INTER-BLOPAGE

Nous voulons démontrer que l'application de l'algorithme CDF permet effectivement de résoudre le problème de l'inter-blocage tel que précédemment décrit.

Démonstration. Comme énoncé précédemment, afin de remédier à la situation d'inter-blocage il faut supprimer la condition d'attente circulaire. Cette condition d'attente circulaire peut être représentée par un cycle dans le graphe des dépendances. Pour supprimer ce cycle, des arêtes (ou relations de dépendances) doivent être supprimées. Néanmoins, seules des relations de dépendances virtuelles peuvent être modifiées.

Soit $G = (N, E)$ un graphe dirigé dans lequel les noeuds $N = 1, 2, \dots, N \cup S$ représentent les véhicules et la contrainte d'arrêt, et l'ensemble d'arêtes dirigées E connecte chaque noeud v au noeud v' si le véhicule v doit réguler sa vitesse par rapport à l'obstacle v' . On a $E = E_r \cup E_i \cup E_s$ où E_r est l'ensemble des arêtes représentant un relation d'obstacle réel, E_i pour les obstacles virtuels et E_s pour la contrainte d'arrêt. La figure 4.7 montre le graphe avant et après la découverte du véhicule 10. Par souci de lisibilité, les arêtes des obstacles réels sont supprimées quand l'ordre est assuré par les arêtes des obstacles virtuels. Dans le schéma *b* de la figure 4.7, quatre cycles sont présents (voir figure 4.8). Ces cycles peuvent être brisés en supprimant deux arêtes de E_i . Quatre solutions sont possibles :

- Supprimer $10 > 9$ et $10 > 8$
- Supprimer $9 > 7$ et $8 > 7$
- Supprimer $7 > 5$ et $7 > 6$
- Supprimer $5 > 3$ et $6 > 3$

En accord avec l'algorithme CDF, les arrêtes $5 > 3$ et $6 > 3$ sont supprimées. La nouvelle séquence de véhicules quand 10 est découvert est $1 > 4 > 5 > 6 > 7 > 8 > 9 > 10 > 2 > 3$.

La condition d'attente circulaire est présente si les deux conditions suivantes sont réunies :

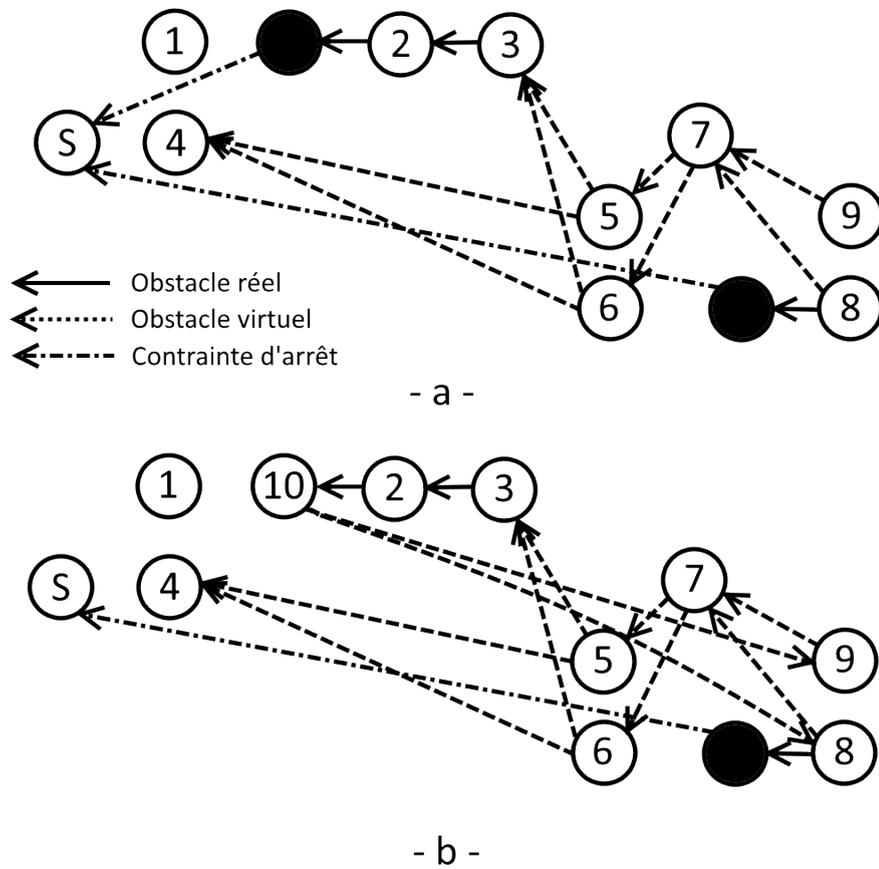


FIGURE 4.7 – Représentation sous forme de graphe de l'intersection : a- avant réception du message du véhicule 10 ; b- après réception du message du véhicule 10

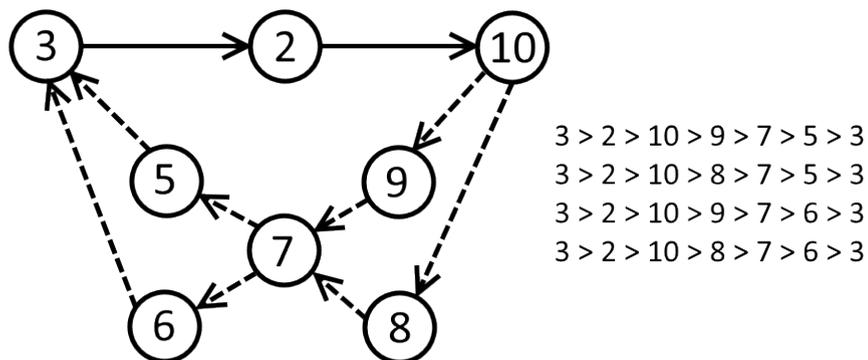


FIGURE 4.8 – Les quatre cycles du graphe de la figure 4.7

- L'ensemble S des véhicules physiquement derrière le nouveau véhicule n'est pas vide
 - Il y a au moins un véhicule dans S qui précède dans la séquence un véhicule en conflit avec le véhicule nouvellement découvert
- D'après CDF, étant donné que tous les véhicules physiquement derrière le véhicule nou-

vement découvert sont placés après celui-ci, le lien qui existe avec le véhicule conflictuel est supprimé, donc le cycle est supprimé et l'inter-blocage est résolu.

□

4.3/ TESTS ET RÉSULTATS

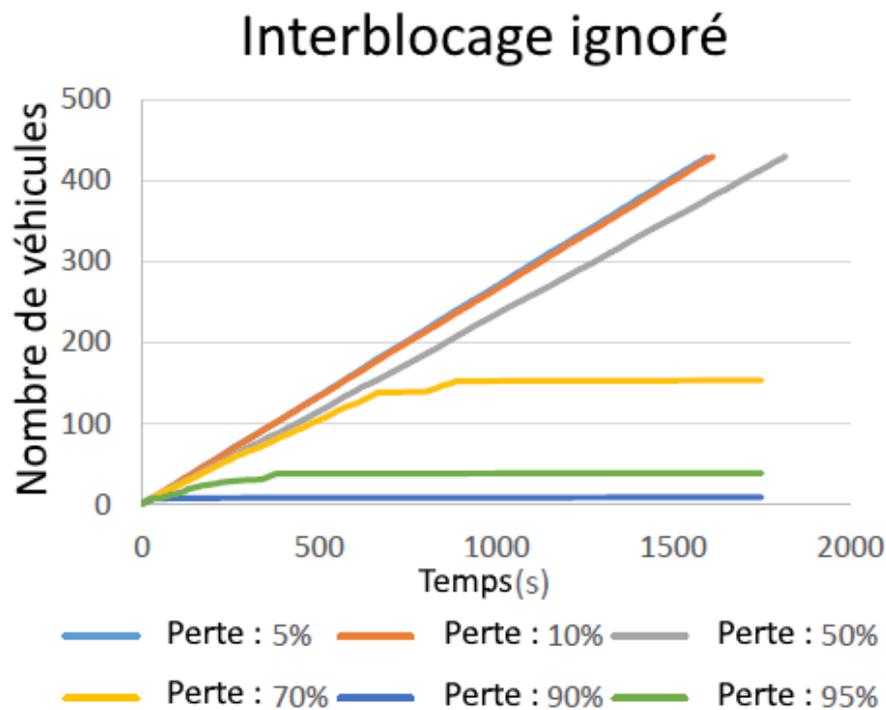


FIGURE 4.9 – Évacuation des véhicules sans stratégie anti-inter-blocage

Afin de valider expérimentalement les propriétés théoriques de l'algorithme CDF, les simulations d'intersection autonome ont été ré-effectuées selon les conditions précédemment définies, mais en utilisant l'algorithme CDF pour la construction de la séquence.

Les figures 4.9 et 4.10 montrent les résultats sans et avec application de l'algorithme anti-inter-blocage pour différentes probabilités de pertes de messages. La figure 4.9 montre que les véhicules cessent d'être évacués lorsque la probabilité de perte de message est élevée, ce qui correspond à l'apparition d'un inter-blocage. Cette situation n'apparaît pas dans la figure 4.10 où la stratégie anti-inter-blocage est appliquée.

Des tests ont également été réalisés avec des robots Lego EV3 (voir figure 4.11). Les robots EV3 communiquent entre eux en utilisant le protocole UDP en WiFi. Par l'expérience nous avons pu constater des défauts de communication importants avec les robots (latence allant jusqu'à plusieurs secondes en cas de perte de connexion avec le point d'accès). Néanmoins, aucune collision ni aucun inter-blocage ne s'est produit.

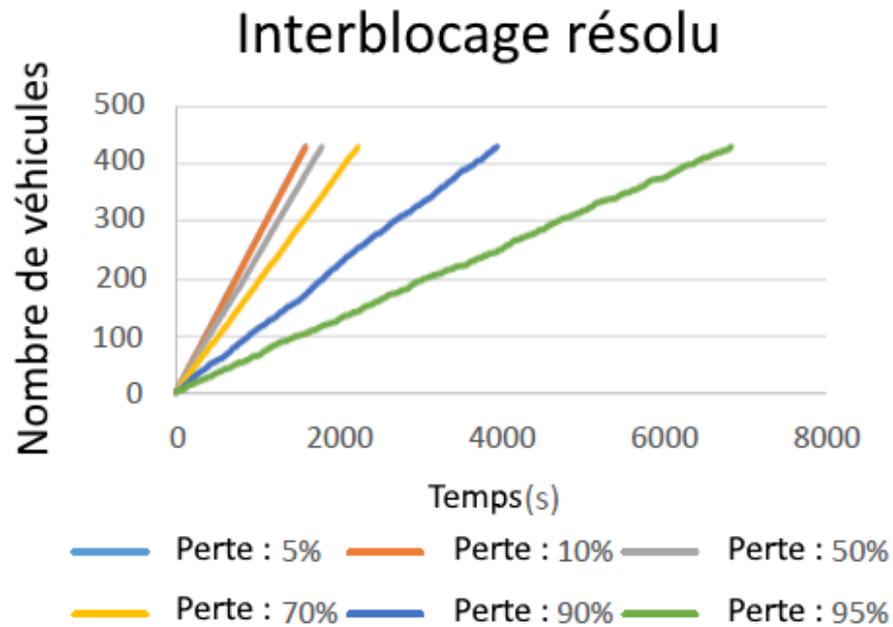


FIGURE 4.10 – Évacuation des véhicules avec stratégie anti-inter-blocage

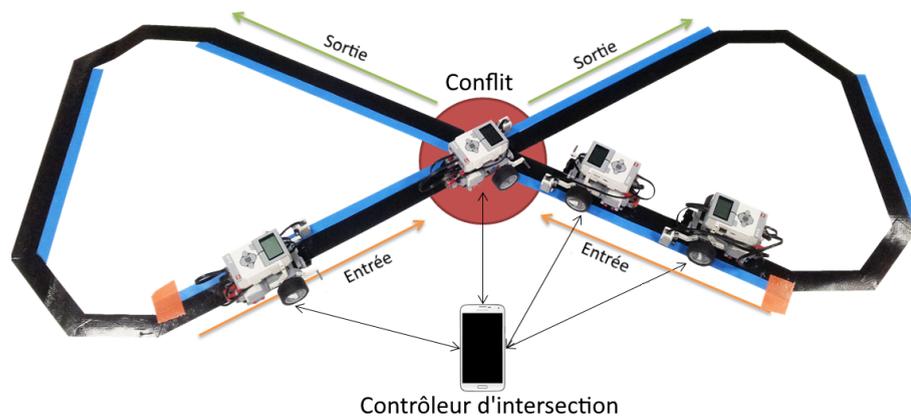


FIGURE 4.11 – Prototype d'intersection autonome composée de robots EV3 et d'un smartphone Android

4.4/ CONCLUSION

Par la simulation, nous avons pu mettre en évidence un cas d'inter-blocage possible lors de la distribution du droit de passage avec le protocole TIM en cas de difficultés de communication réseau. Toutefois, ce défaut n'est pas directement dépendant de ces conditions spécifiques car il peut survenir dès que la priorité accordée à un véhicule ne correspond pas à sa position physique, indépendamment du protocole utilisé. En effet, le problème provient d'une caractéristique de la régulation coopérative du trafic qui est la création d'une relation d'ordre (que nous nommons ordre virtuel) distincte de la relation d'ordre physique (que nous nommons ordre réel).

Cette situation d'inter-blocage paralyse complètement l'intersection, nous proposons donc un algorithme permettant d'y remédier. Contrairement aux solutions classiques de prévention ou résolution de l'inter-blocage, il faut respecter des contraintes spécifiques au trafic : seule la réorganisation de la séquence de passage est possible, et cette dernière ne doit pas induire un risque de collision.

L'algorithme proposé consiste à réordonner la séquence de passage lors de la découverte de véhicules supplémentaire, tout en évitant de modifier la priorité effective des véhicules présents à l'intersection.

Enfin, nous démontrons que l'algorithme répond aux contraintes posées (c'est-à-dire qu'il résout effectivement l'inter-blocage, et qu'il n'introduit pas de risque de collision). Les résultats sont ensuite validés expérimentalement par la simulation et par la réalisation d'un prototype réel d'intersection autonome conçu avec des robots EV3.

IMPLÉMENTATION DE L'INTERSECTION AUTONOME

L'analyse théorique du problème et la réalisation de simulations permettent de prévenir en amont les erreurs lors du processus de conception. Cependant, bien que la simulation tente de reproduire au mieux le comportement du réel, il est possible que des problèmes potentiels n'aient pas été identifiés, et ne se manifestent que lors des phases d'implémentation et de déploiement.

Ce chapitre s'attache à la description de cette phase finale, par la création d'un prototype réel d'intersection autonome. En premier lieu, une architecture matérielle et une architecture logicielle sont proposées, sur la base des énoncés des chapitres précédents. Ensuite, nous décrivons précisément les conditions d'expérimentation, puis les résultats obtenus. En ce sens, nous réalisons ici une synthèse des travaux présentés dans les chapitres précédents.

L'objectif est multiple : présenter une expérimentation en conditions réelles de l'intersection autonome et les résultats obtenus, fournir les informations nécessaires pour reproduire l'expérimentation, et rapporter les difficultés rencontrées afin de fournir un retour d'expérience à destination des personnes impliquées dans la réalisation de projets similaires.

Cette expérimentation a fait l'objet d'une démonstration au congrès ITS WC 2015 à Bordeaux.

5.1/ MÉTHODOLOGIE ET OUTILS LOGICIELS

Le travail réalisé dans le cadre de l'expérimentation de l'intersection autonome a consisté à, d'une part équiper un véhicule de capteurs et d'actionneurs, d'autre part réaliser le logiciel de contrôle du véhicule et du contrôleur d'intersection.

Les difficultés inhérentes à la conception de systèmes cyber-physiques (en termes de conception, d'implémentation et de tests) [Al-Jaroodi et al., 2016], et les délais de développement très courts ont nécessité la mise en place d'une méthodologie permettant d'encadrer le développement. Pour ce faire, il existe de nombreuses méthodologies d'ingénierie (Agile, XP, Scrum, etc.) fondées sur le principe du développement itératif : le développement est partitionné en une succession de cycles de courte durée à l'issue de chacun desquels résulte un produit testable. Pour la réalisation d'un produit commercial,

cela permet de confronter régulièrement le produit aux attentes du client afin de pouvoir réadapter rapidement les spécifications du projet. Dans notre cas, cela permet de confirmer régulièrement la faisabilité et la validité des solutions proposées.

Néanmoins, ces méthodologies sont initialement conçues pour du développement logiciel et, de fait, ne prennent pas explicitement en considération certaines problématiques spécifiques, telles que le développement, la configuration, l'intégration et la validation de systèmes cyberphysiques [Berger et al., 2012]. Ainsi, il a fallu réaliser quelques adaptations pour s'inscrire dans le cadre proposé par les méthodologies Agile.

Ces modifications ont pour l'essentiel déjà été présentées dans ce mémoire et ont consisté en la création d'un banc de test virtuel : dans un premier temps, l'ensemble du système a été développé en simulation puis les différents composants simulés ont été successivement remplacés par les composants réels. Ce banc de test virtuel était notamment composé d'une plateforme de simulation multi-agent permettant de valider les comportements émergeant de l'interaction de plusieurs agents.

Le processus global de production de l'intersection autonome est illustré par la figure 5.1.

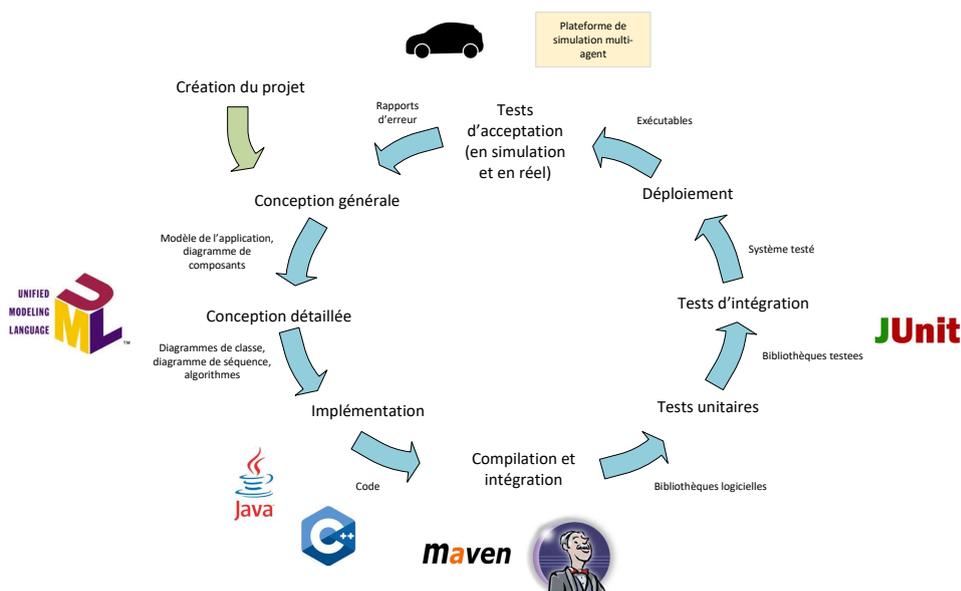


FIGURE 5.1 – Diagramme illustrant les étapes du processus de développement de l'intersection autonome et les outils associés

Les tests réels étant coûteux en temps et en ressources, ceux-ci ont été réduits au minimum. Ainsi deux cycles de développement étaient menés en parallèle : un cycle court (d'une durée d'environ une semaine) dans lequel tous les tests étaient réalisés en simulation, et un cycle long (d'une durée d'environ huit semaines) permettant de valider le prototype (ou des aspects de celui-ci) en conditions réelles.

Enfin, parmi les méthodes Agile, nous avons choisi la méthodologie BDD. Sur la base du modèle présenté dans le chapitre 2, les composants simulés ont été implémentés de manière à reproduire au mieux le comportement attendu des composants réels, et des tests unitaires ont été développés simultanément auxquels pouvaient être soumis indifférem-

ment les composants simulés et leurs équivalents réels. De cette façon, nous pouvons valider que les composants simulés et réels présentent les mêmes comportements attendus, et prévenir d'éventuelles régressions lors de l'intégration d'un composant réel en lieu et place d'un composant simulé dans le système.

Dans ce cadre, les outils utilisés sont :

- UML pour la modélisation de l'application
- Maven pour la gestion du cycle de vie du produit et Git pour la gestion du code source
- Java et C++ pour l'implémentation
- JUnit avec Mockito pour la conception et l'exécution des tests unitaires et d'intégration
- Hudson pour l'intégration continue
- Bitbucket Issues pour le suivi des problèmes et la gestion de projet

Grâce à l'application de cette méthodologie et de ces outils, nous sommes parvenus à limiter considérablement les temps de tests. Les premiers tests de commande du véhicule ont eu lieu en juillet 2015, les premiers tests du système complet ont commencé en septembre 2015 et la démonstration en situation réelle a eu lieu du 5 au 9 octobre 2015.

5.2/ ARCHITECTURE MATÉRIELLE

Les composants requis pour la réalisation d'une intersection autonome ont été détaillés dans le chapitre 2. Le focus de ce chapitre étant placé sur l'implémentation réelle de l'intersection autonome, ce sont les composants réels, provenant de l'implémentation du modèle de composant, qui sont présentés.

5.2.1/ PRÉPARATION DES VÉHICULES

Pour réaliser l'intersection autonome, il faut plusieurs véhicules pilotables. Cela signifie que les commandes d'accélération, de freinage et de direction du véhicule doivent pouvoir être pilotées par un ordinateur programmable, sans que l'interaction d'un utilisateur soit nécessaire.

Pour ce faire des véhicules Renault Scenic III 5.2 ont été préparés par FAAR Industry.

Suite à la préparation de ces véhicules, la pression sur la pédale d'accélérateur et sur la pédale de frein, et l'application d'un angle sur le volant peuvent être pilotées par un ordinateur embarqué dans le véhicule.

5.2.1.1/ ACCÉLÉRATION

Dans le cas du Renault Scénic, la transmission de l'information de pression sur la pédale d'accélération est réalisée par la transmission d'un signal numérique passant sur le bus Controller Area Network (CAN). FAAR Industry a équipé les véhicules d'un calculateur dont le rôle est d'intercepter le message émis par le calculateur de la pédale d'accélération. Le message peut alors être réémis tel quel (mode non-autonome) ou une valeur différente peut être retransmise (mode autonome).



FIGURE 5.2 – Véhicule Xi.cars pilotable en accélération, freinage et direction

Ce calculateur intermédiaire est lui-même relié à un bus CAN secondaire. L'envoi de messages sur ce bus CAN secondaire permet de piloter le calculateur afin d'activer ou non le mode autonome, et d'indiquer la valeur de pression sur la pédale d'accélération à simuler (dans le cas du mode autonome).

Les messages de commande de l'accélération prennent deux formes :

- Désactivation de la commande d'accélération (passage en mode manuel)
- Contrôle d'accélération : ce message contient une valeur allant de 0 à 100 correspondant à un pourcentage de pression sur la pédale d'accélération

La latence d'application de la commande est de l'ordre de 50 ms.

5.2.1.2/ FREINAGE

La commande de freinage, dans le cas du Renault Scenic, est mécanique : la pression sur la pédale de frein déplace des pistons dans les étriers qui vont mettre en contact les plaquettes de frein et le disque de freinage. La transmission entre la commande et les pistons est mécanique et réalisée par le liquide de frein.

Afin de simuler la pression pédale d'un utilisateur, un moteur est installé au niveau de la pédale de frein, et la rotation de ce moteur vient déplacer la pédale de frein par le biais d'un câble. Ce moteur se pilote de la même façon que le calculateur d'accélération intermédiaire : relié au bus CAN secondaire, il en reçoit des messages qui permettent d'activer ou non le mode autonome, et d'indiquer la position de la pédale de frein. Ces messages prennent deux formes :

- Désactivation de la commande de freinage (mode manuel), sans retour à la position initiale (c'est-à-dire la position sans freinage)
- Contrôle du freinage : ce message contient une valeur allant de 0 à 100 correspondant à un pourcentage de pression sur la pédale de frein

La latence d'application de la commande est inférieure à une seconde.

5.2.1.3/ COMMANDE VOLANT

De même que pour la commande de freinage, la direction du véhicule est commandée de manière mécanique (la modification de l'angle du volant vient mécaniquement modifier l'angle des roues). Ainsi, un moteur a été installé sous la colonne de direction et a été relié par une courroie à celle-ci : une rotation du moteur engendre une rotation du volant.

Le principe de fonctionnement est le même que pour la commande de frein : le calculateur du moteur pilotant le volant est relié au bus CAN secondaire. Il peut en recevoir des messages, qui permettent d'activer ou non le mode autonome, et qui permettent d'indiquer la valeur désirée pour la position du volant (le volant étant commandé en position). Ainsi les deux types de messages disponibles pour la commande volant sont :

- Désactivation du contrôle du volant (passage en mode manuel), sans retour à la position initiale
- Contrôle du volant : ce message contient une valeur sur l'intervalle $[-12500, 12500]$ linéairement associé à l'intervalle des angles possibles du volant allant de -450deg à $+450\text{deg}$, la valeur est appliquée sans délai

La latence entre l'envoi de la commande et le début de son application est inférieure à 100 ms. La vitesse de rotation du volant est supérieure à 300deg par seconde.

5.2.1.4/ CAPTEURS INTERNES

Le véhicule est équipé de capteurs *internes*. Ces capteurs incluent les capteurs proprioceptifs d'origine du véhicule, ainsi que les capteurs supplémentaires ayant été intégrés au véhicule par FAAR Industry.

Remontée d'information CAN véhicule Un calculateur connecté simultanément au bus CAN du véhicule et au bus CAN secondaire retransmet certaines informations provenant du CAN véhicule vers le CAN secondaire.

Ces informations concernent les capteurs proprioceptifs du véhicule :

- Vitesse du véhicule
- Accélération latérale
- Accélération transversale
- Vitesse de lacet
- Position du volant
- Vitesse du volant

Radar et Lidar Un radar et un lidar 4 plans (IbeoLux) ont été installés dans le véhicule pour la détection d'obstacle.

Le radar est relié au bus CAN secondaire et transmet les informations relatives aux objets perçus par ce biais.

Le lidar est accessible via une interface Ethernet RJ45 placée à l'arrière du véhicule.



FIGURE 5.3 – Boitier LIDAR IbeoLux

5.2.1.5/ ALIMENTATION ÉLECTRIQUE ET ARRÊT D'URGENCE

Pour alimenter l'ensemble des équipements supplémentaires (moteurs et capteurs internes), deux batteries 12V ont été ajoutées à l'arrière du véhicule. Ces batteries sont connectées à la batterie du véhicule (mais peuvent être déconnectées via un interrupteur). En marche normale, l'alternateur permet donc d'alimenter ces batteries.

Un boitier d'alimentation permet d'alimenter ou non les différents composants supplémentaires. Lorsque l'interrupteur central de ce boitier est *off*, les équipements supplémentaires ne sont plus alimentés et sont désactivés.

Deux boutons d'arrêt d'urgence placés à l'intérieur du véhicule permettent de contrôler le boitier d'alimentation et de couper l'alimentation des équipements supplémentaires du véhicule. Cette désactivation arrête les calculateurs et moteurs, permettant au conducteur de reprendre la main sur l'ensemble des commandes du véhicule.

5.2.2/ CAPTEURS EXTERNES ET COMMUNICATION SANS-FIL

Pour la réalisation de l'intersection autonome, des capteurs externes ont été rajoutés aux véhicules.

Capteur de positionnement Un récepteur GPS RTK (modèle ProFlex 500 de Magellan/AshTech, voir figure 5.4) a été installé dans chaque véhicule. Afin d'obtenir des informations de correction, une station correctrice similaire était utilisée. Les informations de correction étaient transmises par liaison radio Ultra Haute Fréquence (UHF).



FIGURE 5.4 – Boitier ProFlex 500

Les antennes de réception satellite et de liaison UHF étaient placées sur le toit du véhicule.

Unités de communication Pour permettre aux véhicules de communiquer entre eux, des unités ITS-RS2-M de Commsignia (voir figure 5.5) ont été installées dans les véhicules. Ces unités respectent le standard IEEE 802.11p et permettent des communications DSRC. Après tests, nous avons observé que la communication entre les unités fonctionnait mal (pertes de messages très importantes au-delà de 60 mètres), nous empêchant de les utiliser en conditions réelles pour la démonstration : même si des pertes de messages sont acceptables du point de vue de la sécurité, elles entravent le déroulement de l'expérimentation lorsqu'elles sont trop importantes.



FIGURE 5.5 – Boîtier de communication Commsignia

Pour pallier ce problème, nous avons utilisé des antennes WiFi 802.11n à longue portée. Après expérimentation (échanges de messages en UDP jusqu'à 80 mètres) nous avons constaté de meilleurs résultats qu'avec les unités 802.11p. Nous avons donc retenu cette solution.

Nous pouvons noter que le standard IEEE 802.11p prévoit une portée supérieure à celle que nous avons pu observer, et nous pouvons raisonnablement supposer que les unités qui seront distribuées commercialement à l'avenir n'auront pas ce défaut. Ainsi, nous pouvons affirmer que l'utilisation de communication WiFi en lieu et place de communication 802.11p ne remet pas en cause l'intérêt de l'expérimentation quant à l'aspect communication inter-véhiculaire.

5.2.3/ ORDINATEUR EMBARQUÉ

Afin d'agrèger les données provenant des différents capteurs, et de commander les différents actionneurs du véhicule, un ordinateur embarqué a été intégré au véhicule. Cet ordinateur est relié au bus CAN secondaire via un adaptateur PCAN USB de Peak Systems et au GPS RTK via un adaptateur USB/DB9. Une antenne WiFi longue portée y est également connectée en USB.

Cet ordinateur était un ordinateur portable standard avec Windows 8.1, Java Runtime Environment 8, et les pilotes appropriés installés.

5.2.4/ SYNTHÈSE

La figure 5.6 synthétise les dispositifs matériels mis en place sur le véhicule afin d'assurer les fonctions de positionnement, de communication et de contrôle.

5.3/ ARCHITECTURE LOGICIELLE

L'architecture logicielle est celle définie dans le chapitre 2. Ici, nous entrons dans le détail de l'implémentation en définissant les composants logiciels implémentant les interfaces exposées dans le diagramme de composants.

La production de l'intersection autonome a conduit à la réalisation de deux logiciels : le contrôleur d'intersection (ou *intersection-server*), et le contrôleur du véhicule (ou *cooperative-autonomous-car*). La figure 5.7 détaille l'architecture en termes de modules du contrôleur d'intersection, tandis que la figure 5.8 détaille celle du contrôleur du véhicule.

Le rôle de chacun des modules présents dans cette figure est donné dans l'annexe B.

5.3.1/ CAPTEURS ET ACTIONNEURS

En accord avec une approche de type ingénierie assistée par ordinateur Computer-Aided Engineering (CAE), le processus de développement a commencé par des expérimentations en simulation. Pour cela, nous avons défini les interfaces que devaient exposer les composants du système, puis développé des composants logiciels simulant le comportement des composants réels et implémentant ces interfaces.

Dans un second temps, nous avons créé des composants logiciels capables de se connecter aux composants réels, et formatant les données provenant de ceux-ci de façon à répondre aux exigences des interfaces préalablement définies. En disposant de la même interface que les composants simulés, les composants réels s'intégraient alors avec le système existant sans devoir modifier la structure de celui-ci.

Par exemple, le module *positioning* (voir figure 5.8) définit une interface générique *IPositionProvider*, dont une implémentation est *ProFlex500SerialPortReader*. Nous avons parallèlement développé une implémentation permettant de simuler le fonctionnement du GPS RTK intégrant imprécision et latence, afin de pouvoir effectuer les tests d'intégration du système y compris en l'absence du dispositif matériel.

5.3.2/ COMMUNICATION

Les fonctions de communication sont regroupées dans les modules *communication-core*, *intersection-client* et *intersection-server* (voir figure 5.7) qui proposent une implémentation des protocoles CVAS et TIM sur la base des protocoles réseaux User Datagram Protocol (UDP) et WSMP. Le détail de l'architecture de ces modules est donné dans l'annexe B.

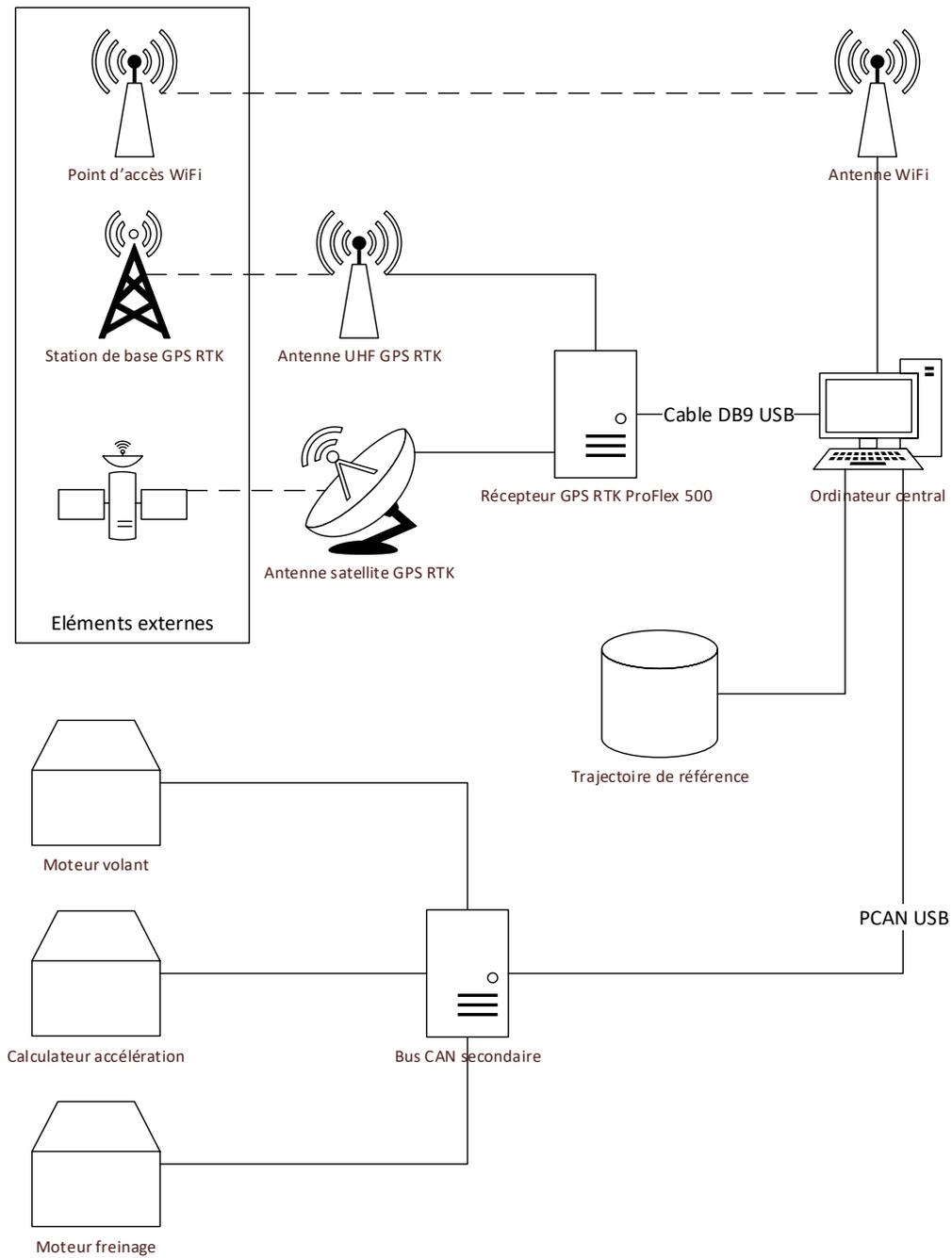


FIGURE 5.6 – Illustration schématique de l'architecture matérielle d'un véhicule

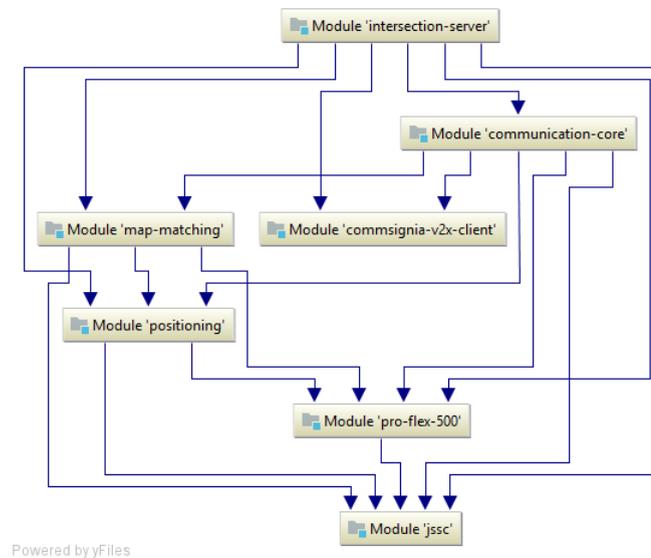


FIGURE 5.7 – Architecture du logiciel contrôleur d'intersection au niveau *modules*

5.4/ DÉFINITIONS DES CONDITIONS D'EXPÉRIMENTATION

5.4.1/ OBJECTIFS DE L'EXPÉRIMENTATION

La réalisation de l'intersection autonome a pour but principal de valider la faisabilité de l'intersection autonome dans les conditions matérielles préalablement décrites, mais elle constitue également une synthèse des contributions des chapitres précédents.

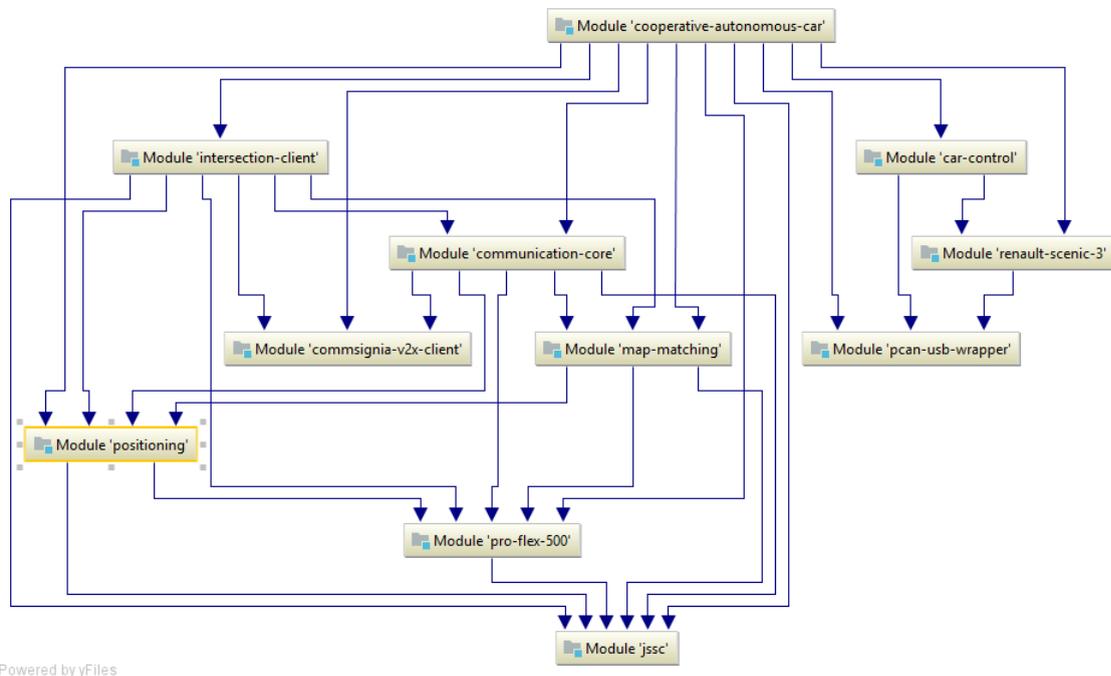
De ce fait, les objectifs de l'expérimentation en conditions réelles sont multiples. Celle-ci doit permettre de valider l'intersection autonome via le protocole TIM (chapitre 2), valider les travaux relatifs à la commande de direction et au suivi de trajectoire (chapitre 3), et valider la stratégie anti-inter-blocage (chapitre 4).

5.4.2/ SCÉNARIO D'ESSAI

Afin de réaliser l'intersection autonome, nous commençons par définir un circuit. Nécessairement ce circuit doit comporter une intersection, nous choisissons donc un circuit en forme de huit. Ainsi, les véhicules peuvent rouler indéfiniment et négocient une intersection deux fois par tour de circuit.

Trois véhicules automatisés sont placés sur ce circuit. La présence d'au moins trois véhicules permet notamment de créer les conditions permettant l'apparition d'un inter-blocage tel que présenté dans le chapitre 4. Elle complexifie également la création de la séquence de passage par rapport à une intersection à 2 véhicules dans laquelle l'un est forcément prioritaire sur l'autre.

Dans le cadre de l'expérimentation, les radars des véhicules sont désactivés étant donné qu'ils ne sont pas nécessaires : l'ensemble des opérations d'évitement de collision (longitudinales ou latérales) étant prévenues par l'utilisation de la communication couplée au positionnement par GPS RTK. Nous pouvons ainsi valider que même en l'absence de

FIGURE 5.8 – Architecture du logiciel contrôleur du véhicule au niveau *modules*

perception via radar, les véhicules n'entrent pas en collision à l'intersection.

5.4.3/ PROCESSUS DE TEST

Le processus d'expérimentation se fait en trois étapes :

1. Acquisition du tracé de référence à suivre et marquage du tracé
2. Installation et positionnement des véhicules
3. Démarrage des véhicules et du contrôleur d'intersection

L'acquisition du tracé se fait en conduite manuelle avec le GPS RTK activé. Le conducteur effectue un tour du tracé à suivre à vitesse constante afin d'acquérir la liste de points composant le tracé. Le tracé obtenu est édité manuellement afin de définir les différentes sections de l'intersection (zone d'entrée, zone de conflit, zone de sortie). Le fichier généré est ensuite partagé entre tous les véhicules devant suivre le tracé.

Les véhicules sont par la suite installés, positionnés et démarrés conformément à la procédure de démarrage définie dans l'annexe C.

Dans le cadre de la démonstration, les véhicules sont configurés pour attendre d'avoir reçu l'information de position des trois véhicules présents avant de démarrer. La stratégie d'ACC reposant uniquement sur la communication sans-fil, l'absence de communication d'un véhicule pourrait résulter en une collision.

5.5/ EXPÉRIMENTATION ET RÉSULTATS

L'expérimentation a fait l'objet de multiples démonstrations qui ont eu lieu du 5 octobre 2015 au 9 octobre 2015 au congrès ITS WC à Bordeaux. Quatre démonstrations avaient lieu par jour, au cours desquelles des invités montaient à bord de chacun des trois véhicules. L'ensemble des fonctions de conduite du véhicule était piloté par ordinateur. Un conducteur restait cependant derrière le volant afin de pouvoir reprendre la main en cas de problème.

5.5.1/ SUIVI DE TRAJECTOIRE

Afin d'obtenir une précision optimale, les récepteurs GPS RTK et la station correctrice ont dû être placés à distance des bâtiments environnants. Une ligne de vue directe entre la station correctrice et les récepteurs est également requise.

La figure 5.9 montre le tracé du circuit enregistré via le GPS RTK lors de la phase d'acquisition (en conduite manuelle). Les points verts et jaunes indiquent respectivement la position de la station correctrice GPS RTK et du point d'accès WiFi. Les bandes orange, rouges et vertes indiquent respectivement les zones d'entrée, de conflit et de sortie.

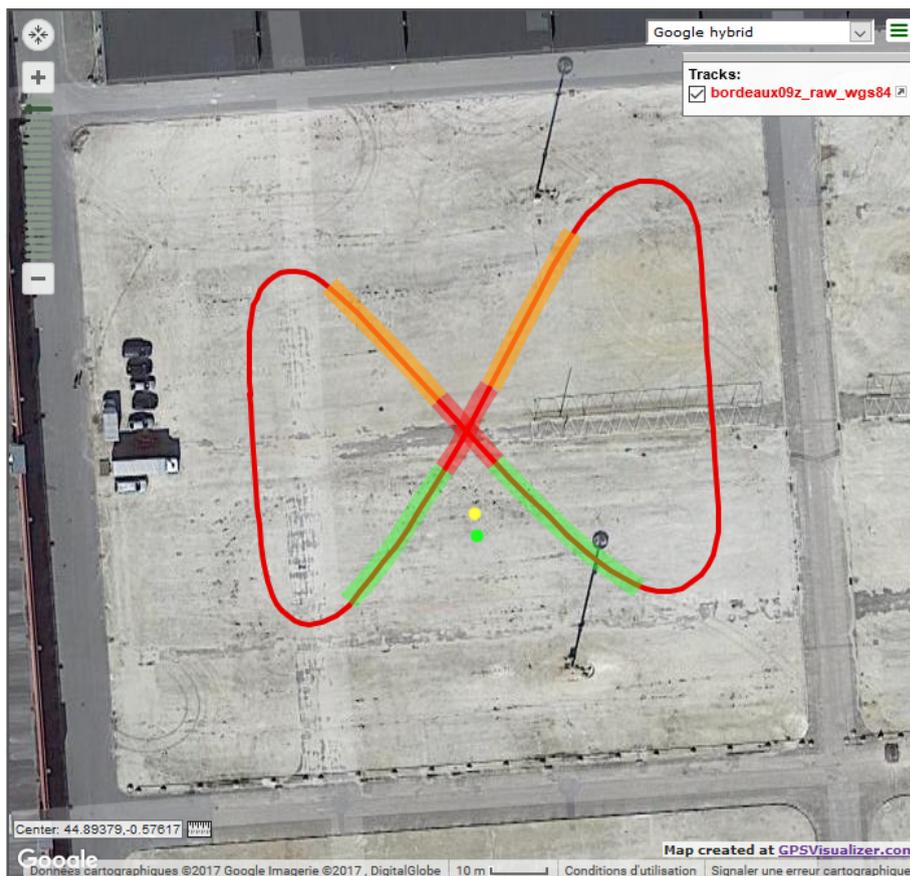


FIGURE 5.9 – Circuit de référence enregistré en utilisant le GPS RTK

Lors du suivi de trajectoire, le tracé était encadré par des délimiteurs de voie espacés de 5 mètres, laissant donc au plus 1,575m de chaque côté du véhicule large de 1,85m.



FIGURE 5.10 – Tests préliminaires de suivi de trajectoire sur le site du congrès ITS de Bordeaux : le véhicule suit une trajectoire prédéfinie et balisée en utilisant uniquement le positionnement satellite

La vidéo https://www.youtube.com/watch?v=xIndBZK96_A (dont est extraite la capture 5.10) montre le véhicule suivant la trajectoire préenregistrée, en utilisant uniquement le positionnement satellite et sans intervention humaine.

5.5.2/ INTERSECTION AUTONOME

L'utilisation de la communication UDP par WiFi en lieu et place de la communication IEEE 802.11p par DSRC a impacté le déroulement de certaines démonstrations. Malgré la longue portée des antennes WiFi utilisée, des déconnexions du point d'accès ont été constatées (qui ne seraient pas arrivées avec des messages DSRC). La reprise de la connexion pouvait alors prendre plusieurs secondes, et pendant ce temps le véhicule était incapable d'émettre ou recevoir des messages.

Aucune collision n'a été observée, même lors des pertes de communication. En effet, trois scénarii étaient possibles et aucun n'aboutissait à une possibilité de collision :

- Un véhicule est incapable de communiquer alors qu'il n'a pas le droit de passage : il s'arrête avant l'intersection, et les véhicules le suivant s'arrêtent derrière la dernière position connue du véhicule
- Un véhicule perd la communication alors que le contrôleur d'intersection l'a enregistré, mais n'a pas reçu la confirmation du contrôleur : le véhicule s'arrête avant l'intersection, et les autres véhicules s'arrêtent soit à l'intersection (le véhicule ayant perdu la connexion a malgré tout été accepté et est prioritaire), soit derrière la dernière position connue du véhicule
- Un véhicule perd la connexion après avoir été accepté et avoir été notifié de son droit de passage : le véhicule traverse l'intersection, mais puisqu'il ne peut notifier sa sortie de l'intersection, les autres véhicules ne franchissent pas celle-ci

Enfin, lors de la reprise de la connexion, la stratégie de résolution de l'inter-blocage présentée dans le chapitre 4 permettait d'éviter les cas d'inter-blocage.

En conséquence, l'arrêt de la démonstration était déclenché par l'arrêt du contrôleur d'in-



FIGURE 5.11 – Deux véhicules autonomes négocient leur droit de passage à l'intersection (un troisième véhicule non présent sur l'image se trouve également sur le circuit)

tersection : en perdant toute possibilité de communication et de négociation les véhicules tombaient nécessairement dans un des trois scénarii préalablement définis et s'arrêtaient sans collision.

La vidéo <https://www.youtube.com/watch?v=KBx6qks8kCI> (dont est extraite la capture 5.11) est un enregistrement réalisé au cours de tests préalables à la démonstration, et montre l'intersection de trois véhicules en utilisant uniquement la communication sans fil pour la gestion du droit de passage et la prévention des collisions.

5.6/ CONCLUSION

La réalisation et la démonstration d'une intersection autonome à plus de deux véhicules complètement automatisés fut un succès et une première mondiale. Un large public put y assister, y compris depuis l'intérieur des véhicules, et de nombreux articles de presse généraliste (voir annexe A) ont couvert l'évènement.

Ce résultat est la synthèse de différents travaux de l'Université de Technologie de Belfort-Montbéliard (UTBM) sur le véhicule autonome et la gestion coopérative du trafic. Il est également le résultat de l'intégration des différentes contributions présentées dans ce mémoire : les travaux relatifs à la commande du véhicule ont permis de le contrôler de façon à ce qu'il suive le circuit, et les travaux sur le protocole de négociation ont permis d'éviter l'apparition d'une situation d'inter-blocage. Ainsi l'intersection autonome constitue une validation expérimentale des contributions préalablement énoncées.

De cette expérience nous pouvons conclure que le principe d'intersection autonome est réalisable en conditions réelles, et que pour sa mise en place il suffit de disposer d'une solution de positionnement relativement précise, d'une communication réseau (même très imparfaite), et de stratégies de contrôle du véhicule prenant ces informations en entrée.

Ainsi, de nouvelles perspectives s'ouvrent. D'une part, nous pouvons nous attacher à standardiser l'intersection autonome (protocoles de communication et de négociation,

capteurs requis, etc.) afin de commencer, dès aujourd'hui, à envisager un déploiement du système à grande échelle, d'abord dans des espaces restreints (usines, navettes d'aéroport, etc.) puis sur l'infrastructure routière.

D'autre part, nous pouvons poursuivre en généralisant le concept à l'ensemble des opérations routières impliquant de déterminer formellement un ordre de passage des véhicules, dans l'optique de pouvoir les sécuriser et les optimiser. De telles opérations incluent les changements de voie sur autoroute, les insertions, et éventuellement certaines sorties de stationnement. Ces opérations diffèrent de l'intersection étant donné qu'elles n'ont pas systématiquement lieu dans un espace bien délimité. Un travail est donc nécessaire pour adapter les protocoles existants et fait l'objet du chapitre suivant.



GÉNÉRALISATION DE L'INTERSECTION AUTONOME

DE L'INTERSECTION AU CHANGEMENT DE VOIE COOPÉRATIF

Depuis quelques années, certains véhicules haut-de-gamme, tels que les Mercedes Classe E ou les Tesla Model S, sont équipés de solutions de dépassement semi-autonome. Sur autoroute, sur une action du conducteur sur le clignotant, le véhicule réalise un changement de voie vers la direction désignée. Pour ce faire, le véhicule applique une commande sur le volant, après avoir préalablement vérifié (grâce aux détecteurs d'obstacle) que la zone était libre et que le mouvement était réalisable.

Ces solutions de changement de voie automatiques assurent juste la sécurité de l'opération, mais le choix de changer de voie ou non revient encore au conducteur. De plus, la stratégie de régulation de vitesse appliquée est la stratégie du système d'ACC : le véhicule adapte sa vitesse en fonction du véhicule qui le précède, ainsi lorsqu'un changement de voie est réalisé et qu'il n'y a plus d'obstacle réduisant la vitesse du véhicule, celui-ci peut accélérer. Il n'y a pas, dans les solutions actuelles, de stratégies destinées à optimiser la fluidité du trafic lors du changement de voie par synchronisation des vitesses avec les véhicules voisins.

Ces problématiques rappellent celles de l'intersection autonome. En effet, le cas du dépassement peut être vu comme une forme d'intersection, où la zone de conflit n'est plus définie de manière statique, mais est mobile. En abstrayant les principes de la gestion coopérative des intersections, il est possible de définir un cadre générique applicable aux situations de dépassement et d'insertion.

Ce chapitre propose une solution de gestion coopérative du changement de voie, en utilisant la communication sans-fil pour introduire une courtoisie automatisée entre véhicules. L'objectif est d'améliorer la fluidité de l'opération pour améliorer la qualité du trafic sur autoroute. A partir des fondations ainsi posées, un modèle logiciel d'optimisation basé sur la théorie des algorithmes évolutionnistes, et plus spécifiquement des algorithmes métriques, est proposé pour calibrer automatiquement et dynamiquement le système.

6.1/ GÉNÉRALISATION DU CONCEPT D'INTERSECTION

6.1.1/ DÉFINITIONS

Un changement de voie, sur une route à plusieurs voies, est le déplacement d'un véhicule d'une voie d'origine (ou ancienne voie) à destination d'une voie cible (ou nouvelle voie). Nous définissons les véhicules potentiellement impliqués dans l'opération de la façon suivante :

- Le véhicule principal (VP) est le véhicule qui effectue le changement de voie
- Le véhicule qui précédait le véhicule principal avant le changement de voie est l'ancien prédécesseur (AP)
- Le véhicule qui précède le véhicule principal après le changement de voie est le nouveau prédécesseur (NP)
- Le véhicule qui suit le véhicule principal après le changement de voie est le nouveau suiveur (NS)

Ces définitions sont celles utilisées dans [Erdmann, 2014]. La figure 6.1 illustre l'ensemble de ces définitions.

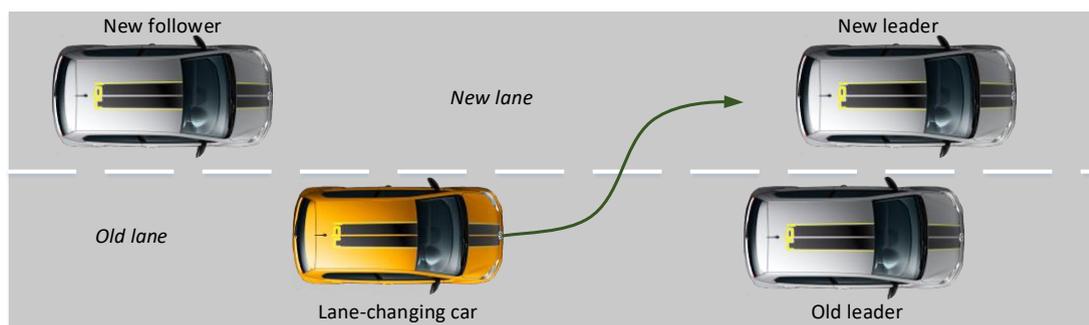


FIGURE 6.1 – Définitions du changement de voie

6.1.2/ INTERSECTION MOBILE

Dans le chapitre 2, nous avons présenté la définition d'une intersection. Celle-ci est caractérisée par la présence d'une zone de conflit entre deux mouvements distincts.

Lors d'une opération de changement de voie, nous retrouvons les mêmes caractéristiques : le mouvement du véhicule qui dépasse entre en conflit avec le mouvement des véhicules sur la voie cible. L'intersection entre ces mouvements forme une zone de conflit. Cependant, au contraire de l'intersection « classique », cette intersection n'a pas une position géographique statique, c'est une zone d'intersection « mobile » (voir figure 6.2).

Ainsi de la même façon qu'une intersection de véhicule peut être régulée par la communication sans fil, il est possible de réguler l'opération de changement de voie. Une différence importante existe néanmoins : l'absence d'une position fixe pour l'intersection empêche l'installation d'une infrastructure propre à l'intersection, ainsi il n'est pas possible d'utiliser un contrôleur d'intersection centralisé comme c'était le cas pour l'intersection classique.

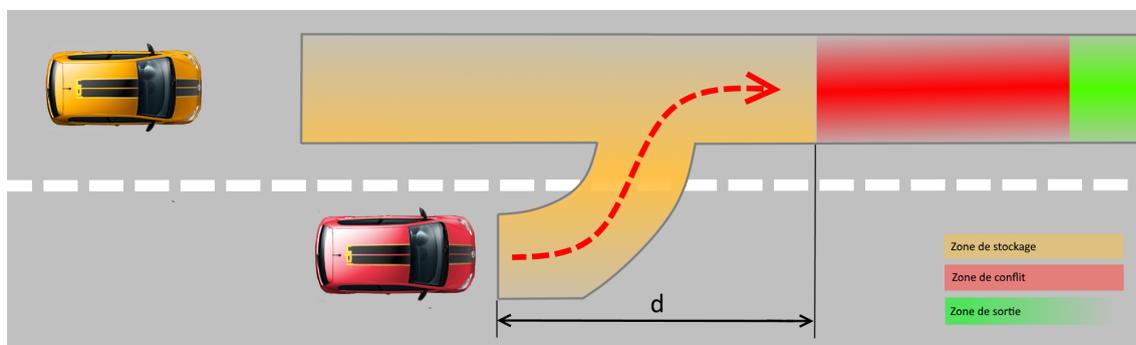


FIGURE 6.2 – Schéma de l'intersection mobile avec découpage en zones ; la distance d est la distance du véhicule désirant changer de voie à la zone de conflit

La coopération doit donc se faire de façon décentralisée. Cette opération, dangereuse dans le cas classique, est rendue possible ici par le fait que les véhicules circulent sur une même route et peuvent donc se percevoir mutuellement. Ainsi les stratégies de changement de voie automatiques agissent en système de secours et préviennent le risque de collision.

6.2/ ADAPTATION AU CHANGEMENT DE VOIE

6.2.1/ MODÈLE THÉORIQUE DU CHANGEMENT DE VOIE

Différents travaux ont proposé un modèle théorique du changement de voie. Le rôle de ces modèles est multiple :

- Définir sous quelles conditions un conducteur est amené à effectuer un changement de voie
- Définir sous quelles conditions le changement de voie peut être effectué en toute sécurité
- Définir comment ce changement de voie est effectué, notamment en ce qui concerne l'accélération et la direction du véhicule

6.2.1.1/ MOBIL

Le modèle MOBIL (minimizing overall braking induced by lane change) a été proposé par [Kesting et al., 2007] afin de modéliser le changement de voie. Son objectif est double : fournir un modèle du changement de voie pour la simulation, et proposer une stratégie de changement de voie applicable à des véhicules autonomes.

Son principe consiste à minimiser les décélérations induites par le freinage, pour cela le modèle repose sur IDM afin de fournir une estimation des accélérations et décélérations potentielles pouvant survenir du fait du changement de voie. En effet, lorsqu'un véhicule change de voie, son nouveau suiveur possède un nouvel obstacle frontal plus proche que le précédent. Le nouveau suiveur peut être amené à réduire sa vitesse pour respecter le temps inter-véhiculaire avec le véhicule principal.

6.2.1.2/ MODÈLE SUMO

[Erdmann, 2014] présente les différentes causes de changement de voie :

- Le changement de voie stratégique : le changement de voie est requis pour que le véhicule atteigne sa destination.
- Le changement de voie coopératif : le changement de voie est réalisé pour faciliter le mouvement d'autres véhicules (dans le cas d'une insertion par exemple)
- Le changement de voie tactique : le véhicule désire augmenter sa vitesse en dépassant un véhicule plus lent
- L'obligation de se rabattre : dans la plupart des pays, un véhicule doit circuler sur la voie située la plus à droite (à gauche dans les pays où la circulation est à gauche)

En intégrant ces différentes conditions, nous établissons un modèle de décision relatif au changement de voie qui, pour une situation donnée, définit si un dépassement doit être effectué.

6.2.1.3/ SYNTHÈSE

En prenant comme base ces modèles, nous pouvons définir l'architecture générale d'un système de changement de voie complètement automatisé.

Pour cela, trois composants sont nécessaires afin de répondre à chacune des interrogations suivantes :

- Le véhicule doit-il changer de voie ?
- Le véhicule peut-il changer de voie ?
- Comment le véhicule change-t-il de voie ?

Ainsi, le composant répondant à la première question fournit une méthode *isLaneChangeDesired()* indiquant si le changement de voie doit être réalisé, le composant répondant à la seconde fournit une méthode *isLaneChangePossible* indiquant si le changement peut être réalisé, et le dernier composant fournit une méthode *performLaneChange* permettant de réaliser le changement de voie.

6.2.2/ AUTOMATED COOPERATIVE LANE CHANGE (ACLC)

Sur la base de ces composants et modèles nous définissons l'algorithme 2 (résumé par la figure 6.4) [Lombard et al., 2017] permettant de réaliser un changement de voie en utilisant la communication sans-fil.

6.2.2.1/ PRINCIPE GÉNÉRAL

Le principe de base est similaire à celui de TIM, à savoir qu'une séquence de passage est créée pour une zone de conflit donnée. Les principales différences sont que la zone de conflit est définie par le véhicule qui souhaite changer de voie, et qu'il n'y a pas d'arrêt par défaut.

En effet, lorsqu'un véhicule désire changer de voie, il définit une zone d'intersection mobile et la transmet aux autres véhicules. Lors de la création de l'intersection, le VP se

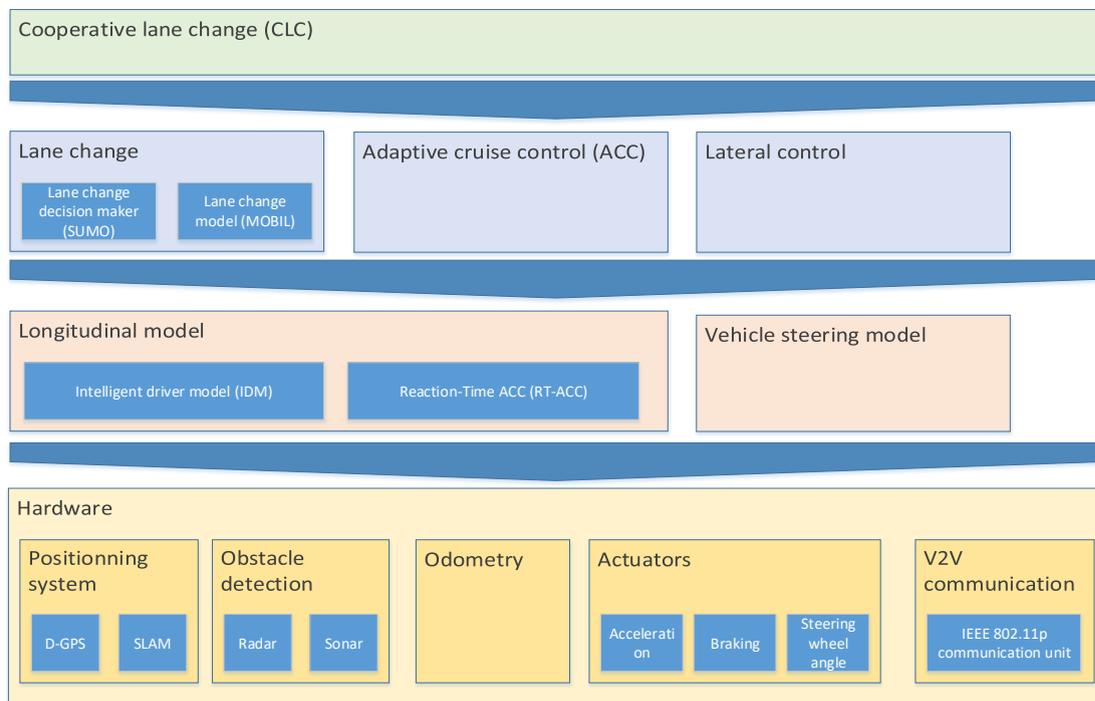


FIGURE 6.3 – Architecture du système de dépassement coopératif

place en première position dans la séquence de passage. Ensuite, les autres véhicules peuvent envoyer des requêtes de passage au VP (qui agit alors en tant que contrôleur d'intersection). Ces requêtes de passage étant systématiquement placées après le véhicule changeant de voie, elles constituent un accord de la part d'un NS signifiant qu'il ne franchira la zone d'intersection qu'après le VP. En accord avec TIM ces requêtes contiennent toute l'information nécessaire pour la synchronisation de vitesse. Enfin, pour des raisons de sécurité si un véhicule en position de NS n'émet pas de requête de passage (soit par défaut de communication, soit volontairement), cela signifie qu'il refuse le passage prioritaire au VP. Le VP passera donc après ce véhicule.

En créant la zone d'intersection mobile, nous permettons aux véhicules de synchroniser leur vitesse. Dans le cas d'une intersection fixe, cela sert à limiter l'occupation de la zone de conflit en coordonnant les véhicules pour qu'ils la franchissent à vitesse maximale. Dans le cas d'une intersection mobile, l'idée est de faire ralentir le NS afin de créer un espace inter-véhiculaire suffisant entre le NS et le NP pour que le VP puisse s'insérer.

6.2.2.2/ DÉFINITION DE LA ZONE MOBILE D'INTERSECTION

La particularité principale du protocole ACLC par rapport à TIM réside dans la définition de la zone mobile d'intersection. Celle-ci doit être suffisamment longue et éloignée pour permettre à l'opération de synchronisation de créer l'espace inter-véhiculaire requis tout en limitant la décélération requise par les NS. Afin d'anticiper les accélérations et les décélération faisant évoluer la position des véhicules à proximité (et donc de l'espace

Algorithm 2 Protocole Automated Cooperative Lane-Change (ACLCLC)

```

if isLaneChangeDesired() then
  if isLaneChangePossible() then
    performLaneChange()
  else
    intersection ← defineIntersection()
    cooperativeVehicles ← []
    repeat
      broadcastIntersectionData(intersection)
      getCooperativeVehicles(cooperativeVehicles)
      synchronizeSpeed()
    until isCLCPossible(cooperativeVehicles) or
    !isLaneChangeDesired()
    if isCLCPossible(cooperativeVehicles) then
      performLaneChange()
    end if
    broadcastIntersectionEnd(intersection)
  end if
end if
  
```

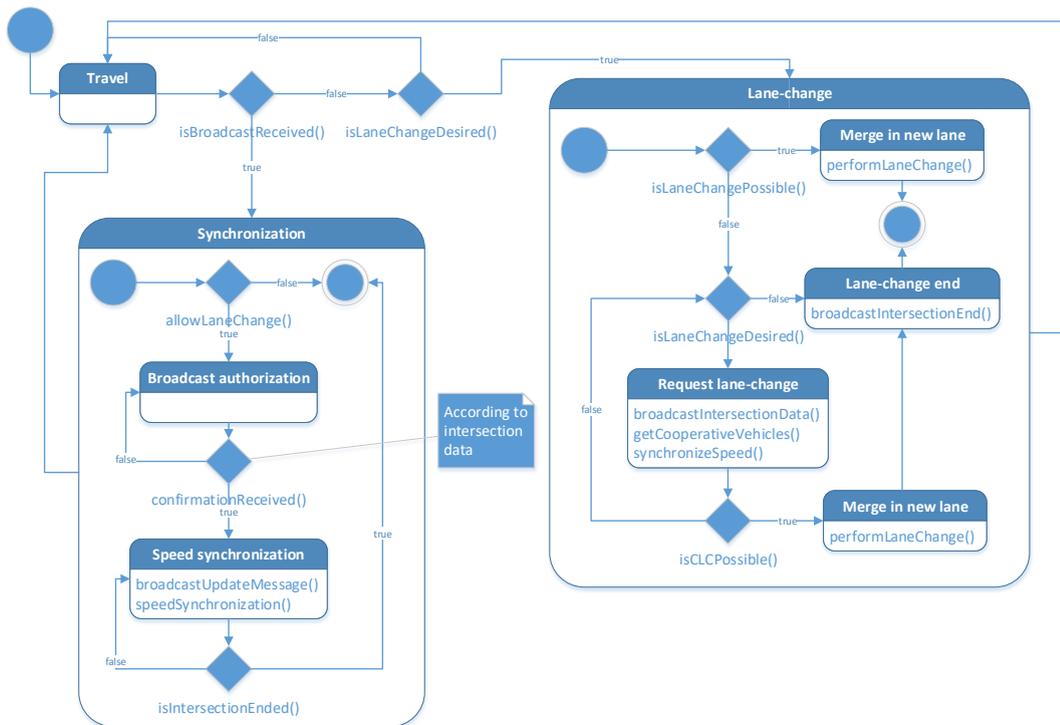


FIGURE 6.4 – Diagramme états-transitions du dépassement coopératif

inter-véhiculaire), nous nous appuyons sur un modèle de contrôle longitudinal.

Ainsi, nous définissons via l'équation 6.1 d_i la distance minimale à l'entrée de la zone de conflit.

$$d_i = \min(u, v_0 \times \left(\tau + \sqrt{\frac{h+l}{-b_f}} \right)) \quad (6.1)$$

Dans 6.1, u est une potentielle limite pour le point d'intersection (par exemple, la fin d'une voie d'insertion), h est la distance inter-véhiculaire minimale permise par le modèle longitudinal, v_0 est la vitesse initiale, b_f la valeur de décélération confortable, l est la longueur du véhicule et τ est un temps de réaction, c'est-à-dire le délai estimé entre l'émission de la requête de changement de voie et l'application de la synchronisation de vitesse par les NS.

La distance minimale de sécurité entre deux véhicules dans la nouvelle voie est supposée valoir h , afin d'introduire un espace suffisant celle-ci doit être augmentée à $2h + l$. Nous supposons que la vitesse du NP est constante tandis que le NS présente une décélération valant b_f . Ainsi, la position du NP est $p_l(t) = p_l(0) + v_0 t$ et celle du NS est $p_f(t) = p_f(0) + v_0 t + b_f t^2$.

On a donc :

$$\begin{aligned} p_l(t) - p_f(t) &\geq 2h + l \\ \Leftrightarrow p_l(0) + v_0 t - p_f(0) - v_0 t - b_f t^2 &\geq 2h + l \\ \Leftrightarrow p_l(0) - p_f(0) - b_f t^2 &\geq 2h + l \\ \Leftrightarrow h - b_f t^2 &\geq 2h + l \\ \Leftrightarrow t &\geq \sqrt{\frac{h+l}{-b_f}} \end{aligned} \quad (6.2)$$

Cette équation 6.2 donne le temps requis pour obtenir la distance inter-véhiculaire minimale compte tenu des hypothèses. Ensuite, le temps τ par souci de sécurité, et donc la distance parcourue pendant ce temps total est donnée par $v_0 \times (\tau + t) = v_0 \times \left(\tau + \sqrt{\frac{h+l}{-b_f}} \right)$.

$$T_{min} = t_r + \frac{kv_0}{2} \left(\frac{1}{a_f} - \frac{1}{a_l} \right) \quad (6.3)$$

En appliquant le temps inter-véhiculaire minimum de sécurité donné par 6.3 [Anderson, 1978], la relation 6.1 devient 6.4.

$$d_i = \min(u, v_0 \times \left(\tau + \sqrt{\frac{v_0 t_r + \frac{kv_0^2}{2} \left(\frac{1}{a_f} - \frac{1}{a_l} \right) + l}{-b_f}} \right)) \quad (6.4)$$

6.2.2.3/ CRITÈRE D'ACCEPTATION

Lorsqu'un véhicule en position de NS envoie une requête de passage au VP, il l'informe qu'il va ralentir pour le laisser s'insérer. L'objectif est alors de permettre le changement de voie tout en minimisant la décélération propre, ou en la limitant à une valeur donnée. Nous appelons critère d'acceptation la règle permettant de définir si oui ou non la requête

de passage doit être envoyée. Plusieurs paramètres influencent la décision de laisser ou non s'insérer un véhicule : la distance à la zone d'intersection, la vitesse relative du NS et du VP, la distance inter-véhiculaire entre le NS et le NP.

Quand la requête de changement de voie est reçue par le potentiel NS, il connaît la distance de la zone d'intersection par rapport au VP d_i , la position p du VP, et sa vitesse actuelle v . Le NS définit alors son véhicule prédécesseur comme étant le VP et considère une décélération de b_f . Nous supposons également que la vitesse du VP est constante.

Le temps alors requis pour que le VP atteigne la zone d'intersection est $t_i = \frac{d_i}{v}$.

La distance parcourue par le NS durant ce même temps est $d = v_0 t + b_f \times t^2$ avec $t = \min(t_i, \left| \frac{v_0}{b_f} \right|)$. Etant donné p_f la position du NS, alors celui-ci autorisera le changement de voie si $p_f + d < p + d_i$. Ainsi, l'équation 6.5 résume le critère d'acceptation.

$$p_f + v_0 t + b_f \times t^2 < p + d_i, t = \min(t_i, \left| \frac{v_0}{b_f} \right|) \quad (6.5)$$

6.2.2.4/ SYNTHÈSE

L'objectif du protocole ACLC est d'apporter les bénéfices de la régulation coopérative des intersections au cas du changement de voie. Pour ce faire, il en reprend les principes (découpage en zones, définition des messages de négociation, création d'une séquence de passage), et ainsi introduit les points clés suivants :

- **Décélération acceptée (ou courtoisie automatisée)** : quand le changement de voie est autorisé, le NS accepte de décélérer pour laisser le VP s'insérer, lui permettant ainsi de s'insérer alors qu'il ne le pourrait pas autrement
- **Refus par défaut** : sans une autorisation explicite de la part des NS, l'autorisation de dépassement est supposée refusée. De fait, un message d'autorisation perdu ou un NS refusant d'envoyer un message d'autorisation interdisent les dépassements qui produiraient une décélération des NS
- **Amélioration de la sécurité** : un NS ne peut donner l'autorisation d'insertion qu'à un véhicule simultanément, interdisant ainsi à plus d'un véhicule de s'insérer devant lui
- **Synchronisation de vitesse** : le NS connaît la position et la vitesse du VP, lui permettant de synchroniser sa vitesse avec lui et lissant l'opération de dépassement

6.2.3/ SIMULATION ET RÉSULTATS

L'objectif de la simulation est de vérifier si l'utilisation de la coopération par communication sans-fil permet d'obtenir un gain en termes de qualité de trafic (temps de trajet moyen, fluidité, etc.).

Pour cela nous réalisons le scénario de simulation suivant : sur une section de voie double, des véhicules sont générés (voie de droite) à fréquence fixe f ; un obstacle situé à une distance L du point de génération (par exemple des travaux) oblige les véhicules à se rendre sur la voie de gauche. Des simulations sont lancées sans et avec utilisation du protocole de coopération. Sans coopération, le véhicule dépasse dès qu'il peut augmenter sa vitesse et que son mouvement ne provoquera pas la décélération du nouveau

suiveur. La stratégie de suivi utilisée (pour le suivi et l'estimation de l'accélération ou décélération potentielle des autres véhicules) est RT-ACC. Le scénario de simulation est illustré par la figure 6.5.

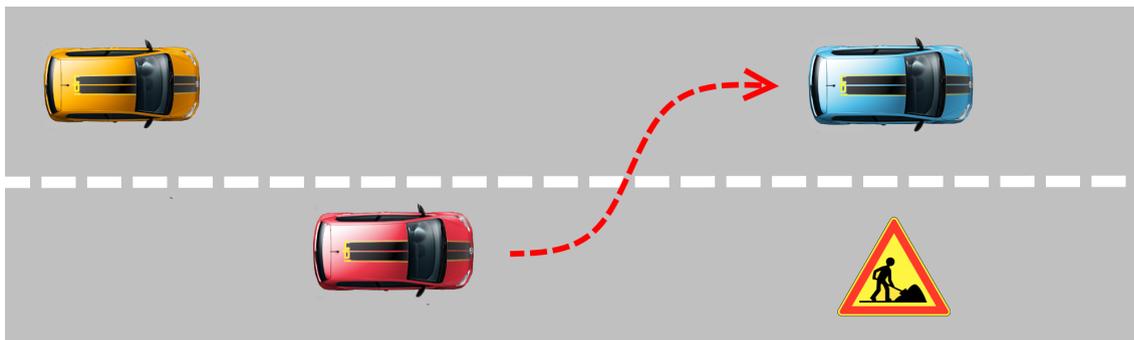


FIGURE 6.5 – Scénario de simulation du changement de voie coopératif

Pour estimation de la fluidité du trafic, nous utilisons la relation 6.6 donnée par [Cohen, 1993], où T est le temps de simulation, $a(t)$ l'accélération instantanée du véhicule, a_{avg} et v_{avg} les accélérations et vitesses moyennes des véhicules.

$$G = \frac{\sqrt{\frac{1}{T} \int_0^T (a(t) - a_{avg})^2 dt}}{v_{avg}} \quad (6.6)$$

Les tables 6.1 et 6.2 montrent respectivement les mesures réalisées avec ACLC et avec une politique utilisant uniquement le radar (c'est-à-dire sans communication).

	Vitesse moyenne	G moyen	Temps d'arrêt	Véhicules sortis
Moyenne	8.072	$2.18e^{-2}$	0.198	43.62
σ	0.361	0.002	0.135	0.701

TABLE 6.1 – Mesures utilisant ACLC pour la négociation du changement de voie - 0.25veh/s pour 240s

	Vitesse moyenne	G moyen	Temps d'arrêt	Véhicules sortis
Moyenne	8.170	$2.44e^{-2}$	5.999	43.27
σ	0.354	0.002	2.557	1.009

TABLE 6.2 – Mesures utilisant le radar seul pour le changement de voie - 0.25veh/s pour 240s

Nous pouvons observer qu'il y a une légère réduction de la vitesse moyenne en utilisant ACLC (-1.21%) bien que le nombre de véhicules sortis ne soit pas significativement réduit. Cela s'explique par le fait que des véhicules plus lents (vitesse cible inférieure), initialement situés dans la voie de droite, sont alors capables de s'insérer dans la voie de gauche et en réduisent donc la vitesse moyenne. Dans ce scénario, le débit du trafic est limité par la taille de la route, ainsi une augmentation du débit ne peut être attendue par l'utilisation de la communication. De plus, nous pouvons voir une légère amélioration de la valeur de G (-10.64%) indiquant une meilleure fluidité.

Enfin, l'amélioration la plus significative est la réduction du temps d'arrêt (-96.7 %). En effet, sans communication et négociation, les véhicules dans la voie de droite étaient fréquemment bloqués derrière l'obstacle sans être capables de s'insérer dans la voie de gauche. En utilisant la négociation par V2X, les véhicules dans la voie de droite s'inséraient avec fluidité dans la voie de gauche, ne s'arrêtant presque jamais derrière l'obstacle.

Des simulations ont été réalisées avec d'autres valeurs de véhicules par seconde, et les résultats observés ont été similaires (par exemple, pour une génération d'un véhicule chaque 6 secondes, l'utilisation d'ACLIC a donné -0.24% pour la vitesse moyenne -6.87% pour G et -94.46% de temps d'arrêt par rapport à la solution *radar-only*).

Pour compléter cette étude, nous avons effectué une comparaison entre les résultats donnés par notre simulateur avec ACLIC et les résultats donnés par VISSIM [Fellendorf et al., 2010] en cas de trafic fort. À la différence de notre simulateur, VISSIM est un simulateur ayant vocation à reproduire le comportement humain. Les résultats sont donnés par la table 6.3

	Vitesse moyenne	G moyen	Temps d'arrêt	Véhicules sortis
ACLIC	8.072	0.015	1.39	81.8
VISSIM	6.422	0.038	2.5	87.7

TABLE 6.3 – Comparaison entre notre simulation avec ACLIC et VISSIM pour 0,5 veh/s pendant 240s

Les résultats en termes de véhicules sortis sont similaires bien qu'une meilleure performance ait été obtenue par VISSIM. Celle-ci s'explique notamment par des distances inter-véhiculaires plus faibles dans ce dernier. La valeur moyenne G est plus importante pour VISSIM, signifiant une moindre fluidité. Enfin, des collisions ont été observées avec VISSIM mais pas dans notre simulation.

6.3/ STRATÉGIE D'OPTIMISATION

L'utilisation de la coopération dans le cadre de l'intersection permet l'application de stratégies d'optimisation, notamment pour réduire le temps d'évacuation de l'intersection. De façon similaire, nous proposons d'utiliser la communication dans le cadre du dépassement coopératif afin d'optimiser le trafic.

6.3.1/ DÉFINITION DU PROBLÈME

Soit un ensemble de n véhicules V sur une section de route de longueur s comportant l voies. Chaque véhicule dispose d'une vitesse cible qui lui est propre. Afin d'optimiser le trafic, nous cherchons à réduire le temps de trajet moyen 6.7.

$$TM = \frac{1}{n} \times \sum_1^n \int_0^s v(t) dt \quad (6.7)$$

Pour cela nous proposons de ré-agencer les véhicules en opérant des dépassements. De manière simplifiée cela revient à passer de la situation de la figure 6.6 à celle de la figure 6.7. En effet, la situation décrite par la figure 6.7 est celle qui minimise le temps de trajet moyen (principe de l'ordonnancement selon le temps restant le plus court [Schrage et al., 1966]).

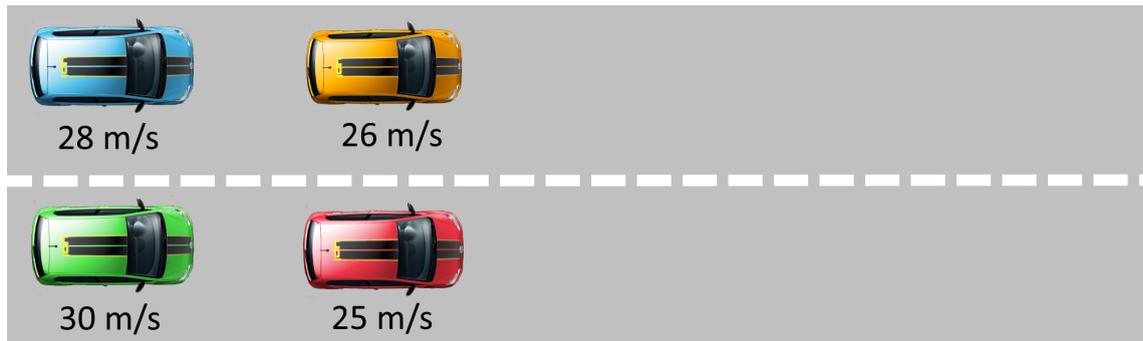


FIGURE 6.6 – Situation initiale du trafic : les véhicules sont disposés de façon aléatoire



FIGURE 6.7 – Situation optimisée du trafic : les véhicules sont ré-agencés en fonction de leur vitesse cible

Nous pouvons représenter ce problème sous la forme d'un graphe dirigé acyclique $G = (V, E)$ où chaque noeud représente une position sur la voie (sous forme discrète) et où les arcs permettent de décrire les mouvements des véhicules (avancer ou changer de voie). La figure 6.8 illustre ce graphe. Le problème consiste alors à trouver les chemins dans le graphe minimisant le temps de trajet total (recherche du chemin le plus court), sous la contrainte que deux véhicules ne peuvent occuper simultanément un même noeud. Exprimé ainsi, c'est une forme du problème de recherche du plus court chemin multi-agent, ou Multi-Agent Path-Finding (MAPF) [Sharon et al., 2015].

Nous pouvons supposer un cas trivial dans lequel les vitesses des véhicules sont supposées constantes et dans lequel les véhicules sont groupés, c'est-à-dire que le temps inter-véhiculaire entre deux véhicules successifs est réduit à son minimum, interdisant à un véhicule tiers de s'insérer entre eux. Le temps laissé aux véhicules pour se ré-organiser n'est pas limité. Nous ramenons alors le problème à un problème de séquençage de tâches, chaque véhicule constituant une tâche.

Dans ce cas précis, nous définissons l'heuristique 3 (pensée d'après [Schrage et al., 1966]) qui, appliquée à chaque véhicule, permet de les ré-agencer de façon optimale.

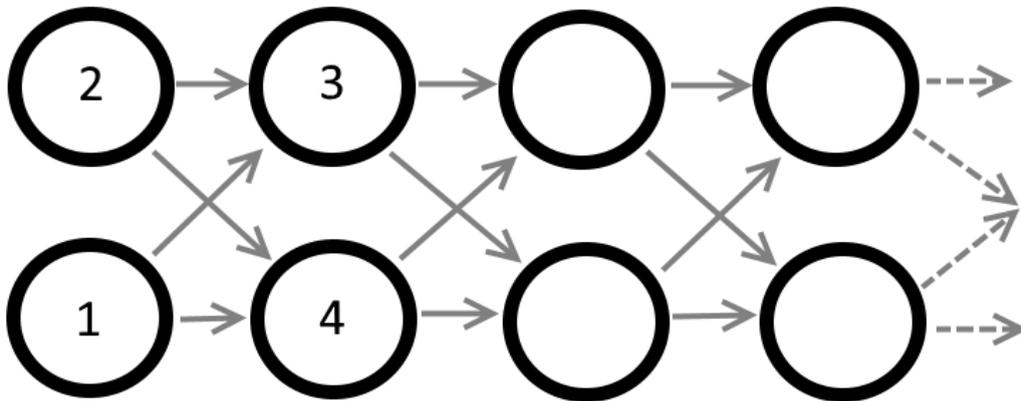


FIGURE 6.8 – Représentation sous forme de graphe de la situation décrite par la figure 6.6 : les chiffres représentent les véhicules ordonnés du plus rapide au plus lent

Algorithm 3 Heuristique pour le cas trivial

```

if  $l < N - 1 \wedge \exists c \in L(l)/v_c < v$  then
   $l \leftarrow l + 1$ 
else
  if  $l > 0 \wedge \nexists c \in L(l - 1)/v_c < v$  then
     $l \leftarrow l - 1$ 
  end if
end if

```

Avec V l'ensemble des véhicules, $L(i)$ l'ensemble ordonné des véhicules v de la voie i (i allant de 0 pour désigner la voie la plus à droite, à $N - 1$ où N est le nombre de voie), l désignant l'indice de la voie du véhicule considéré, v sa vitesse, v_c la vitesse du véhicule du véhicule c . Lors de l'affectation de l , le véhicule réalise un changement de voie. Indépendamment de l'algorithme présenté, les véhicules continuent à avancer.

Cet algorithme n'est néanmoins pas applicable en conditions réelles, les hypothèses requises n'étant pas réunies. Une autre stratégie d'optimisation doit donc être pensée.

6.3.2/ STRATÉGIE D'OPTIMISATION MÉMÉTIQUE

Dans le cas général, le problème, complexe, ne peut être résolu par une solution triviale : en effet, nous ne pouvons plus supposer que les véhicules sont groupés, ni que leur vitesse est constante.

Nous définissons pour chaque véhicule un comportement et un score (un score élevé signifiant que le comportement du véhicule est bien adapté aux conditions locales de circulation). Par l'utilisation de la communication inter-véhiculaire, chaque agent (ou véhicule) peut transmettre son comportement et son score aux autres agents. Ceux-ci peuvent exploiter ces informations pour mettre à jour leur propre comportement afin d'améliorer leur propre score. Cette stratégie de transmission du comportement interne de chaque agent

est inspirée par la transmission de la culture définie dans [Dawkins, 2016], avec le comportement du véhicule ayant le rôle de meme. Cela conduit naturellement à l'utilisation des algorithmes mémétiques [Moscato et al., 1989] comme outil pour la modélisation et l'optimisation du système.

Les algorithmes mémétiques sont une classe d'algorithmes évolutionnistes combinant un algorithme génétique avec une stratégie de recherche locale. Toutefois, l'application d'un algorithme mémétique au cas de l'optimisation du changement de voie requiert de prendre en considération certains aspects :

1. le problème est continu [Chelouah et al., 2000]
2. l'évaluation de la performance se fait en temps réel, du fait que la seule façon d'évaluer l'impact de la modification du comportement d'un véhicule et d'observer l'évolution de son score pendant une période donnée
3. l'environnement change continuellement et rapidement [Grefenstette et al., 1992]
4. la taille de la population ne peut être contrôlée
5. la convergence vers un optimum n'est pas requise : notre objectif est d'obtenir un gain sur la qualité du trafic, mais nous ne pouvons pas espérer atteindre la solution optimale

Dans ce cadre, et sur la base de RT-ACC, nous définissons les paramètres de conduite suivants :

- v_t : la vitesse cible
- b_f : la décélération désirée, c'est-à-dire la valeur de décélération que le véhicule ne dépassera pas sauf en cas de danger immédiat
- τ : la valeur de temps inter-véhiculaire de RT-ACC
- Δ_a : le bénéfice minimum requis en termes d'accélération pour réaliser un changement de voie tactique
- b_r : la décélération maximale pouvant être imposée à un nouveau suiveur lors du changement de voie (une valeur de $0m.s^{-2}$ interdit à un véhicule changeant de voie de faire ralentir un nouveau suiveur)
- b_v : la décélération maximale imposée dans le cadre d'un changement de voie coopératif (b_v remplace b_r lors d'une manoeuvre de changement de voie reposant sur la communication)

Le vecteur comportement se définit par $B = (\alpha_\tau, \alpha_{b_f}, \alpha_{\Delta_a}, \alpha_{v_t}, \alpha_{b_r}, \alpha_{b_v}) \in [0.5, 1.5]^6$ où chaque α_X est un coefficient appliqué sur la variable X correspondante.

6.3.2.1/ ALGORITHME

Le principe de l'algorithme est résumé par la figure 6.9. Etant donné que le problème est continu, nous utilisons comme base un algorithme génétique à codage réel (ou Real Coded Genetic Algorithm (RCGA)) [Wright et al., 1991]. A pas fixes, chaque véhicule émet son vecteur comportement et son score aux véhicules alentours. A réception de ces informations, un véhicule peut mettre à jour son propre comportement en fonction des informations reçues en croisant son propre comportement avec les comportements reçus, et en appliquant une opération de mutation sur le résultat. Un algorithme de recherche locale est ensuite appliqué pour modifier le comportement du véhicule. Cette recherche

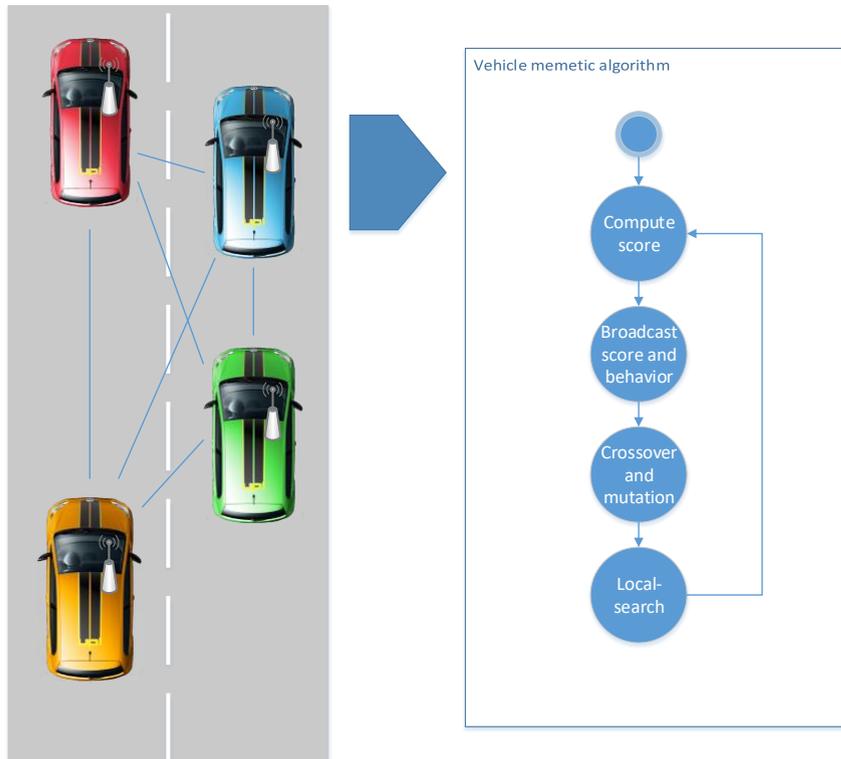


FIGURE 6.9 – Illustration du principe de l'algorithme mémétique

locale consiste en une heuristique destinée à améliorer l'impact du véhicule sur le trafic dans des circonstances spécifiques.

Pour chacun des opérateurs requis nous proposons ci-après des algorithmes. Ces algorithmes permettent de valider la faisabilité et l'intérêt potentiel du cadre proposé, mais ce cadre n'est pas limité à ces seuls algorithmes.

Score Plusieurs fonctions de score sont possibles selon la nature du gain à apporter au trafic. Ici, nous utilisons l'opérateur défini par 6.8.

$$s(v) = \frac{\bar{v}_t - |v - v_t|}{\bar{v}_t} \quad (6.8)$$

L'intervalle de définition de cette fonction est $]-\infty, 1]$ où 1 est le score optimal, c'est-à-dire que le véhicule est à sa vitesse cible. De ce fait, la fonction objectif globale est donnée par 6.9.

$$\text{maximize } Z = \frac{1}{N} \sum_{i=1}^N s_i \quad (6.9)$$

Mutation L'objectif de l'opérateur de mutation est d'éviter l'apparition d'un optimum local en introduisant une part d'aléatoire dans l'évolution de la population. Nous utilisons ici

l'opérateur de mutation défini par l'algorithme 4 où $\alpha_m \in [0, 1]$ est le taux de mutation. Cet opérateur est connu sous le nom *real-number creep* [Davis, 1989], [Herrera et al., 1998] : une valeur aléatoire (distribution uniforme) est générée dans l'intervalle de définition du vecteur de comportement, puis une interpolation linéaire est effectuée entre la valeur actuelle du comportement et la valeur générée avec α_m étant le coefficient d'interpolation.

Algorithm 4 Opérateur de mutation

$$m(B, \alpha_m) = ($$

$$m(\alpha_\tau, \alpha_m)m(\alpha_{b_f}, \alpha_m),$$

$$m(\alpha_{\Delta_a}, \alpha_m), m(\alpha_{v_i}, \alpha_m),$$

$$m(\alpha_{b_r}, \alpha_m), m(\alpha_{b_v}, \alpha_m))$$

$$m(\alpha_i, \alpha_m) = \text{lerp}(\alpha_i, \text{rand}() + 0.5, \alpha_m)$$

Croisement De nombreux opérateurs existent pour les RCGA [Herrera et al., 1998]. Ici, nous faisons l'hypothèse que si un véhicule est dans une situation similaire (proche en distance) et qu'il possède un meilleur score, cela signifie que son comportement est mieux adapté aux conditions de trafic locales. Ainsi, pour privilégier les comportements mieux adaptés, nous utilisons l'opérateur *flat crossover* [Radcliffe, 1991] comme base, mais avec quelques changements :

- l'influence d'un parent est pondérée en fonction de son score
- le croisement n'est pas réalisé par interpolation linéaire entre deux parents, mais par une interpolation multilinéaire entre le vecteur comportement à modifier et les meilleurs comportements

Ainsi, l'algorithme 5 est utilisé comme opérateur de croisement. Chaque voiture sélectionne les 10% meilleurs comportements reçus (d'après leur score), puis calcule un coefficient h_i pour chacun d'entre eux. La valeur de h_i dépend de la différence entre les scores des comportements $s_i - s$: $\lim_{s_i - s \rightarrow -\infty} h_i = \frac{\alpha_0}{2}$ et $\lim_{s_i - s \rightarrow +\infty} h_i = \frac{\alpha_0 + 1}{2}$.

$\alpha_0 \in [0, 1]$ est utilisé afin d'avoir un coefficient non-nul h_i même lorsque $s_i - s$ est faible, et $k \in]0, +\infty[$ est utilisé comme coefficient multiplicateur de la différence des scores.

Algorithm 5 Opérateur de croisement

Get the set $B_{10\%}$ behaviors with 10% best scores

for all (s_i, B_i) in $B_{10\%}$ **do**

$$h_i = \frac{1}{2} \times \left(\alpha_0 + \frac{1}{1 + e^{-k(s_i - s)}} \right)$$

end for

$$B \leftarrow \frac{1}{N} \sum_{i=1}^N B_i h_i + (1 - h_i) B$$

Par application de cet algorithme, la pondération du vecteur initial B est délibérément plus importante que le poids des autres comportements utilisé dans le croisement. Cela permet de rendre les comportements plus stables.

Recherche locale L'application d'un algorithme génétique seul est rapidement limitée dans l'amélioration des conditions de trafic. En effet, celles-ci évoluent rapidement, et de mêmes paramètres peuvent impacter positivement ou négativement le score d'un véhicule selon le contexte, rendant ainsi la convergence difficile.

C'est pourquoi l'algorithme 6 est utilisé. Il s'agit d'une heuristique permettant aux véhicules d'améliorer leur performance.

Algorithm 6 Opérateur de recherche locale

$\{(B_0, s_0), (B_1, s_1), \dots, (B_n, s_n)\}$: Set of received broadcasted behaviors and associated score
 $\bar{s} \leftarrow \frac{1}{n} \sum_{i=0}^n s_i$
if $\bar{s} < s_{self}$ **then**
 $\alpha_{v_i} \leftarrow \alpha_{v_i} + 0.1$
else
 $\alpha_{v_i} \leftarrow \max(1.0, \alpha_{v_i} - 0.05)$
end if

Cet algorithme augmente la vitesse cible du véhicule quand le score des véhicules alentours est plus faible que le score propre du véhicule. Si le score moyen des véhicules alentours est plus élevé que le score des véhicules, l'algorithme restaure la vitesse cible initiale. Cette heuristique est inspirée par l'analyse préalable du problème (voir 6.3.1), mais également par le problème d'éco-résolution [Ferber, 1999] dans le sens où chaque agent applique indirectement une pression aux agents voisins de manière à augmenter leur propre score.

6.3.2.2/ SIMULATION ET RÉSULTATS

Afin d'évaluer l'intérêt et la faisabilité de la solution proposée, nous réalisons des simulations multi-agent. Dans ces simulations, des véhicules sont générés d'un point fixe sur une route à 4 voies à fréquence fixe ($r = 0.25 \text{ veh/s}$), leur vitesse initiale est définie par $v_i = 10 \pm 5 \text{ m.s}^{-1}$ (distribution uniforme), et leur vitesse cible est $v_i = 10 \pm 5 \text{ m.s}^{-1}$ (distribution uniforme). La longueur totale de la section utilisée pour les mesures est $l = 4 \text{ km}$, et le temps total de simulation est $T_{total} = 960 \text{ s}$.

Pour évaluer la performance de la solution, trois mesures sont effectuées :

1. \bar{v} : la vitesse moyenne des véhicules
2. $\overline{\Delta_v}$: la différence moyenne entre la vitesse effective d'un véhicule et sa vitesse cible
3. IF : l'indice de fluidité définit par l'équation 6.10 [Danech-Pajouh, 2001] avec :
 1. q_i : le nombre de véhicules ayant traversé la section i
 2. l_i : la longueur de la section
 3. v_i^f : la vitesse libre sur la section (la vitesse cible moyenne dans notre cas)
 4. v_i : la vitesse effective sur la section

$$10 \times \frac{\sum_{i=1}^n \frac{q_i l_i}{v_i^f}}{\sum_{i=1}^n \frac{q_i l_i}{v_i}} \quad (6.10)$$

Enfin, trois scénarii de simulation sont comparés :

1. Radar seul : aucune coopération n'est utilisée, seules les informations fournies par les radars sont utilisées afin de déterminer si le changement de voie est possible ;

un changement de voie est réalisé dès qu'un véhicule ne peut atteindre sa vitesse cible à cause du véhicule qui le précède et que le changement de voie peut lui permettre d'augmenter sa vitesse (aucun nouveau prédécesseur, ou nouveau prédécesseur suffisamment éloigné), et que le changement de voie est sûr (aucun nouveau suiveur, ou nouveau suiveur n'ayant pas à ralentir suite au changement de voie).

2. CLC : avec changement de voie coopératif par communication ; similaire à la stratégie radar seul, mais la communication est utilisée pour requérir la coopération des véhicules dans la voie cible (selon le principe de l'algorithme 2.
3. CLC-MA : changement de voie coopératif par communication avec optimisation météorique ; similaire à la stratégie CLC, mais le vecteur comportement de chaque véhicule est modifié par application du cadre décrit par la figure 6.9.

Dans les politiques radar seul et CLC le comportement initial du véhicule est B_{id} et est constant. Avec CLC-MA, le comportement initial est défini par la moyenne du comportement des véhicules à proximité.

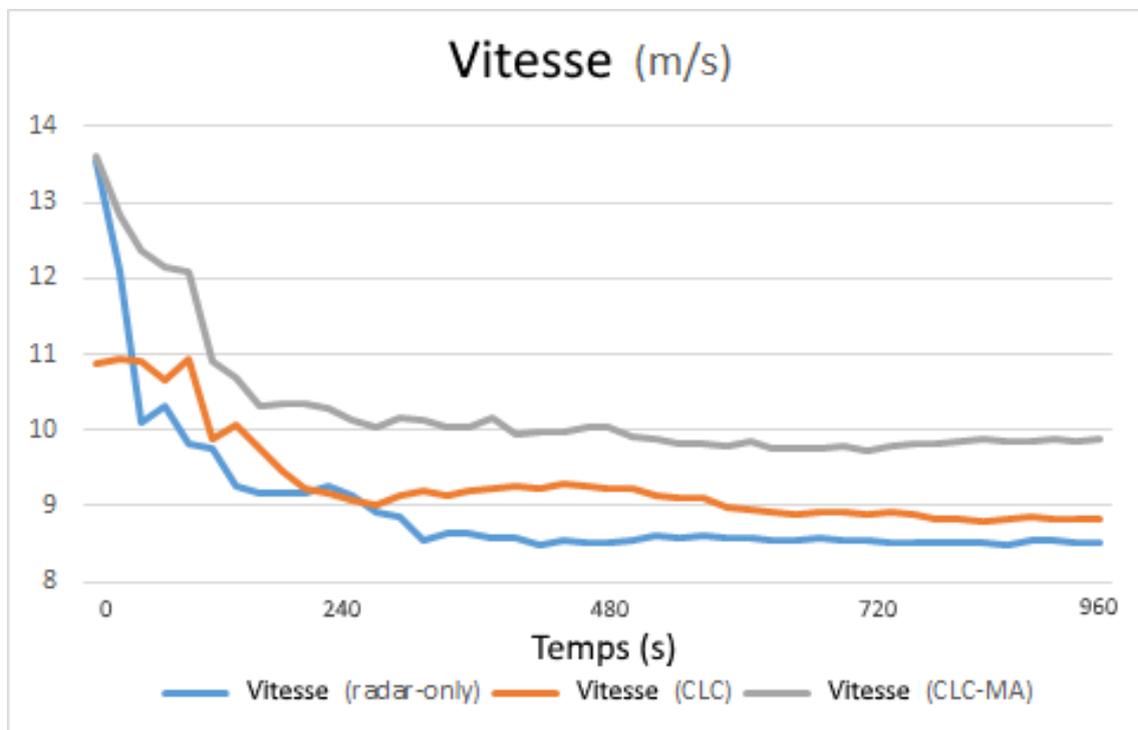


FIGURE 6.10 – Vitesse moyenne des véhicules (en $m.s^{-1}$)

Dans la figure 6.10, nous pouvons observer qu'avec les politiques radar seul et CLC, la vitesse cible ($10 m.s^{-1}$) n'est pas atteinte. Cela s'explique par le nombre important de véhicules dont les vitesses cibles sont en dessous de $10 m.s^{-1}$ et qui ralentissent les véhicules désirant aller plus vite. La vitesse moyenne est relativement élevée au départ car la moyenne est calculée sur un petit échantillon et les véhicules les plus rapides peuvent atteindre leur vitesse cible en dépassant. La vitesse moyenne initiale n'est donc pas significative. Par la suite des véhicules lents commencent à dépasser des véhicules plus lents, occupant ainsi les voies de dépassement et limitant les capacités de dépassement des véhicules les plus rapides. Cela limite la vitesse moyenne du trafic. On constate à la fin (après 480 secondes) une convergence de la vitesse moyenne.

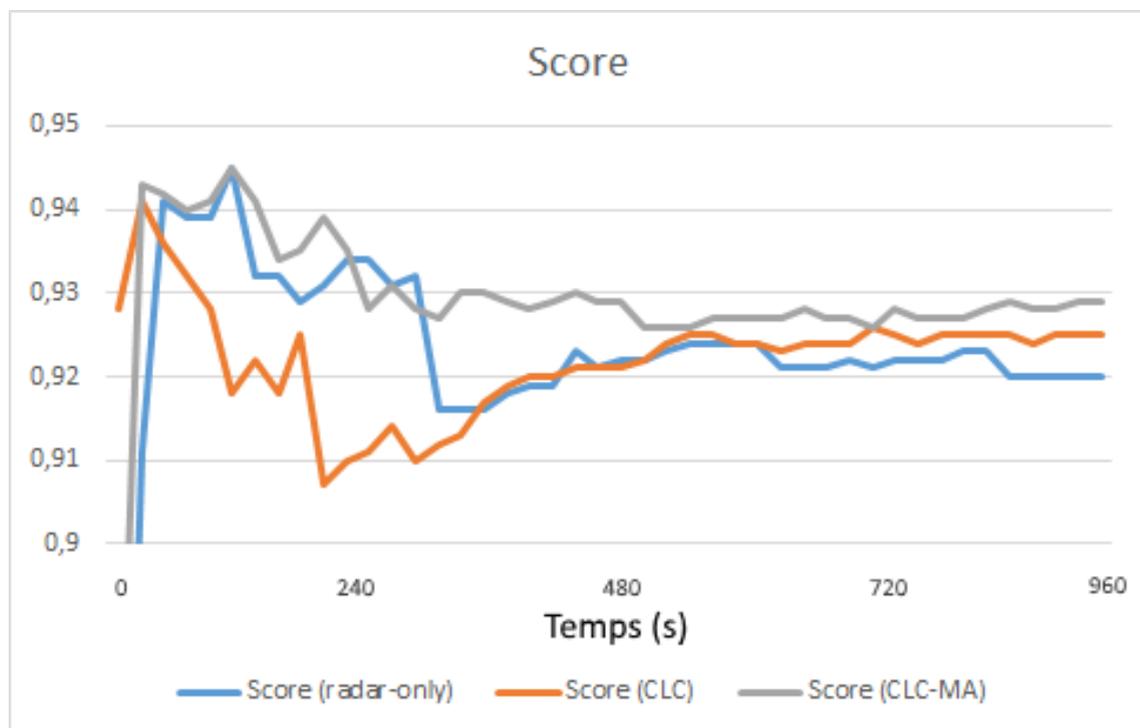


FIGURE 6.11 – Score moyen des véhicules

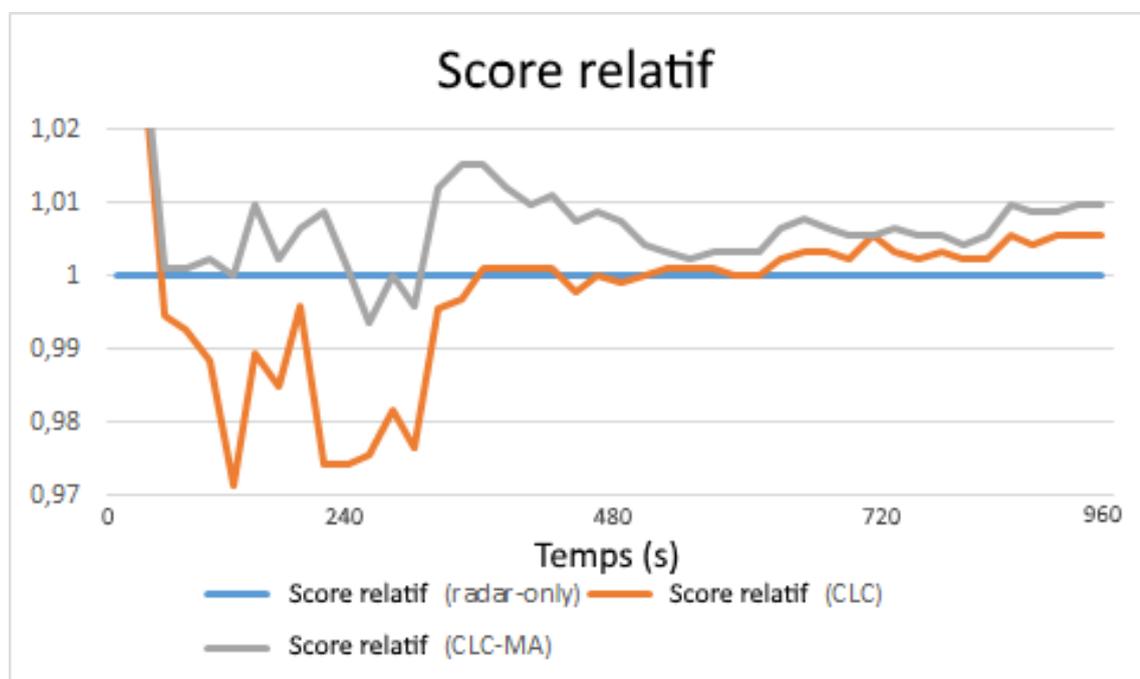


FIGURE 6.12 – Score relatif moyen des véhicules (le score de la politique radar seul étant utilisé comme référence)

La figure 6.10 montre qu'avec la politique CLC-MA, la vitesse moyenne des véhicules est égale à la vitesse cible moyenne ($10m.s^{-1}$).

En comparant le score des véhicules (figure 6.11 et 6.12), nous pouvons faire la même

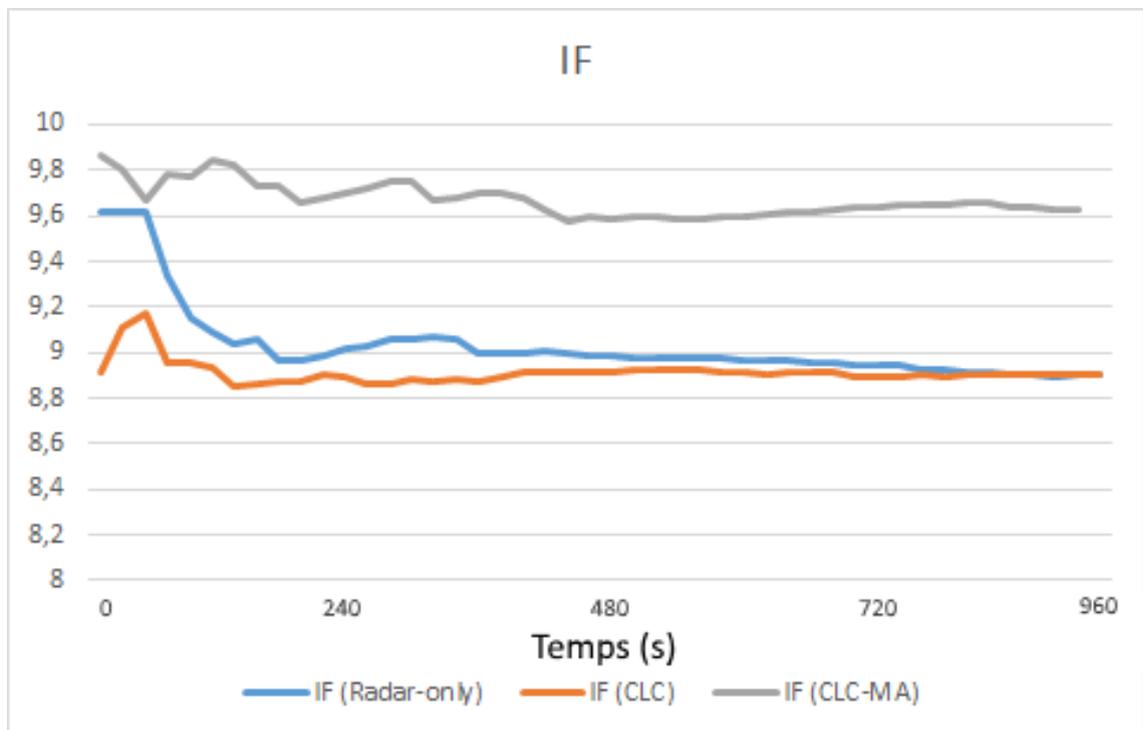


FIGURE 6.13 – IF de la simulation en fonction des différentes politiques

constatation en observant que le score moyen avec CLC-MA est supérieur aux scores moyens des politiques radar seul et CLC. Cette observation est logique compte tenu des mesures effectuées sur la vitesse moyenne, et sachant que le score est lié à l'écart entre vitesse et vitesse cible. Nous pouvons ainsi observer l'efficacité de la stratégie CLC-MA pour l'amélioration du score.

Enfin, CLC-MA montre une amélioration considérable en termes de fluidité avec l'indicateur IF : la figure 6.13 montre une valeur de IF à 8.9 pour les politiques radar seul et CLC, et une valeur de 9.6 pour la politique CLC-MA, soit un gain de 7.8%.

Les tests ont été réalisés pour d'autres valeurs de vitesse cible. La figure 6.14 présente l'IF pour $v_i = v_t = 40 \pm 10 m.s^{-1}$, $r = 0.33 veh/s$. Le gain est plus faible (mais présent) de par le fait que les distances inter-véhiculaires sont plus grandes et que les phases d'accélération et de décélération sont plus longues (à cause de l'augmentation de la vitesse et de la largeur de l'intervalle de vitesses). Cette augmentation du temps passé en phases d'accélération ou de décélération réduit la proportion du temps passé à la vitesse cible.

6.4/ CONCLUSION

Le changement de voie est une forme d'intersection : deux véhicules provenant de voies différentes doivent passer en une même zone, zone qui devient de fait une zone de conflit.

Ainsi, dans ce chapitre nous proposons d'étendre les principes de la régulation coopérative des intersections au cas du changement de voie. Pour cela nous proposons les adaptations nécessaires aux protocoles existants (en l'occurrence TIM) afin de se pas-

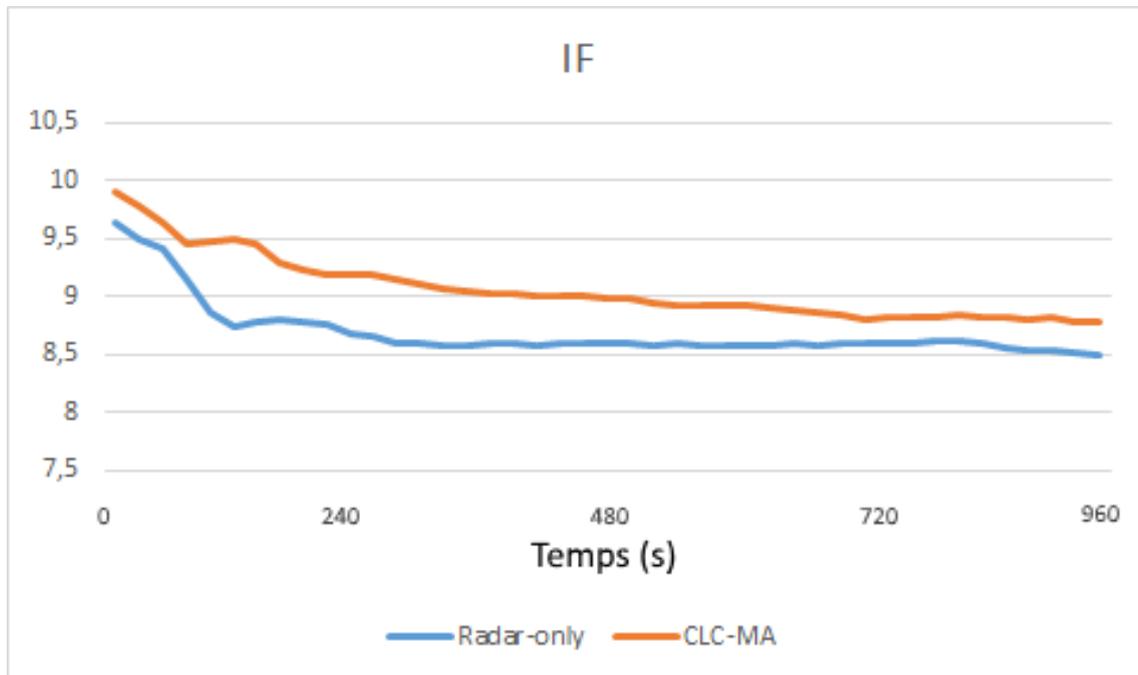


FIGURE 6.14 – IF de la simulation pour une vitesse cible dans l'intervalle $[30, 50]m.s^{-1}$

ser d'un contrôleur d'intersection centralisé. Cette adaptation est rendue possible sous l'hypothèse que les véhicules peuvent se percevoir mutuellement via leurs capteurs, hypothèse raisonnable dans le cas de la conduite sur autoroute. En effet, les véhicules les plus modernes disposent déjà de stratégies de changement de voie automatisé. Nous utilisons ces stratégies comme base pour sécuriser les opérations même en cas d'échec de la communication, et nous utilisons la communication pour optimiser l'opération.

Cette optimisation se fait en deux temps. Dans un premier temps, nous utilisons simplement les bénéfices apportés par TIM, en introduisant le concept de synchronisation de vitesse entre véhicules provenant de voies différentes et en introduisant une forme de courtoisie automatisée entre les véhicules. Bien que cette stratégie n'apporte pas un gain notable à l'échelle globale du trafic (en termes de vitesse moyenne ou de débit), nous pouvons noter une réduction des temps d'arrêt due à la coopération inter-véhiculaire.

Dans un second temps, nous définissons une stratégie d'optimisation du trafic. Le problème étant complexe (système continu, environnement changeant rapidement, information partielle), nous proposons l'application d'un algorithme mémétique. Les véhicules sont définis par un vecteur comportement représentant leur réaction aux conditions environnementales sous la forme de grandeurs physiques, telles que la décélération désirée, l'espace inter-véhiculaire souhaité, la vitesse cible, etc. La communication inter-véhiculaire est alors utilisée pour que chaque véhicule puisse transmettre son vecteur comportement aux véhicules alentour, ainsi que son *score* ayant vocation à montrer l'adéquation d'un comportement aux conditions de trafic locales. En utilisant ces informations, les véhicules peuvent tenter d'améliorer leur propre comportement selon les principes de l'algorithme mémétique (sélection, croisement et mutation).

En conclusion, nous avons pu évaluer la faisabilité d'une telle solution, et estimer son intérêt potentiel en notant des gains intéressants en termes de vitesse moyenne et de fluidité de trafic. Le cadre général étant défini, des travaux restent à faire dans cette voie,

afin d'évaluer et comparer différentes stratégies d'optimisation, notamment via l'application d'autres algorithmes (pour les opérations de sélection, de croisement et de mutation, mais également pour l'heuristique de recherche locale), et la définition d'autres opérateurs de score permettant de choisir la variable du trafic à améliorer.

IV

CONCLUSION ET PERSPECTIVES

CONCLUSION GÉNÉRALE

7.1/ BILAN

Dans ce mémoire, nous proposons une analyse détaillée de l'intersection autonome, intersection de plusieurs véhicules routiers automatisés sans intervention humaine, allant jusqu'à l'implémentation d'un prototype réel. Dans le cadre de cette réalisation, plusieurs contributions sont produites sur différents aspects du système.

Dans un premier temps, nous proposons un modèle d'architecture du système basé sur les principes de l'ingénierie logicielle et de l'ingénierie des systèmes. Ce cadre fournit une base pour l'étude du système et sa simulation.

Ensuite, ce mémoire présente une contribution à la commande du véhicule. D'une part nous réalisons une étude des solutions relatives à la commande longitudinale du véhicule dans le cas spécifique de l'intersection autonome : les véhicules utilisent pour seule source d'information les informations transmises par le contrôleur d'intersection via la communication sans fil. Nous étudions les différentes stratégies de contrôle de l'accélération existante et en choisissons une intégrant dans sa définition la notion de temps de réaction. Par la simulation, et par une expérience avec un véhicule automatisé, nous validons expérimentalement le fonctionnement de cette commande. D'autre part, nous proposons une contribution à la commande latérale du véhicule. Après un état de l'art des stratégies de contrôle latéral des véhicules routiers, nous remarquons que la commande « pure poursuite » est résistante aux problèmes de latences et d'imprécision relatives au positionnement et à l'application de la commande. Celle-ci présente néanmoins quelques défauts sur la qualité du suivi de trajectoire, nous proposons donc une méthode inspirée de la pure poursuite et destinée à pallier ces problèmes. Par la simulation nous montrons que cette méthode améliore fortement la qualité du suivi par rapport aux solutions existantes, même lorsqu'il y a imprécisions et délais.

Par la suite, une étude focalisée sur le protocole de régulation à l'intersection est réalisée. En définissant un ordre de passage des véhicules pour une zone de conflit donnée, nous prenons le risque d'avoir une incohérence entre l'ordre réel des véhicules dans la voie, et l'ordre dans lequel ils doivent franchir l'intersection. Nous montrons que cette situation peut conduire à l'apparition d'un inter-blocage au niveau de l'intersection, puis nous proposons un algorithme permettant de résoudre l'inter-blocage. Cet algorithme devant réordonner la séquence de passage, nous démontrons que cette réorganisation n'introduit pas de risques de collisions (même en cas d'échec de la transmission de la nouvelle séquence aux véhicules perdant leur priorité), et résout effectivement le problème de l'inter-blocage. Par simulation et expérimentation sur robots EV3 nous validons

expérimentalement la faisabilité et les propriétés de l'algorithme.

Pour conclure sur l'intersection autonome, nous réalisons une expérimentation en conditions réelles. Cette expérimentation à trois véhicules, constituant une première mondiale, a été présentée au congrès ITS WC 2015 à Bordeaux. Le chapitre 5 détaille les conditions de réalisation de cette expérimentation. Par cette démonstration, nous avons pu montrer la possibilité de gérer une intersection de véhicules en utilisant la communication sans-fil et sans nécessiter la moindre intervention humaine. Ainsi, grâce à la réalisation d'un prototype en environnement significatif, le Technology Readiness Level (TRL) [Mankins, 1995] de l'intersection autonome passe au niveau 6.

Dans le chapitre 6, dans notre poursuite des travaux sur l'intersection autonome, nous étudions la possibilité d'étendre le système au cas du changement de voie. Nous définissons cette situation comme une forme d'intersection dans laquelle la zone de conflit ne possède pas de position fixe (nous parlons d'intersection mobile), puis nous proposons un protocole inspiré de la régulation coopérative des intersections pour réguler l'opération de changement de voie par l'utilisation de la communication inter-véhiculaire. Nous pouvons ainsi introduire une forme de courtoisie automatisée entre véhicules en permettant à des véhicules de ralentir pour faciliter les changements de voie des autres véhicules. Grâce à ce protocole, nous arrivons déjà en simulation à améliorer les conditions de trafic en diminuant les temps d'arrêt moyens.

De plus, l'Automated Cooperative Lane Change (ACLC) nous permet également de construire un cadre permettant d'optimiser le trafic. Nous proposons ainsi une stratégie inspirée des algorithmes mémétiques afin d'améliorer des indicateurs du trafic choisis. Dans ce cadre, chaque véhicule se voit attribuer un comportement et estime un score interne en fonction de ses objectifs. Les comportements de chacun des véhicules sont transmis aux véhicules voisins qui par opérations de sélection, croisement et mutation, tentent d'améliorer leur propre score. L'objectif est ainsi de pouvoir dynamiquement ajuster le comportement des véhicules en fonction des conditions locale de trafic. Les simulations donnent des résultats prometteurs en montrant un gain sensible de certains indicateurs (en l'occurrence vitesse moyenne et indice de fluidité) par l'application du cadre logiciel ainsi défini.

7.2/ PERSPECTIVES

Cette thèse propose un certain nombre de contributions concernant l'intersection autonome, mais cela reste bien faible au regard des possibilités dans ce domaine.

Premièrement, certains points ont été validés en simulation mais pas en conditions réelles, c'est le cas notamment du dépassement automatisé. Néanmoins, il est ici difficile d'estimer la faisabilité du protocole en conditions réelles en utilisant un faible nombre de véhicules automatisés. C'est pourquoi nos perspectives de recherches incluent la réalisation d'un simulateur en réalité mixte dans lequel la communication pourra être utilisée pour faire interagir des véhicules réels avec un environnement virtuel. De cette façon, les véhicules réels pourront adapter leur accélération et leur freinage vis-à-vis d'obstacles existant uniquement dans l'univers virtuel, et réciproquement des véhicules virtuels pourront utiliser les informations provenant des véhicules réels. L'avantage de cette approche par rapport à la simulation seule est qu'elle limite les erreurs de modélisation : en utilisant du matériel réel, nous nous exposons aux défaillances et limitations de celui-ci (et

qui potentiellement n'ont pas été modélisées) et pouvons mettre en évidence leur impact. Les avantages de cette approche par rapport aux tests réels seuls sont la réduction des coûts liés au matériel (réunir une flotte entière de véhicules automatisés est difficile) et la prévention des risques (tester un freinage d'urgence face à un obstacle virtuel est moins dangereux que face à un obstacle réel). Ces travaux de validation pourront ensuite être étendus afin d'établir une procédure de validation stricte permettant de vérifier qu'un système véhiculaire coopératif peut être déployé en toute sécurité.

Dans un second temps, il est intéressant de noter que la régulation coopérative des intersections n'est actuellement pas prévue par les standards de communication ITS. Le domaine étant émergent, il est normal que différentes stratégies et différents protocoles apparaissent indépendamment. Néanmoins, au fur et à mesure que les principes se concrétisent et se formalisent il devient nécessaire de définir de manière stable les concepts, afin de fournir un cadre solide sur lequel travailler. Cela passe par la standardisation des messages échangés (dans leur format et dans leur nature) et des protocoles d'échanges de ces messages.

BIBLIOGRAPHIE

- [Abbas, 2011] Abbas, M. A. (2011). **Non-linear model predictive control for autonomous vehicles**. PhD thesis, UOIT.
- [Abbas-Turki et al., 2012] Abbas-Turki, A., Ahmane, M., Gao, F., Wu, J., El-Moudni, A., et Miraoui, A. (2012). **Using robots for prototyping autonomous intersection management : Feasibility and feedbacks**. *IFAC Proceedings Volumes*, 45(4) :308–313.
- [AbuAli et al., 2016] AbuAli, N., et Abou-zeid, H. (2016). **Driver behavior modeling : Developments and future directions**. *International Journal of Vehicular Technology*, 2016.
- [Adams et al., 2015] Adams, S., et Rutherford, M. J. (2015). **Decentralized intersection management through peer-to-peer technology**. Dans *Peer-to-Peer Computing (P2P), 2015 IEEE International Conference on*, pages 1–5. IEEE.
- [Ahmed et al., 2013] Ahmed, S. A., Ariffin, S. H., et Faisal, N. (2013). **Overview of wireless access in vehicular environment (wave) protocols and standards**. *Indian Journal of Science and Technology*, 6(7) :4994–5001.
- [Al-Jaroodi et al., 2016] Al-Jaroodi, J., Mohamed, N., Jawhar, I., et Lazarova-Molnar, S. (2016). **Software engineering issues for cyber-physical systems**. Dans *Smart Computing (SMARTCOMP), 2016 IEEE International Conference on*, pages 1–6. IEEE.
- [Anderson, 1978] Anderson, J. E. (1978). **Transit systems theory**.
- [Antonelli et al., 2007] Antonelli, G., Chiaverini, S., et Fusco, G. (2007). **A fuzzy-logic-based approach for mobile robot path tracking**. *IEEE Transactions on Fuzzy Systems*, 15(2) :211–221.
- [Aoyama et al., 1998] Aoyama, M., et others (1998). **New age of software development : How component-based software engineering changes the way of software development**. Dans *1998 International Workshop on CBSE*, pages 1–5. Citeseer.
- [Arpaci-Dusseau et al., 2014] Arpaci-Dusseau, R. H., et Arpaci-Dusseau, A. C. (2014). **Operating systems : Three easy pieces**, volume 151. Arpaci-Dusseau Books Wisconsin.
- [Beck et al., 2001] Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., et others (2001). **Manifesto for agile software development**.
- [Bellet et al., 2011] Bellet, T., Bornard, J.-C., Mayenobe, P., et Saint Pierre, G. (2011). **A computational model for car drivers situation awareness simulation : Cosmo-drive**. Dans *DHM 2011 First International Symposium on Digital Human Modeling*, page 8p.
- [Berger et al., 2012] Berger, C., et Rumpe, B. (2012). **Engineering autonomous driving software**. *Experience from the DARPA Urban Challenge*, pages 243–271.

- [Boer et al., 1998] Boer, E. R., et Hoedemaeker, M. (1998). **Modeling driver behavior with different degrees of automation**. Dans *17 th European Annual Conference on Human Decision Making and Manual Control*.
- [Bokc et al., 2007] Bokc, T., Maurer, M., et Farber, G. (2007). **Validation of the vehicle in the loop (vil) ; a milestone for the simulation of driver assistance systems**. Dans *Intelligent Vehicles Symposium, 2007 IEEE*, pages 612–617. IEEE.
- [Chelouah et al., 2000] Chelouah, R., et Siarry, P. (2000). **A continuous genetic algorithm designed for the global optimization of multimodal functions**. *Journal of Heuristics*, 6(2) :191–213.
- [Chen et al., 2004] Chen, Q., Ozguner, U., et Redmill, K. (2004). **Ohio state university at the 2004 darpa grand challenge : Developing a completely autonomous vehicle**. *IEEE Intelligent Systems*, 19(5) :8–11.
- [Cho et al., 1995] Cho, Y. H., et Kim, J. (1995). **Design of optimal four-wheel steering system**. *Vehicle System Dynamics*, 24(9) :661–682.
- [CITEPA/CORALIE, 2011] CITEPA/CORALIE (2011). **Les chiffres du transport**. Rapport technique.
- [Coffman et al., 1971] Coffman, E. G., Elphick, M., et Shoshani, A. (1971). **System deadlocks**. *ACM Computing Surveys (CSUR)*, 3(2) :67–78.
- [Cohen, 1993] Cohen, S. (1993). **Ingénierie du trafic routier**. *Éléments de théorie du trafic et applications (Presses de l'Ecole National des Ponts et Chaussées)*.
- [Committee, 2014] Committee, S. O.-R. A. V. S. (2014). **Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems**.
- [Coulter, 1992] Coulter, R. C. (1992). **Implementation of the pure pursuit path tracking algorithm**. Rapport technique, DTIC Document.
- [Council, 2000] Council, N. R. (2000). **Highway capacity manual**. Washington, DC.
- [Danech-Pajouh, 2001] Danech-Pajouh, M. (2001). **Road traffic indicators and their prediction**. Dans *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pages 1132–1136. IEEE.
- [Das et al., 2012] Das, A. K., et Asundi, J. (2012). **A simple explicit model approximating the relationship between speed and density of vehicular traffic on urban roads**. *International Journal of Critical Infrastructures* 4, 8(2-3) :195–204.
- [Davis, 1989] Davis, L. (1989). **Adapting operator probabilities in genetic algorithms**. Dans *proc. 3rd International conference on genetic algorithms*, pages 61–69.
- [Dawkins, 2016] Dawkins, R. (2016). **The selfish gene**. Oxford university press.
- [Day et al., 2012] Day, C., Ernst, J., Brennan Jr, T., Chou, C.-S., Hainen, A., Remias, S., Nichols, A., Griggs, B., et Bullock, D. (2012). **Performance measures for adaptive signal control : Case study of system-in-the-loop simulation**. *Transportation Research Record : Journal of the Transportation Research Board*, (2311) :1–15.
- [De La Fortelle et al., 2015] De La Fortelle, A., et Qian, X. (2015). **Autonomous driving at intersections : combining theoretical analysis with practical considerations**. Dans *ITS World Congress 2015*.
- [Delorme, 2001] Delorme, D. (2001). **Modeling driver cognition**. Dans *International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design. Iowa : The University of Iowa Public Policy Center*, pages 276–279.

- [Demmel et al., 2012] Demmel, S., Lambert, A., Gruyer, D., Rakotonirainy, A., et Monacelli, E. (2012). **Empirical iee 802.11 p performance evaluation on test tracks**. Dans *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 837–842. IEEE.
- [Dijkstra, 1982] Dijkstra, E. W. (1982). **The mathematics behind the banker's algorithm**. Dans *Selected Writings on Computing : A Personal Perspective*, pages 308–312. Springer.
- [Dong, 2011] Dong, W. (2011). **An overview of in-vehicle route guidance system**. Dans *Australasian Transport Research Forum*, volume 2011. Citeseer.
- [Dresner et al., 2004] Dresner, K., et Stone, P. (2004). **Multiagent traffic management : A reservation-based intersection control mechanism**. Dans *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 530–537. IEEE Computer Society.
- [Dresner et al., 2008] Dresner, K., et Stone, P. (2008). **A multiagent approach to autonomous intersection management**. *Journal of artificial intelligence research*, 31 :591–656.
- [Eichler, 2007] Eichler, S. (2007). **Performance evaluation of the iee 802.11 p wave communication standard**. Dans *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pages 2199–2203. IEEE.
- [Erdmann, 2014] Erdmann, J. (2014). **Lane-changing model in sumo**. *Proceedings of the SUMO2014 modeling mobility with open data*, 24 :77–88.
- [Falcone, 2007] Falcone, P. (2007). **Nonlinear model predictive control for autonomous vehicles**. PhD thesis, Università del Sannio.
- [Fellendorf et al., 2010] Fellendorf, M., et Vortisch, P. (2010). **Microscopic traffic flow simulator vissim**. Dans *Fundamentals of traffic simulation*, pages 63–93. Springer.
- [Ferber, 1999] Ferber, J. (1999). **Multi-agent systems : an introduction to distributed artificial intelligence**, volume 1. Addison-Wesley Reading.
- [Fok et al., 2012] Fok, C.-L., Hanna, M., Gee, S., Au, T.-C., Stone, P., Julien, C., et Vishwanath, S. (2012). **A platform for evaluating autonomous intersection management policies**. Dans *Proceedings of the 2012 IEEE/ACM Third International Conference on Cyber-Physical Systems*, pages 87–96. IEEE Computer Society.
- [Fu et al., 2006] Fu, L., Sun, D., et Rilett, L. R. (2006). **Heuristic shortest path algorithms for transportation applications : state of the art**. *Computers & Operations Research*, 33(11) :3324–3343.
- [Gamez et al., 2016] Gamez, C., Lombard, A., Ruichek, Y., et Abbas-turki, A. (2016). **Gps-based curve estimation for an adaptive pure pursuit algorithm**. Dans *15th Mexican International Conference on Artificial Intelligence (MICAI 2016)*, Cancún, Mexico.
- [Gipps, 1981] Gipps, P. G. (1981). **A behavioural car-following model for computer simulation**. *Transportation Research Part B : Methodological*, 15(2) :105–111.
- [Grefenstette et al., 1992] Grefenstette, J. J., et others (1992). **Genetic algorithms for changing environments**. Dans *PPSN*, volume 2, pages 137–144.
- [Gregoire et al., 2014] Gregoire, J., Bonnabel, S., et De La Fortelle, A. (2014). **Priority-based coordination of robots**.
- [Guizzo, 2011] Guizzo, E. (2011). **How google's self-driving car works**.
- [Hao et al.,] Hao, X., Abbas-Turki, A., Perronnet, F., et Bouyekhf, R. **V2i-based velocity synchronization at intersection**.

- [Hao et al., 2015] Hao, X., Abbas-Turki, A., Perronnet, F., Bouyekhf, R., et El Moudni, A. (2015). **Intersection management based on v2i velocity synchronization**. Dans *Computational Problems in Science and Engineering*, pages 449–470. Springer.
- [Havender, 1968] Havender, J. W. (1968). **Avoiding deadlock in multitasking systems**. *IBM systems journal*, 7(2) :74–84.
- [Herrera et al., 1998] Herrera, F., Lozano, M., et Verdegay, J. L. (1998). **Tackling real-coded genetic algorithms : Operators and tools for behavioural analysis**. *Artificial intelligence review*, 12(4) :265–319.
- [Huang et al., 2010] Huang, S., et Lin, G. (2010). **Parallel auto-parking of a model vehicle using a self-organizing fuzzy controller**. *Proceedings of the Institution of Mechanical Engineers, Part D : Journal of Automobile Engineering*, 224(8) :997–1012.
- [Ioannou et al., 1993] Ioannou, P., Xu, Z., Eckert, S., Clemons, D., et Sieja, T. (1993). **Intelligent cruise control : theory and experiment**. Dans *Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on*, pages 1885–1890. IEEE.
- [Kent, 2002] Kent, S. (2002). **Model driven engineering**. Dans *International Conference on Integrated Formal Methods*, pages 286–298. Springer.
- [Kesting et al., 2007] Kesting, A., Treiber, M., et Helbing, D. (2007). **General lane-changing model mobil for car-following models**. *Transportation Research Record : Journal of the Transportation Research Board*, (1999) :86–94.
- [Kesting et al., 2010] Kesting, A., Treiber, M., et Helbing, D. (2010). **Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity**. *Philosophical Transactions of the Royal Society of London A : Mathematical, Physical and Engineering Sciences*, 368(1928) :4585–4605.
- [Krauß, 1998] Krauß, S. (1998). **Microscopic modeling of traffic flow : Investigation of collision free vehicle dynamics**.
- [Kreuzen, 2012] Kreuzen, C. (2012). **Cooperative adaptive cruise control : Using information from multiple predecessors in combination with mpc**.
- [Leclercq, 2007] Leclercq, L. (2007). **Hybrid approaches to the solutions of the “lighthill–whitham–richards” model**. *Transportation Research Part B : Methodological*, 41(7) :701–709.
- [Lee et al., 2012] Lee, J., et Park, B. (2012). **Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment**. *IEEE Transactions on Intelligent Transportation Systems*, 13(1) :81–90.
- [Liang et al., 1999] Liang, C.-Y., et Peng, H. (1999). **Optimal adaptive cruise control with guaranteed string stability**. *Vehicle system dynamics*, 32(4-5) :313–330.
- [Linderoth et al., 2008] Linderoth, M., Soltesz, K., et Murray, R. M. (2008). **Nonlinear lateral control strategy for nonholonomic vehicles**. Dans *American Control Conference, 2008*, pages 3219–3224. IEEE.
- [Lombard et al., 2016a] Lombard, A., Hao, X., Abbas-Turki, A., El Moudni, A., Galland, S., et others (2016a). **Lateral control of an unmaned car using gnss positionning in the context of connected vehicles**. *Procedia Computer Science*, 98 :148–155.
- [Lombard et al., 2016b] Lombard, A., Perronnet, F., Abbas-Turki, A., et El Moudni, A. (2016b). **Decentralized management of intersections of automated guided vehicles**. *IFAC-PapersOnLine*, 49(12) :497–502.

- [Lombard et al., 2017] Lombard, A., Perronnet, F., Abbas-Turki, A., et El Moudni, A. (2017). **On the cooperative automatic lane change : Speed synchronization and automatic “courtesy”**. Dans *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*, pages 1655–1658. IEEE.
- [Lombard et al., 2015] Lombard, A., Xuguang, H., Perronnet, F., Abbas-Turki, A., El Moudni, A., et Bouyekhf, R. (2015). **V2x for vehicle speed synchronisation at intersections**. Dans *22nd ITS World Congress*.
- [Makarem et al., 2011] Makarem, L., et Gillet, D. (2011). **Decentralized coordination of autonomous vehicles at intersections**. *IFAC Proceedings Volumes*, 44(1) :13046–13051.
- [Mankins, 1995] Mankins, J. C. (1995). **Technology readiness levels**. *White Paper*, 6(6) :1995.
- [Medina et al., 2017] Medina, A. I. M., van de Wouw, N., et Nijmeijer, H. (2017). **Co-operative intersection control based on virtual platooning**. *IEEE Transactions on Intelligent Transportation Systems*.
- [Meraihi et al., 2008] Meraihi, R., Senouci, S.-M., Meddour, D.-E., et Jerbi, M. (2008). **Vehicle-to-vehicle communications : Applications and perspectives**. *Wireless Ad Hoc and Sensor Networks*, pages 285–308.
- [Michon, 1985] Michon, J. A. (1985). **A critical view of driver behavior models : what do we know, what should we do ?** Dans *Human behavior and traffic safety*, pages 485–524. Springer.
- [Moscato et al., 1989] Moscato, P., et others (1989). **On evolution, search, optimization, genetic algorithms and martial arts : Towards memetic algorithms**. *Caltech concurrent computation program, C3P Report*, 826 :1989.
- [Naumann et al., 1997] Naumann, R., Rasche, R., Tacke, J., et Tahedi, C. (1997). **Validation and simulation of a decentralized intersection collision avoidance algorithm**. Dans *Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on*, pages 818–823. IEEE.
- [Normey-Rico et al., 2001] Normey-Rico, J. E., Alcalá, I., Gómez-Ortega, J., et Camacho, E. F. (2001). **Mobile robot path tracking using a robust pid controller**. *Control Engineering Practice*, 9(11) :1209–1214.
- [North et al., 2006] North, D., et others (2006). **Introducing bdd**. *Better Software, March*.
- [Paquel et al., 2016] Paquel, V., Herbet, J.-B., Albenque, A., Baude, M., Boudour, D., Brouillet, F., Colussi, C., Lavertu, J., Le Moullec, A., Lutinier, B., Mathieu, B., Picard, S., Rageau, F., Wemelbeke, G., et Zoyem, J.-P. (2016). **Les comptes des transports en 2015**. Rapport technique 53, Commission des comptes des transports de la Nation.
- [Payton, 1986] Payton, D. (1986). **An architecture for reflexive autonomous vehicle control**. Dans *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, volume 3, pages 1838–1845. IEEE.
- [Perronnet, 2015] Perronnet, F. (2015). **Régulation coopérative des intersections : protocoles et politiques**.
- [Perronnet et al., 2012] Perronnet, F., Abbas-Turki, A., Buisson, J., El Moudni, A., Zeo, R., et Ahmane, M. (2012). **Cooperative intersection management : real implementation and feasibility study of a sequence based protocol for urban applications**. Dans *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 42–47. IEEE.

- [Perronnet et al., 2013a] Perronnet, F., Abbas-Turki, A., et El Moudni, A. (2013a). **A sequenced-based protocol to manage autonomous vehicles at isolated intersections**. Dans *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 1811–1816. IEEE.
- [Perronnet et al., 2013b] Perronnet, F., Abbas-Turki, A., El-Moudni, A., Buisson, J., et Zéo, R. (2013b). **Cooperative vehicle-actuator system : A sequence-based optimal solution algorithm as tool for evaluating policies**. Dans *Advanced Logistics and Transport (ICALT), 2013 International Conference on*, pages 19–24. IEEE.
- [Qian et al., 2014] Qian, X., Gregoire, J., Moutarde, F., et De La Fortelle, A. (2014). **Priority-based coordination of autonomous and legacy vehicles at intersection**. Dans *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 1166–1171. IEEE.
- [Quinlan et al., 2010] Quinlan, M., Tsz-Chiu, A., Zhu, J., Sturca, N., et Stone, P. (2010). **Bringing simulation to life : A mixed reality autonomous intersection. iros**. Dans *IEEE/RSJ International Conference on*.
- [Radcliffe, 1991] Radcliffe, N. J. (1991). **Equivalence class analysis of genetic algorithms**. *Complex systems*, 5(2) :183–205.
- [Raja et al., 2012] Raja, P., et Pugazhenthii, S. (2012). **Optimal path planning of mobile robots : A review**. *International Journal of Physical Sciences*, 7(9) :1314–1320.
- [Salvucci, 2006] Salvucci, D. D. (2006). **Modeling driver behavior in a cognitive architecture**. *Human Factors : The Journal of the Human Factors and Ergonomics Society*, 48(2) :362–380.
- [Samuel et al., 2016] Samuel, M., Hussein, M., et Mohamad, M. B. (2016). **A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle**. *International Journal of Computer Applications*, 135(1) :35–38.
- [Scharf et al., 1969] Scharf, L., Harthill, W., et Moose, P. (1969). **A comparison of expected flight times for intercept and pure pursuit missiles**. *IEEE Transactions on Aerospace and Electronic Systems*, (4) :672–673.
- [Scheuer et al., 1997] Scheuer, A., et Fraichard, T. (1997). **Continuous-curvature path planning for car-like vehicles**. Dans *Intelligent Robots and Systems, 1997. IROS'97., Proceedings of the 1997 IEEE/RSJ International Conference on*, volume 2, pages 997–1003. IEEE.
- [Schrage et al., 1966] Schrage, L. E., et Miller, L. W. (1966). **The queue m/g/1 with the shortest remaining processing time discipline**. *Operations Research*, 14(4) :670–684.
- [Shahriari et al., 2016] Shahriari, M., et Biglarbegian, M. (2016). **A prioritized collision avoidance methodology for autonomous driving**. Dans *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*, pages 1832–1837. IEEE.
- [Sharon et al., 2015] Sharon, G., Stern, R., Felner, A., et Sturtevant, N. R. (2015). **Conflict-based search for optimal multi-agent pathfinding**. *Artificial Intelligence*, 219 :40–66.
- [Snider et al., 2009] Snider, J. M., et others (2009). **Automatic steering methods for autonomous automobile path tracking**. *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08*.

- [Solis et al., 2011] Solis, C., et Wang, X. (2011). **A study of the characteristics of behaviour driven development**. Dans *Software Engineering and Advanced Applications (SEAA), 2011 37th EUROMICRO Conference on*, pages 383–387. IEEE.
- [Tachet et al., 2016] Tachet, R., Santi, P., Sobolevsky, S., Reyes-Castro, L. I., Frazzoli, E., Helbing, D., et Ratti, C. (2016). **Revisiting street intersections using slot-based systems**. *PloS one*, 11(3) :e0149607.
- [Tattegrain-Veste et al., 1996] Tattegrain-Veste, H., Bellet, T., Pauzié, A., et Chapon, A. (1996). **Computational driver model in transport engineering : Cosmodrive**. *Transportation Research Record : Journal of the Transportation Research Board*, (1550) :1–7.
- [Thrun et al., 2006] Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., et others (2006). **Stanley : The robot that won the darpa grand challenge**. *Journal of field Robotics*, 23(9) :661–692.
- [Treiber et al., 2000] Treiber, M., Hennecke, A., et Helbing, D. (2000). **Congested traffic states in empirical observations and microscopic simulations**. *Physical review E*, 62(2) :1805.
- [Treiber et al., 2013] Treiber, M., et Kesting, A. (2013). **Car-following models based on driving strategies**. Dans *Traffic Flow Dynamics*, pages 181–204. Springer.
- [van Wageningen-Kessels et al., 2015] van Wageningen-Kessels, F., Van Lint, H., Vuik, K., et Hoogendoorn, S. (2015). **Genealogy of traffic flow models**. *EURO Journal on Transportation and Logistics*, 4(4) :445–473.
- [Wallace et al., 1985] Wallace, R. S., Stentz, A., Thorpe, C. E., Moravec, H. P., Whittaker, W., et Kanade, T. (1985). **First results in robot road-following**. Dans *IJCAI*, pages 1089–1095.
- [Webster, 1958] Webster, F. V. (1958). **Traffic signal settings**. Rapport technique.
- [Weiss, 1999] Weiss, G. (1999). **Multiagent systems : a modern approach to distributed artificial intelligence**. MIT press.
- [Wright et al., 1991] Wright, A. H., et others (1991). **Genetic algorithms for real parameter optimization**. *Foundations of genetic algorithms*, 1 :205–218.
- [Wu et al., 2009] Wu, J., Abbas-Turki, A., et El Moudni, A. (2009). **Discrete methods for urban intersection traffic controlling**. Dans *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pages 1–5. IEEE.
- [Wu et al., 2012] Wu, J., Abbas-Turki, A., et El Moudni, A. (2012). **Cooperative driving : an ant colony system for autonomous intersection management**. *Applied Intelligence*, 37(2) :207–222.
- [Zhao et al., 2012] Zhao, P., Chen, J., Song, Y., Tao, X., Xu, T., et Mei, T. (2012). **Design of a control system for an autonomous vehicle based on adaptive-pid**. *International Journal of Advanced Robotic Systems*, 9(2) :44.
- [Zohdy et al., 2012] Zohdy, I. H., et Rakha, H. (2012). **Game theory algorithm for intersection-based cooperative adaptive cruise control (cacc) systems**. Dans *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1097–1102. IEEE.
- [Zohdy et al., 2016] Zohdy, I. H., et Rakha, H. A. (2016). **Intersection management via vehicle connectivity : The intersection cooperative adaptive cruise control system concept**. *Journal of Intelligent Transportation Systems*, 20(1) :17–32.

TABLE DES FIGURES

2.1	Découpage d'une intersection en zones : en rouge la zone de conflit, en orange les zones de stockage, en vert les zones de sortie	8
2.2	Intersection complexe : intuitivement la meilleure séquence consiste à donner le droit de passage aux véhicules gris puis au rouge	12
2.3	Représentation schématique de la communication V2V : les véhicules communiquent directement entre eux, le champ de vision des capteurs frontaux des véhicules est affiché en rouge	13
2.4	Expérimentation du protocole CVA5 en conditions réelles : un smartphone communique en WiFi avec le contrôleur d'intersection pour obtenir le droit de passage individualisé ; l'affichage rouge indique au conducteur qu'il n'est pas autorisé à franchir l'intersection	15
2.5	Echanges de messages entre le contrôleur d'intersection et le véhicule dans le cadre du protocole TIM	16
2.6	Construction de la séquence de passage en fonction des messages reçus - le mouvement 1 vers 2 est supposé en conflit avec le mouvement 3 vers 4	17
2.7	Illustration du fonctionnement du protocole TIM	18
2.8	Illustration de la politique d'évacuation distribuée : le véhicule gris étant proche du véhicule autorisé (vert), il est autorisé avant le véhicule rouge . .	20
2.9	Diagramme de composants global	31
2.10	Diagramme de composants du contrôleur d'intersection	32
2.11	Diagramme de composants du véhicule autonome et coopératif	32
3.1	Composantes du calcul de la distance s_{int} entre deux véhicules	39
3.2	Scénario de simulation de comparaison entre CIDM et RT-ACC	44
3.3	Mesures obtenues en pilotant l'accélération des véhicules via CIDM	44
3.4	Mesures obtenues en pilotant l'accélération des véhicules via RT-ACC . . .	45
3.5	Mesure de la vitesse du véhicule et de la distance à l'obstacle avec application de la commande RT-ACC	45
3.6	Illustration des paramètres de 3.13	49
3.7	Circuit utilisé en simulation pour les résultats présentés	50
3.8	Erreur latérale instantanée dans le cas d'une simulation sans imprécision de positionnement et sans latence	50

3.9	Erreur latérale instantanée dans le cas d'une simulation sans imprécision de positionnement et avec latence sur l'application de la commande	51
3.10	Erreur latérale instantanée dans le cas d'une simulation avec imprécision de positionnement et avec latence sur l'application de la commande	51
3.11	Erreur latérale moyenne sur l'ensemble du tracé (exprimée en mètres) . . .	52
3.12	Erreur relative par rapport à la solution proposée	52
3.13	Tracé GPS du circuit de référence en rouge, et tracés GPS réalisés en utilisant le contrôle latéral autonome	52
3.14	Erreur latérale mesurée lors de l'expérimentation en conditions réelles de la commande latérale	53
3.15	Schéma représentant les « bordures » virtuelles	54
3.16	Vitesse optimale et accélération normale associée en fonction du rayon de courbure pour $e = 5$ (virage constant)	56
3.17	Vitesse optimale et accélération normale associée en fonction du rayon de courbure pour $e = 10$ (virage constant)	56
3.18	Erreur latérale en valeur absolue (moyenne glissante) en fonction de la distance parcourue par le véhicule	57
3.19	Accélération normale (moyenne glissante) en fonction de la distance parcourue par le véhicule	57
4.1	Représentation des types obstacles pour un véhicule	59
4.2	Illustration d'un cas d'inter-blocage : un véhicule est indirectement son propre obstacle	60
4.3	Représentation schématique des agents intervenants dans la distribution des messages dans le cadre de la simulation de la communication sans-fil	62
4.4	Construction de la séquence de passage : l'ordre de passage dans la séquence (indiqué auprès de chaque véhicule) diffère de l'ordre réel des véhicules	63
4.5	Tentative de résolution de l'inter-blocage en changeant la priorité du véhicule mal situé dans la séquence résultant en une possible collision	65
4.6	A : position des véhicules au moment où ils ont émis leur requête d'accès à l'intersection ; B : position réelle des véhicules	65
4.7	Représentation sous forme de graphe de l'intersection : a- avant réception du message du véhicule 10 ; b- après réception du message du véhicule 10	69
4.8	Les quatre cycles du graphe de la figure 4.7	69
4.9	Évacuation des véhicules sans stratégie anti-inter-blocage	70
4.10	Évacuation des véhicules avec stratégie anti-inter-blocage	71
4.11	Prototype d'intersection autonome composée de robots EV3 et d'un smartphone Android	71

5.1	Diagramme illustrant les étapes du processus de développement de l'intersection autonome et les outils associés	74
5.2	Véhicule Xi.cars pilotable en accélération, freinage et direction	76
5.3	Boitier LIDAR IbeoLux	78
5.4	Boitier ProFlex 500	78
5.5	Boitier de communication Commsignia	79
5.6	Illustration schématique de l'architecture matérielle d'un véhicule	81
5.7	Architecture du logiciel contrôleur d'intersection au niveau <i>modules</i>	82
5.8	Architecture du logiciel contrôleur du véhicule au niveau <i>modules</i>	83
5.9	Circuit de référence enregistré en utilisant le GPS RTK	84
5.10	Tests préliminaires de suivi de trajectoire sur le site du congrès ITS de Bordeaux : le véhicule suit une trajectoire prédéfinie et balisée en utilisant uniquement le positionnement satellite	85
5.11	Deux véhicules autonomes négocient leur droit de passage à l'intersection (un troisième véhicule non présent sur l'image se trouve également sur le circuit)	86
6.1	Définitions du changement de voie	92
6.2	Schéma de l'intersection mobile avec découpage en zones ; la distance d est la distance du véhicule désirant changer de voie à la zone de conflit	93
6.3	Architecture du système de dépassement coopératif	95
6.4	Diagramme états-transitions du dépassement coopératif	96
6.5	Scénario de simulation du changement de voie coopératif	99
6.6	Situation initiale du trafic : les véhicules sont disposés de façon aléatoire	101
6.7	Situation optimisée du trafic : les véhicules sont ré-agencés en fonction de leur vitesse cible	101
6.8	Représentation sous forme de graphe de la situation décrite par la figure 6.6 : les chiffres représentent les véhicules ordonnés du plus rapide au plus lent	102
6.9	Illustration du principe de l'algorithme mémétique	104
6.10	Vitesse moyenne des véhicules (en $m.s^{-1}$)	107
6.11	Score moyen des véhicules	108
6.12	Score relatif moyen des véhicules (le score de la politique radar seul étant utilisé comme référence)	108
6.13	IF de la simulation en fonction des différentes politiques	109
6.14	IF de la simulation pour une vitesse cible dans l'intervalle $[30, 50]m.s^{-1}$	110
B.1	Diagramme de classe du module car-control	139

B.2	Diagramme de classe du module communication-core relatif au protocole CVAS	140
B.3	Diagramme de classe du module communication-core relatif au protocole TIM	141
B.4	Diagramme de classe du module cooperative-autonomous-car	141
B.5	Diagramme de classe du module intersection-client relatif au protocole TIM	142
B.6	Diagramme de classe du module intersection-server relatif au protocole TIM	143
B.7	Diagramme de classe du module map-matching	144
B.8	Diagramme de classe du module positioning	145

LISTE DES TABLES

3.1	Paramètres du modèle de Gipps	40
3.2	Paramètres d'IDM	41
3.3	Paramètres de RT-ACC	43
6.1	Mesures utilisant ACLC pour la négociation du changement de voie - 0.25veh/s pour 240s	99
6.2	Mesures utilisant le radar seul pour le changement de voie - 0.25veh/s pour 240s	99
6.3	Comparaison entre notre simulation avec ACLC et VISSIM pour 0,5 veh/s pendant 240s	100

ACRONYMES

- **ABA** : Applied Behavior Analysis. 131
- **ABS** : Antiblockiersystem. 23, 131
- **ACC** : Adaptive Cruise Control. 24, 26, 37, 38, 40, 41, 55, 83, 91, 131
- **ACLC** : Automated Cooperative Lane Change. 116, 131
- **ADAS** : Advanced Driver Assistance Systems. 23, 24, 131
- **AFU** : Assistance au Freinage d'Urgence. 23, 131

- **BDD** : Behavior-Driven Development. 29, 30, 74, 131

- **CACC** : Cooperative Adaptive Cruise Control. 37, 38, 40, 42, 131
- **CAE** : Computer-Aided Engineering. 80, 131
- **CAN** : Controller Area Network. 75–77, 79, 131
- **CBSE** : Component-Based Software Engineering. 28, 131
- **CC** : Cruise Control. 26, 131
- **CDF** : Collision and Deadlock Free. 66–69, 71, 131
- **CIDM** : Cooperative Intelligent Driver Model. 41–44, 46, 127, 131
- **CVAS** : Cooperative Vehicle-Actuator System. 15, 16, 80, 127, 130, 131, 138

- **DCP** : Distributed Clearing Policy. 15, 131
- **D-GPS** : Differential Global Positioning System. 22, 23, 131
- **DSRC** : Dedicated Short-Range Communications. 24, 79, 131

- **EGNOS** : European Geostationary Navigation Overlay Service. 23, 131
- **ESP** : Electronic Stability Program. 23, 131

- **FCFS** : First-come, first-served. 43, 61, 63, 131
- **FIFO** : First in, first out. 131

- **GAGAN** : GPS Aided Geo Augmented Navigation. 23, 131
- **GLONASS** : GLObal NAVigation Satellite System. 22, 131
- **GNSS** : Global Navigation Satellite System. 22, 131
- **GPS** : Global Positioning System. 22, 131

- **HIL** : Hardware-in-the-loop. 6, 29, 131

- **IDM** : Intelligent Driver Model. 41, 42, 46, 93, 131
- **ITS** : Intelligent Transportation Systems. 24, 25, 61, 117, 131, 138

- **LIDAR** : Laser Detection And Ranging. 23, 131

- **MAPF** : Multi-Agent Path-Finding. 101, 131
- **MDE** : Model Driven Engineering. 28, 131
- **MPC** : Model Predictive Control. 40, 41, 47, 131
- **MSAS** : Multi-functional Satellite Augmentation System. 23, 131

- **NHTSA** : National Highway Traffic Safety Administration. 131
- **OBU** : On-board Unit. 131
- **PID** : Proportionnel, Intégrateur, Dérivateur. 39–41, 47, 131
- **POC** : programmation orientée composant. 28, 131
- **POO** : programmation orientée object. 131

- **RADAR** : RAdio Detection And Ranging. 23, 24, 131
- **RCGA** : Real Coded Genetic Algorithm. 103, 105, 131
- **RSU** : Roadside Unit. 5, 131
- **RT-ACC** : Reaction Time - Adaptive Cruise Control. 42–46, 55–57, 99, 103, 127, 131
- **RTK** : Real Time Kinematic. 23, 131

- **SAE** : Society of Automotive Engineers. 21, 131
- **SBAS** : Satellite Based Augmentation System. 22, 23, 131
- **SIL** : Software-in-the-loop. 29, 131
- **SITL** : System-in-the-loop. 29, 131
- **SJN** : Shortest Job Next. 131
- **SLAM** : Simultaneous Localization And Mapping. 23, 131
- **SMA** : système multi-agent. 131
- **SONAR** : SOund Navigation And Ranging. 23, 24, 131
- **STI** : systèmes de transport intelligent. 4, 131
- **SysML** : Systems Modeling Language. 28, 131

- **TIM** : Transparent Intersection Manager. 16, 18, 37, 43, 61, 63, 64, 67, 71, 80, 82, 94, 95, 109, 110, 127, 130, 131, 138–141
- **TIV** : temps inter-véhiculaire. 9, 131
- **TRL** : Technology Readiness Level. 116, 131

- **UDP** : User Datagram Protocol. 80, 131
- **UHF** : Ultra Haute Fréquence. 78, 79, 131
- **UML** : Unified Modeling Language. 28, 131
- **UTBM** : Université de Technologie de Belfort-Montbéliard. 86, 131

- **V2I** : Vehicle-to-Infrastructure. 4, 14, 131
- **V2V** : Vehicle-to-Vehicle. 4, 13, 17, 127, 131
- **V2X** : Vehicle-to-Anything. 21, 38, 131
- **VANET** : Vehicular Ad-Hoc Network. 17, 24, 131
- **VIL** : Vehicle-in-the-loop. 29, 131

- **WAAS** : Wide Area Augmentation System. 22, 131
- **WSMP** : Wave Short Message Protocol. 24, 80, 131

V

ANNEXES

REVUE DE PRESSE DU CONGRÈS ITS-WC 2015

A.1/ ARTICLES DE PRESSE ÉVOQUANT L'EXPÉRIMENTATION MENÉE AU CONGRÈS INTERNATIONAL DES ITS À BORDEAUX EN 2015.

Le véhicule autonome prêt à démarrer

<http://www.lalsace.fr/actualite/2015/06/26/le-vehicule-autonome-pret-a-demarrer>

Véhicule autonome : des premières mondiales prévues à Bordeaux

<https://congresitsbordeaux2015.wordpress.com/2015/06/26/vehicule-autonome-des-premieres-mondiales-prevues-a-bordeaux/>

Bordeaux s'apprête à accueillir un défilé de voitures autonomes

<https://www.industrie-techno.com/bordeaux-s-apprete-a-accueillir-un-defile-de-vehicules-autonomes.39021>

A Bordeaux, les véhicules autonomes se familiarisent avec leur environnement

<http://www.usine-digitale.fr/article/a-bordeaux-les-vehicules-autonomes-se-familiarisent-avec-leur-environnement.N341122>

Bordeaux, terrain de jeu des voitures autonomes

<https://investir.lesechos.fr/actions/actualites/bordeaux-terrain-de-jeu-des-voitures-autonomes-1083868.php>

Le Pôle Véhicule du Futur satisfait d'ITS Bordeaux

<https://congresitsbordeaux2015.wordpress.com/2015/10/19/le-pole-vehicule-du-futur-satisfait-dits-bordeaux/>

Sommet mondial du transport intelligent à Bordeaux

http://www.paysud.com/Sommet-mondial-du-transport-intelligent-a-Bordeaux_a4344.html

Les PME françaises montent dans la voiture autonome

https://www.lesechos.fr/07/10/2015/LesEchos/22039-085-ECH_les-pme-francaises-montent-dans-la-voiture-autonome.htm

Un avenir qui tient la route

http://www.liberation.fr/futurs/2015/11/25/un-avenir-qui-tient-la-route_1416134

B

DIAGRAMMES DE MODULES

B.1/ RÔLE DES MODULES UTILISÉS DANS LE CONTRÔLEUR D'INTERSECTION ET LE CONTRÔLEUR DU VÉHICULE

Cette annexe décrit les différents modules composants les logiciels « contrôleur d'intersection » et « contrôleur du véhicule » développés dans le cadre de la réalisation du prototype d'intersection autonome.

Les modules se divisent en deux catégories :

- Les modules génériques ou dépendances, développés dans le cadre de la plateforme de véhicules autonomes, mais non spécifiquement pour le prototype d'intersection autonome
- Les modules spécifiques développés dans le cadre de la réalisation du prototype d'intersection autonome, c'est-à-dire spécifiquement pour la réalisation de la démonstration ayant eu lieu à ITS WC à Bordeaux

B.2/ CAR-CONTROL

Module spécifique intersection autonome

Module offrant une surcouche au module *renault-scenic-3* afin de fournir une abstraction de la manipulation des commandes du véhicule.

Le diagramme de classes du module est donné par la figure B.1.

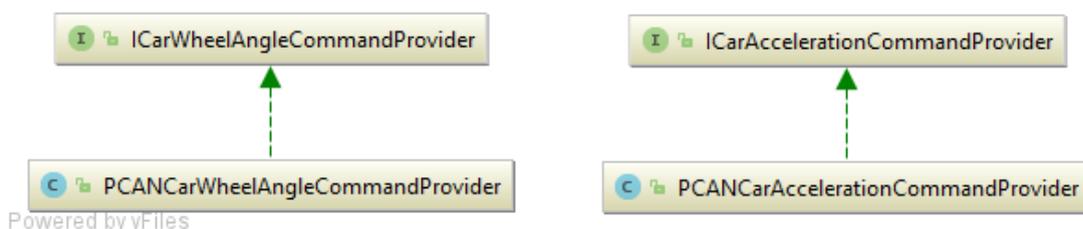


FIGURE B.1 – Diagramme de classe du module car-control

B.3/ COMMSIGNIA-V2X-CLIENT

Module générique / Dépendance

Module permettant de contrôler les boîtiers de communication ITS de Commsignia.

B.4/ COMMUNICATION-CORE

Module spécifique intersection autonome

Module contenant toutes les fonctions relatives à la communication communes à *intersection-client* et *intersection-server*, notamment la définition des messages et de leur types, ainsi que les classes utilitaires permettant de construire ces messages.

Les diagramme de classes du module sont donné par les figure B.2 et B.3, respectivement pour les protocoles CVAS et TIM.

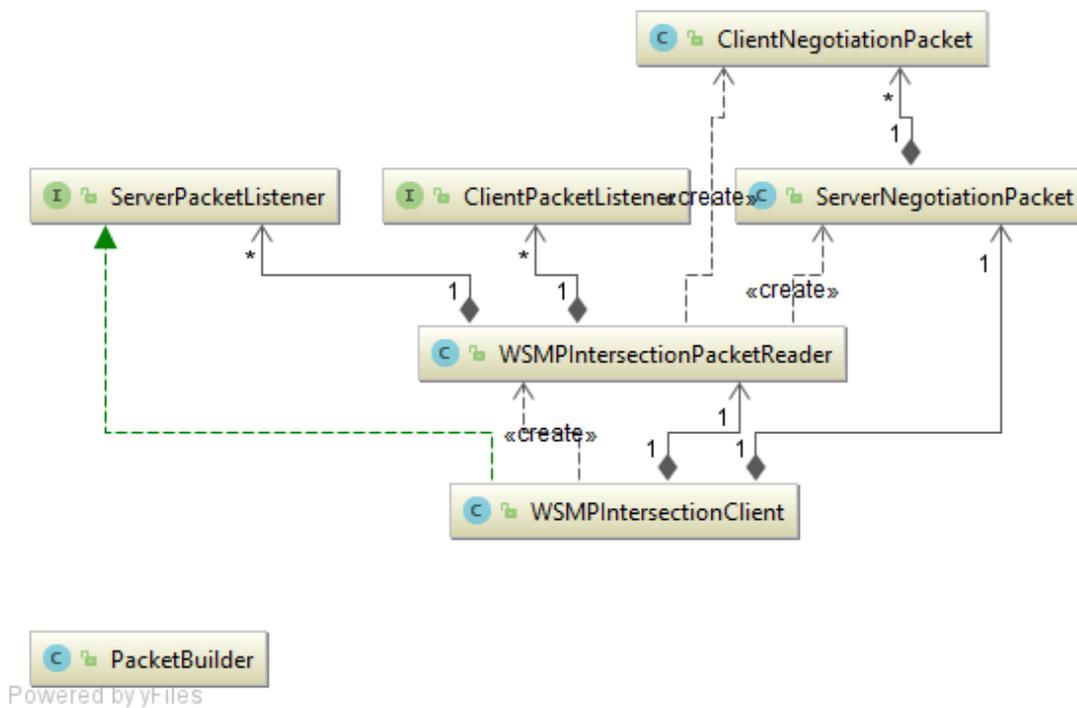


FIGURE B.2 – Diagramme de classe du module communication-core relatif au protocole CVAS

B.5/ COOPERATIVE-AUTONOMOUS-CAR

Module spécifique intersection autonome

Module contenant les classes intégrant la communication et le contrôle du véhicule pour constituer le véhicule autonome et coopératif.

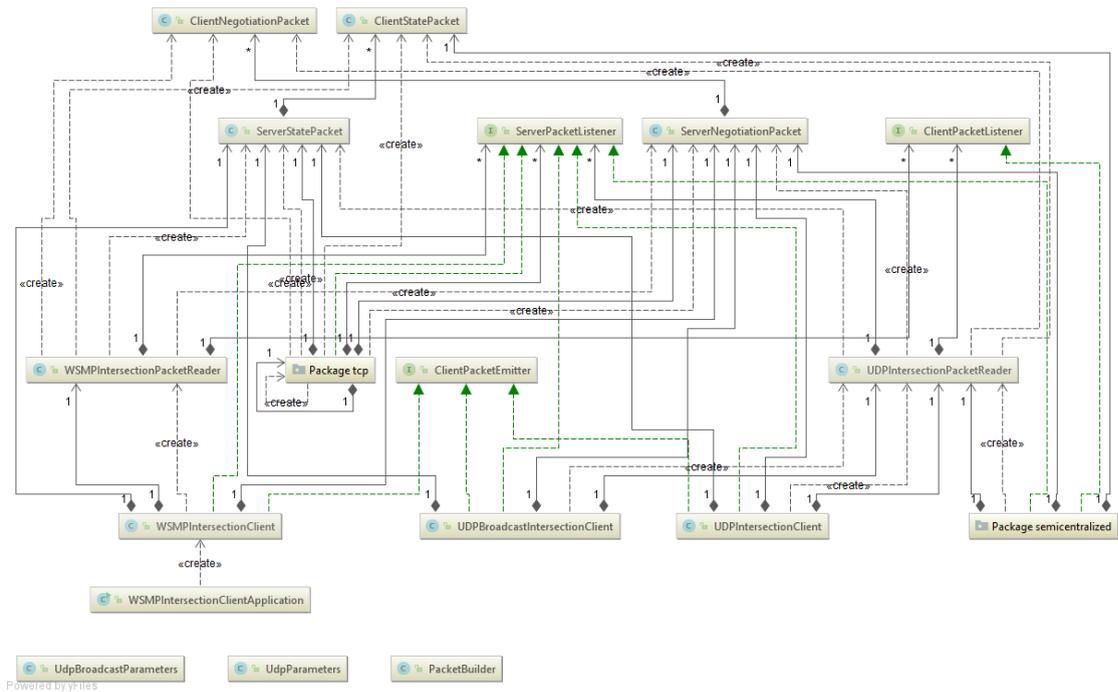


FIGURE B.3 – Diagramme de classe du module communication-core relatif au protocole TIM

Le diagramme de classes du module est donné par la figure B.4.

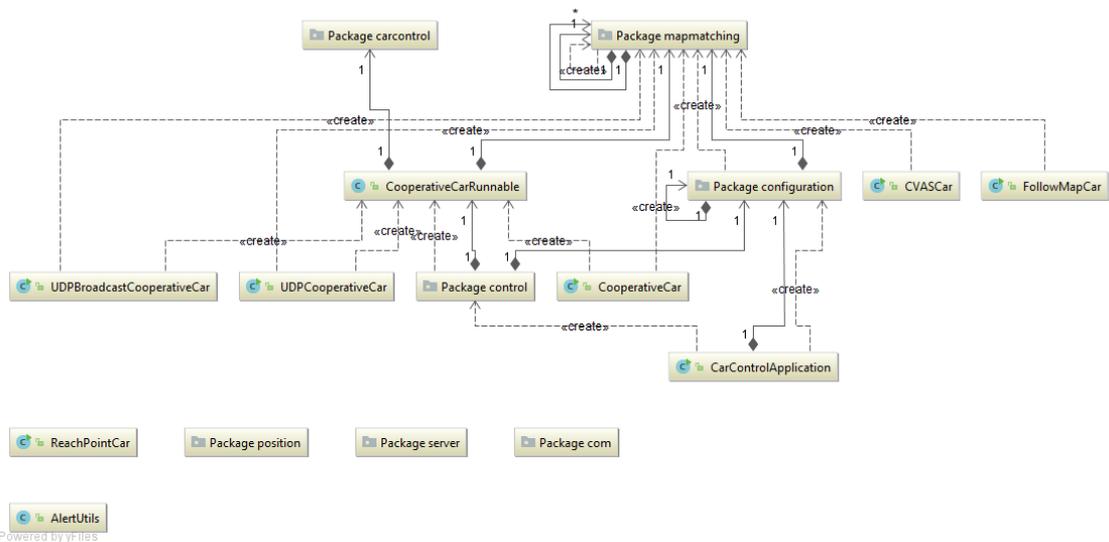


FIGURE B.4 – Diagramme de classe du module cooperative-autonomous-car

B.6/ INTERSECTION-CLIENT

Module spécifique intersection autonome

Module contenant les classes permettant à un client (c'est-à-dire un véhicule) de dialoguer avec l'intersection.

Le diagramme de classes du module est donné par la figure B.5.

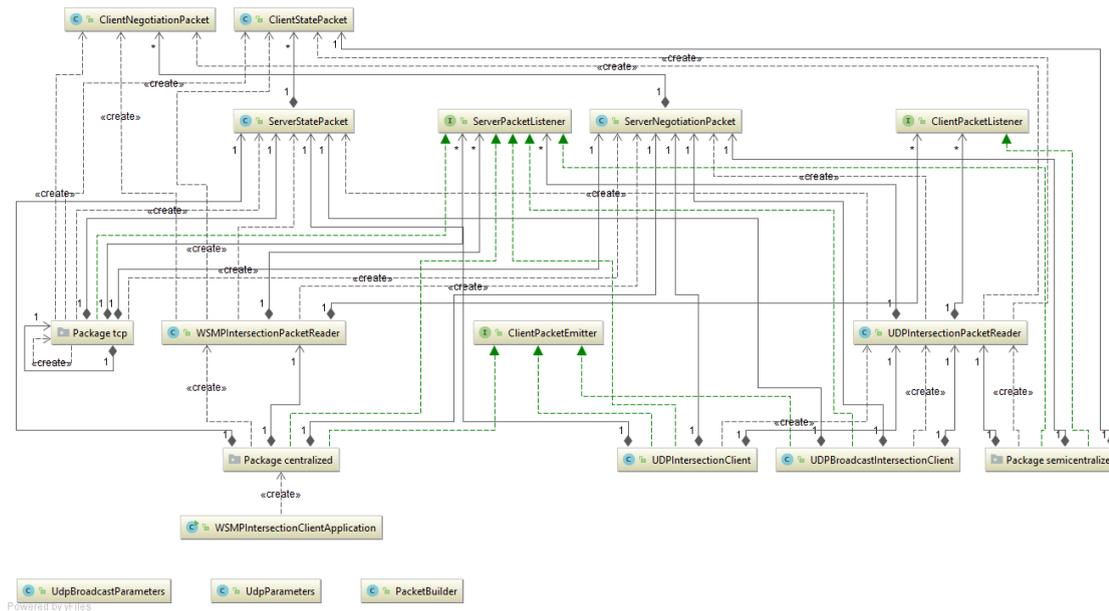


FIGURE B.5 – Diagramme de classe du module intersection-client relatif au protocole TIM

B.7/ INTERSECTION-SERVER

Module spécifique intersection autonome

Module contenant le contrôleur d'intersection, notamment les stratégies de régulation du trafic à l'intersection.

Le diagramme de classes du module est donné par la figure B.6.

B.8/ JSSC

Module générique / Dépendance

Module permettant d'accéder en lecture et en écriture à un port série en Java. Il est notamment utilisé pour configurer et lire les informations du GPS RTK.

B.9/ MAP-MATCHING

Module spécifique intersection autonome

Module contenant les structure de données représentant un circuit, ainsi que les fonctions de map-matching permettant de connaître la position d'un véhicule par rapport à ce

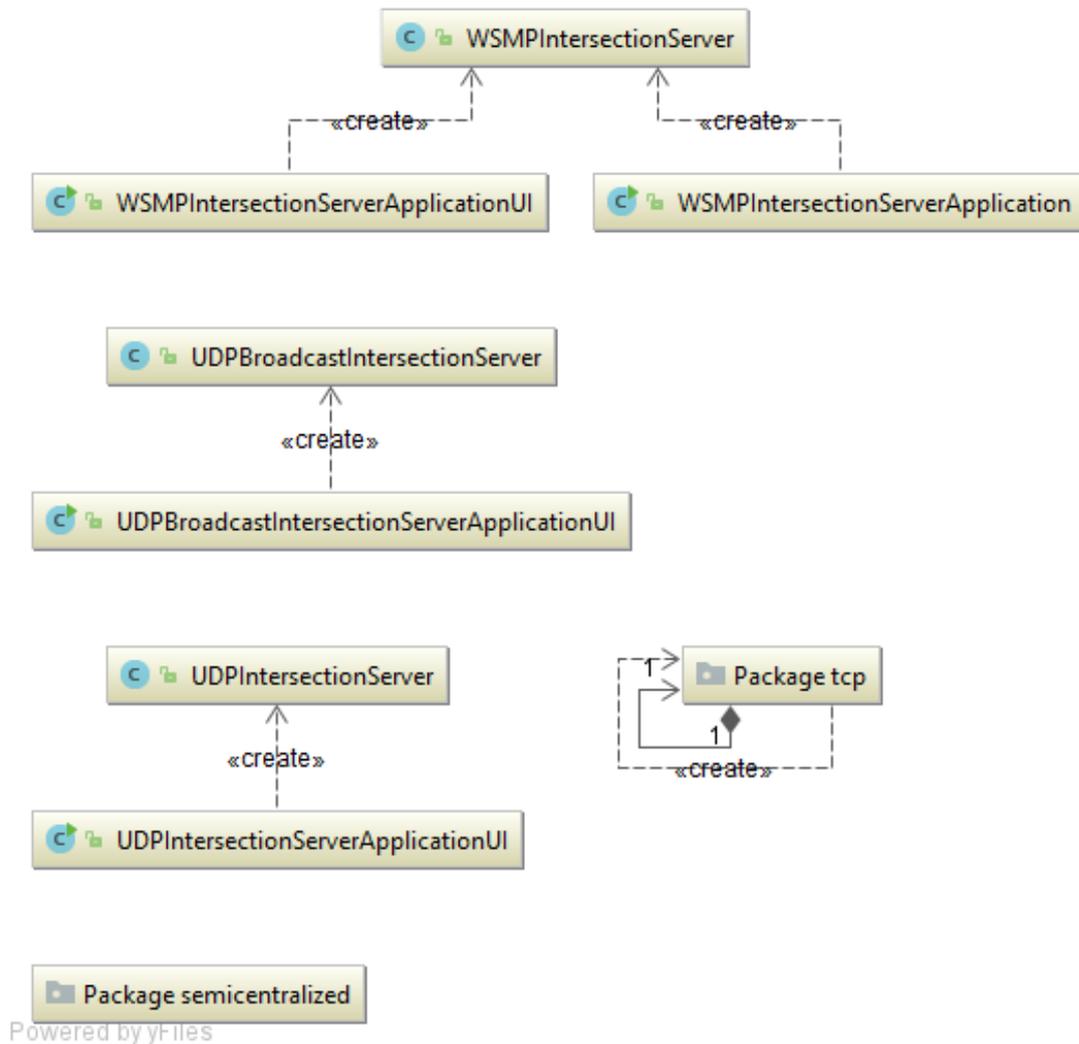


FIGURE B.6 – Diagramme de classe du module intersection-server relatif au protocole TIM

circuit.

Le diagramme de classes du module est donné par la figure B.7.

B.10/ PCAN-USB-WRAPPER

Module permettant d'accéder en lecture et en écriture au dispositif PCAN de Peak Systems en Java, et ainsi d'accéder au bus CAN d'un véhicule.

B.11/ POSITIONING

Module spécifique intersection autonome

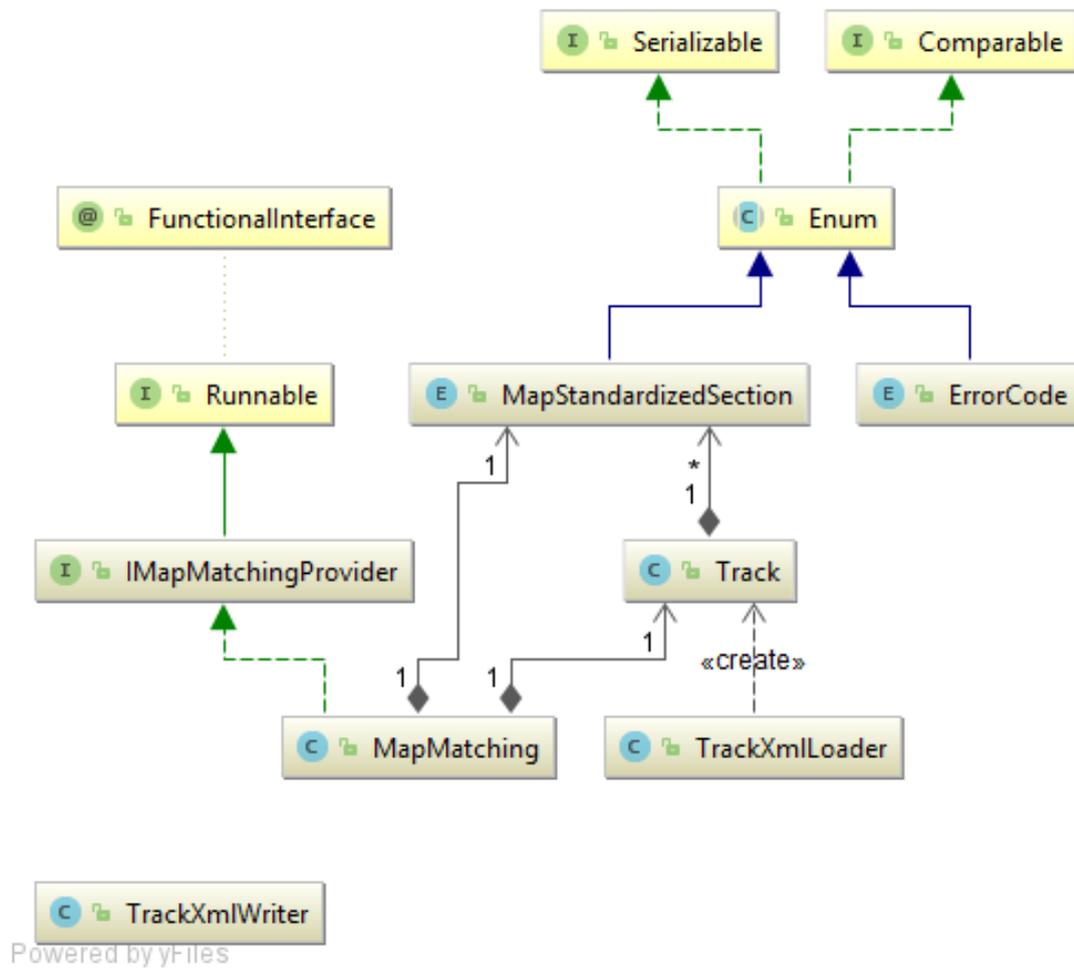


FIGURE B.7 – Diagramme de classe du module map-matching

Module contenant les structures de données représentant la position d'un véhicule, ainsi que des fonctions utilitaires relatives au positionnement satellite et aux systèmes d'information géographique. Ce module fournit également une abstraction des modules d'acquisition de la position, notamment du module *pro-flex-500*.

Le diagramme de classes du module est donné par la figure B.8.

B.12/ PRO-FLEX-500

Module générique / Dépendance

Module permettant de manipuler le GPS RTK ProFlex 500 en Java (configuration et lecture des données).

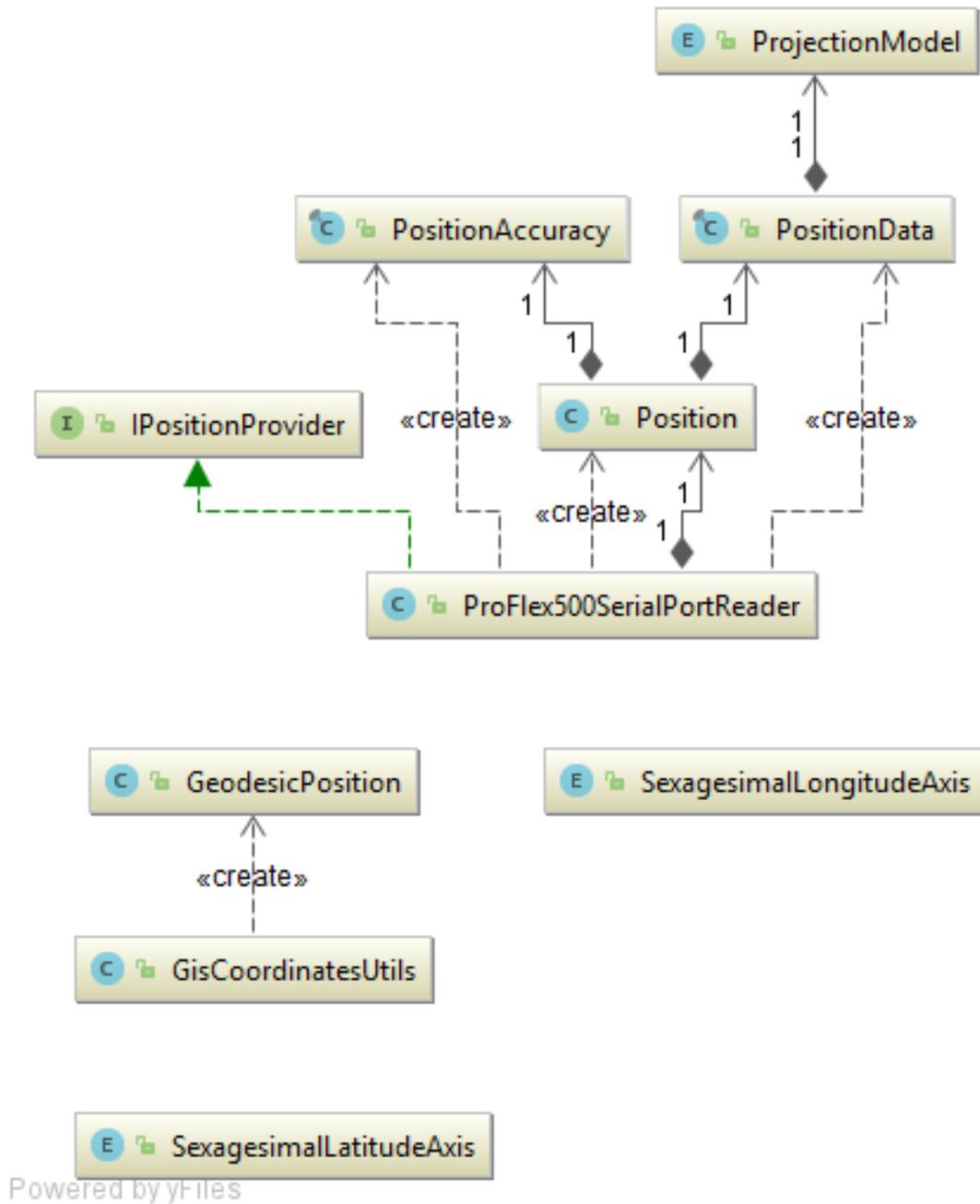


FIGURE B.8 – Diagramme de classe du module positioning

B.13/ RENAULT-SCENIC-3

Module générique / Dépendance

Module accédant au bus CAN des véhicules Renault Scénic 3 et permettant de lire les informations en provenance du véhicule, et également de contrôler celui-ci en envoyant des ordres de contrôle de l'accélération, du freinage et du volant.

C

PROCÉDURES DE LANCEMENT ET D'ARRÊT DE LA DÉMONSTRATION

Cette annexe contient les procédures de lancement et d'arrêt de la démonstration de l'intersection autonome.

148ANNEXE C. PROCÉDURES DE LANCEMENT ET D'ARRÊT DE LA DÉMONSTRATION

Document	Titre	Date de création
ITS-WC-START	Procédure de démarrage	04/10/2015
Révision	Auteur	Date
01	Alexandre Lombard	04/10/2015

Instructions :

- Au siège conducteur, vérifier que les roues sont droites et que la pédale de frein n'est pas enfoncée. Si la pédale de frein est enfoncée, il faut éteindre le calculateur à l'arrière, tirer la pédale de frein vers soi et relancer le calculateur.
- Démarrer la voiture, la mettre sur P avec le frein de parking serré.
- Vérifier l'état du RTK (allumé et "fixed").
- Aller sur l'ordinateur : vérifier qu'il est allumé avec suffisamment de batterie.
- Vérifier le câblage :
 - le GPS/RTK doit être connecté sur le port USB à gauche du PC le plus proche de l'écran
 - le PEAK (connecteur CAN blanc) doit être branché sur le CAN
 - l'antenne Wifi doit être connectée sur le port USB restant
- Vérifier l'état de la connexion Wifi au point d'accès PC-CAR-3_AP.
- Lancer le programme UDPBroadcastCooperativeCar.
- Attendre une dizaine de secondes pour l'apparition des trames GPS (à noter qu'à ce moment des messages en rouges très rapprochés et qui ne s'arrêtent signifient qu'il y a un problème).
- Appuyer sur "Entrée" dans la console pour lancer le programme (les roues vont tourner et la pédale de frein va s'enfoncer).
- Retourner dans le véhicule, attendez le signal pour mettre le levier sur D.
- Dès que chacun a fait signe qu'il était prêt, la démonstration sera lancée.

Document ITS-WC-STOP	Titre Procédure d'arrêt	Date de création 04/10/2015
Révision 01	Auteur Alexandre Lombard	Date 04/10/2015

Instructions :

- Sauf cas exceptionnel, l'arrêt de la démonstration sera déclenché par le serveur, les véhicules s'arrêteront un à un.
- Une fois tous les véhicules à l'arrêt, mettre le frein de parking et mettre le levier sur la position P.
- Sur l'ordinateur, appuyez deux fois sur "Entrée" (attendre 2 secondes entre les deux appuis) : le premier appui enclenche la pédale de frein, le deuxième libère le mode autonome.
- Terminer le programme en appuyant sur le "Stop" rouge à gauche de la console.

Note :

- Ne jamais quitter le poste de conduite avec le levier en position D

ÉTUDE DE LA COMMANDE LATÉRALE

L'objectif de cette annexe est de déduire une valeur à la distance cible de l'équation 3.13 qui s'exprime de la manière suivante :

$$\delta_t = \zeta \tan^{-1} \left(\frac{2E \cos(e_t - \tan^{-1}(\frac{d}{e_d}))}{\sqrt{e_d^2 + d^2}} \right) + \tan^{-1} \left(\frac{2E \sin^{-1} \frac{\kappa}{2}}{L} \right) \quad (\text{D.1})$$

Il est important de rappeler qu'il n'est pas possible, à notre connaissance, d'avoir une expression analytique exacte du mouvement du véhicule. En effet, il n'existe pas, par exemple, d'expression analytique de la distance parcourue sur une ellipse en fonction de l'angle. Ainsi, nous procédons à des simplifications permettant d'approcher le comportement du véhicule sous les conditions suivantes :

1. La courbure induite par la commande $\kappa_\delta \leq 0.25m^{-1}$;
2. La commande vise plutôt à maintenir le véhicule sur la route, à savoir e_d et e_t restent dans une limite raisonnable que nous discuterons par la suite ;
3. Le tracé à suivre est faisable par le véhicule.

Nous pouvons exprimer cette commande sous la forme suivante :

$$\kappa_\delta = \kappa_c + \hat{\kappa} \quad (\text{D.2})$$

Où $\hat{\kappa}$ la courbure du tracé à suivre et κ_c la correction qui s'approche de :

$$\kappa_c \approx 2 \frac{d \sin(e_t) + e_d \cos(e_t)}{e_d^2 + d^2} \quad (\text{D.3})$$

En utilisant l'intégration numérique, la trajectoire du véhicule peut être approchée de la manière suivante :

$$X_{k+1} = \begin{bmatrix} a \\ x \\ y \end{bmatrix}_{k+1} = \begin{bmatrix} a \\ x \\ y \end{bmatrix}_k + s \begin{bmatrix} -\kappa_\delta \\ \cos\left(\alpha_k - \frac{s\kappa_\delta}{\beta}\right) \\ \sin\left(\alpha_k - \frac{s\kappa_\delta}{\beta}\right) \end{bmatrix} \quad (\text{D.4})$$

Avec $\beta \in [1, +\infty[$, α , x , y l'angle du véhicule et ses coordonnées dans un repère cartésien et $s = v\tau$ où v est la vitesse parcourue et τ le temps d'échantillonnage.

En remplaçant κ_δ de (D.2), dans (D.4), nous obtenons :

$$\begin{bmatrix} \hat{\alpha} \\ \hat{x} \\ \hat{y} \end{bmatrix}_{k+1} + \begin{bmatrix} e_t \\ e_x \\ e_y \end{bmatrix}_{k+1} = \begin{bmatrix} \hat{\alpha} \\ \hat{x} \\ \hat{y} \end{bmatrix}_k + \begin{bmatrix} e_t \\ e_x \\ e_y \end{bmatrix}_k + s \begin{bmatrix} -\kappa_c - \hat{\kappa} \\ \cos\left(\hat{\alpha}_k + e_t - \frac{s(\kappa_c + \hat{\kappa})}{\beta}\right) \\ \sin\left(\hat{\alpha}_k + e_t - \frac{s(\kappa_c + \hat{\kappa})}{\beta}\right) \end{bmatrix} \quad (D.5)$$

Où $\hat{\alpha}$, \hat{x} et \hat{y} sont les coordonnées idéales. En développant (D.5) on obtient :

$$\begin{cases} \hat{\alpha}_{k+1} + e_{t_{k+1}} = & \hat{\alpha}_k + e_{t_k} - s\kappa_c - s\hat{\kappa} \\ Y_{k+1} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_{k+1} + \begin{bmatrix} e_x \\ e_y \end{bmatrix}_{k+1} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_k + \begin{bmatrix} e_x \\ e_y \end{bmatrix}_k + sR\left(e_t - \frac{s\kappa_c}{\beta}\right) \begin{bmatrix} \cos\left(\hat{\alpha}_k - \frac{s\hat{\kappa}}{\beta}\right) \\ \sin\left(\hat{\alpha}_k - \frac{s\hat{\kappa}}{\beta}\right) \end{bmatrix} \end{cases} \quad (D.6)$$

Où $R(\Psi)$ est une rotation plane d'angle Ψ . Plus précisément, $R(\Psi) = \begin{bmatrix} \cos(\Psi) & -\sin(\Psi) \\ \sin(\Psi) & \cos(\Psi) \end{bmatrix}$.

De (D.6) et puisqu'il s'agit d'une projection orthogonale, nous déduisons que :

$$\begin{cases} e_{t_{k+1}} = & e_{t_k} - s\kappa_c \\ \begin{bmatrix} e_x \\ e_y \end{bmatrix}_{k+1} = \begin{bmatrix} e_x \\ e_y \end{bmatrix}_k + s\left(R\left(e_t - \frac{s\kappa_c}{\beta}\right) - I\right) \begin{bmatrix} \cos\left(\hat{\alpha}_k - \frac{s\hat{\kappa}}{\beta}\right) \\ \sin\left(\hat{\alpha}_k - \frac{s\hat{\kappa}}{\beta}\right) \end{bmatrix} \end{cases} \quad (D.7)$$

Où I est la matrice identité. Ainsi, comme e_d est une projection orthogonale sur la tangente de la du tracé à suivre, nous avons :

$$\begin{bmatrix} e_t \\ e_d \end{bmatrix}_{k+1} = \begin{bmatrix} e_t \\ e_d \end{bmatrix}_k + s \begin{bmatrix} -\frac{\kappa_c}{\beta} \\ \sin\left(e_{t_k} - \frac{s\kappa_c}{\beta}\right) \end{bmatrix} \quad (D.8)$$

Plus précisément, nous avons :

$$\begin{bmatrix} e_t \\ e_d \end{bmatrix}_{k+1} = \begin{bmatrix} e_t \\ e_d \end{bmatrix}_k + s \begin{bmatrix} -2 \frac{d \sin(e_{t_k}) + e_{d_k} \cos(e_{t_k})}{e_{d_k}^2 + d^2} \\ \sin\left(e_{t_k} - s \frac{2}{\beta} \frac{d \sin(e_{t_k}) + e_{d_k} \cos(e_{t_k})}{e_{d_k}^2 + d^2}\right) \end{bmatrix} \quad (D.9)$$

Sous les hypothèses suivantes :

1. $d \geq 2$ et $d > |e_{d_0}|$;
2. $e_{d_0} \cdot e_{t_0} \leq \frac{\pi}{6}$, à savoir que le véhicule est orienté initialement à l'extérieur du tracé ;
3. $s \leq 5m$;
4. $l = \sqrt{e_{d_k}^2 + d^2}$

Nous pouvons approximer le système (D.9) à l'aide du développement limité d'ordre 2 comme suit :

$$\begin{bmatrix} e_t \\ e_d \end{bmatrix}_{k+1} = \begin{bmatrix} (1 - 2\frac{s}{l}) & -2\frac{s}{\beta} \\ s(1 - \frac{2}{\beta}\frac{s}{l}) & (1 - \frac{2}{\beta}\frac{s^2}{l^2}) \end{bmatrix} \begin{bmatrix} e_t \\ e_d \end{bmatrix}_k + \begin{bmatrix} o(e_t^2) \\ o(e_d^2) \end{bmatrix} \rightarrow \begin{bmatrix} e_t \\ e_d \end{bmatrix}_{k+1} \approx A \begin{bmatrix} e_t \\ e_d \end{bmatrix}_k \quad (D.10)$$

Où par diagonalisation de A nous avons

$$A = Q \begin{bmatrix} 1 - \frac{s}{l} \left(1 + \frac{1}{\beta} \frac{s}{l} + \sqrt{-1 + \frac{2}{\beta} \frac{s}{l} + \frac{1}{\beta^2} \frac{s^2}{l^2}} \right) & 0 \\ 0 & 1 - \frac{s}{l} \left(1 + \frac{1}{\beta} \frac{s}{l} - \sqrt{-1 + \frac{2}{\beta} \frac{s}{l} + \frac{1}{\beta^2} \frac{s^2}{l^2}} \right) \end{bmatrix} Q^{-1} \quad (\text{D.11})$$

Pour assurer la convergence vers 0 du système ^(D.10) nous devons avoir $\| 1 - \frac{s}{l} \left(1 + \frac{1}{\beta} \frac{s}{l} + \sqrt{-1 + \frac{2}{\beta} \frac{s}{l} + \frac{1}{\beta^2} \frac{s^2}{l^2}} \right) \| < 1$, ce qui exige que :

$$l > 2s$$

D'où nous avons posé

$$d > 2s$$

Résumé :

L'augmentation continue du nombre de véhicules a sans cesse conduit à innover pour mieux réguler le trafic routier. Aujourd'hui, l'automatisation des véhicules et la communication inter-véhiculaire rend possible une nouvelle approche : l'intersection autonome et coopérative (IAC) dans laquelle les véhicules autonomes communiquent pour négocier leur droit de passage à l'intersection. Théoriquement plus efficace, plus sûre et plus économique que les approches traditionnelles, son implémentation et son déploiement constituent néanmoins un défi scientifique et technologique. Dans ce mémoire, nous proposons plusieurs contributions soutenant ce développement : une stratégie de contrôle longitudinal et latéral du véhicule conçues pour fonctionner avec des informations discrètes et présentant une latence (provenant de la communication sans-fil ou du positionnement satellite), une solution de prévention des risques d'inter-blocage spécifique à l'IAC, et enfin, l'implémentation et l'expérimentation d'un prototype réel d'IAC avec trois véhicules autonomes. Un travail exploratoire est ensuite fourni pour généraliser les principes de l'IAC, afin d'automatiser et optimiser les opérations de changement de voie et d'insertion.

Mots-clés : Régulation d'intersections, Véhicule autonome, V2X, Interblocage, Optimisation

Abstract:

The continuous increase in the number of vehicles has constantly led to innovation to improve the regulation of road traffic. Today, vehicle automation and inter-vehicular communication make possible a new approach: the autonomous and cooperative intersection (ACI) in which autonomous vehicles communicate to negotiate their right of way at the intersection. Theoretically more effective, safer and more economical than traditional approaches, its implementation and deployment constitute a scientific and technological challenge. In this thesis, we propose several contributions supporting this development: a longitudinal and lateral vehicle control strategy designed to work with information discrete and suffering from latency (coming from wireless communication or satellite positioning), a preventive solution of the risks of deadlock specific to the ACI, and finally the implementation and testing of a real prototype of ACI with three autonomous vehicles. An exploratory work is then provided to generalize the principles of the ACI, in order to automate and optimize the operations of lane change and insertion.

Keywords: Intersection management, Autonomous vehicle, V2X, Deadlock, Optimization

The logo for SPIM (École doctorale SPIM) features a blue horizontal bar on the left, followed by the letters 'S', 'P', 'I', and 'M' in a large, white, sans-serif font.

■ École doctorale SPIM - Université de Technologie Belfort-Montbéliard

F - 90010 Belfort Cedex ■ tél. +33 (0)3 84 58 31 39

■ ed-spim@univ-fcomte.fr ■ www.ed-spim.univ-fcomte.fr

