



HAL
open science

Navigation autonome en milieu urbain en présence d'obstacles mobiles : une approche géométrique

Renaud Poncelet

► **To cite this version:**

Renaud Poncelet. Navigation autonome en milieu urbain en présence d'obstacles mobiles : une approche géométrique. Robotique [cs.RO]. Sorbonne Université, 2022. Français. NNT : 2022SORUS340 . tel-03924438

HAL Id: tel-03924438

<https://theses.hal.science/tel-03924438v1>

Submitted on 5 Jan 2023

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

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

THÈSE DE DOCTORAT
DE SORBONNE UNIVERSITÉ

Préparée à l'Université Pierre et Marie Curie

**Navigation autonome en milieu urbain en présence
d'obstacles mobiles : une approche géométrique**

Soutenue par

Renaud Poncelet

Le 6 Décembre 2022

École doctorale n°130

**École doctorale
Informatique,
Télécommunications
et Électronique (Paris)**

Spécialité

Informatique

Composition du jury :

Paul HONEINE
Professeur, LITIS, Université de Rouen Normandie *Rapporteur*

Olivier ORFILA
Directeur de recherche, Institut VEDECOM *Rapporteur*

Abdelaziz BENSRAHAI
Professeur, INSA de Rouen *Examineur*

Anne SPALANZANI
Maître de conférences, Université Grenoble Alpes *Examinatrice*

Anne VERROUST-BLONDET
Chargée de recherche, Inria Paris *Directrice de thèse*

Fawzi NASHASHIBI
Directeur de recherche, Inria Paris *Co-directeur de thèse*

Remerciements

Entreprendre un doctorat est un long (et même plus long que prévu) cheminement solitaire. On s’y lance souvent avec passion, et c’est cette passion qui permet d’en venir à bout. Pour moi pourtant, cette passion n’existait pas au début. Je voudrais donc d’abord remercier ceux sans qui cette thèse n’aurait pas eu lieu, ceux qui m’ont insufflé la passion. Je remercie Pierre, avec qui j’ai beaucoup appris et ai apprécié travailler. Je remercie aussi mes directeurs Anne et Fawzi qui m’ont guidé dans mes travaux. Ils ont toujours été très présents, nos discussions ont toujours été très enrichissantes et je leur dois d’avoir réussi à finir ma thèse. La passion, c’est une chose, mais comme je l’ai dit, une thèse, c’est long. Heureusement, j’ai pu apprécier mes journées passées à l’Inria grâce à mes collègues que je remercie. J’ai déjà cité Pierre, mais pas encore Pierre. Je remercie aussi Raoul, Luis, Kathia, Anne, Fabio, Maximilian, Dihn, Carlos, Francisco, Fernando, Imane, Kaouther et d’autres que j’oublie tant cela fait longtemps que l’on ne s’est plus vus. Vous avez fait de l’Inria un endroit chaleureux où il a été agréable de travailler. Ayant été confiné pendant une grande partie de la rédaction de cette thèse, elle gardera pour moi une saveur « faite maison ». C’est pourquoi je remercie mes colocataires, transformés pour un temps en collègues de travail. Merci donc à Vadim et à Zob pour leur compagnie. Merci pour les énigmes, la musique, les jeux de société, les sorties sur le parking d’à côté... Merci aussi à Paul avec qui l’on a travaillé sur son jeu de société et fait beaucoup de musique. Merci aussi à tous ceux qui ont été là ces dernières années, Armand, Franzi, Quentin, Mathieu, Manu, Louise, Vicky, Pierre (encore un), Manon, Ilyes, Auriane, Vio, Reb, Coline, Lisa, Elé, Valère. À ceux que j’ai trop peu vus, mais que je remercie d’avoir été là quand l’occasion s’est présentée. Mes grands amis d’enfance Vincent, Daniel, Warren, Ludivine. Merci aussi aux autres que j’oublie ici, mais que je n’oublie pas en vrai (c’est juste que ça commence à être long). Je remercie aussi ma famille qui m’a toujours soutenu dans ma vie et dans mes travaux. Merci Papa, Maman, Zoé et Oscar. Vous êtes les meilleurs parents, frère et sœur que je n’ai jamais eus. Merci à vos partenaires respectifs Ianis et Chibi avec qui l’on a toujours bien rigolé. Merci aussi à Zadig qui m’a été d’une grande aide sur les parties les plus pointues de mes recherches. Merci aussi à ma Mamie et à Ange-Marie. Blagues à part, vous me donnez tous le sourire,

je vous aime. Merci aussi à tous les autres jusqu'au quatrième degré, je vous aime aussi, mais j'économise du papier. J'ai une pensée aussi pour Jacques et Papi qui auraient sûrement aimé me voir aujourd'hui (et qui sait ?) Quoi qu'il en soit, je pense fort à vous. Enfin merci Isaure. Je t'aime et je passe des moments merveilleux avec toi, mais en plus, tu m'as beaucoup aidé lors de ma thèse. Notamment, j'ai trouvé nos petits différends orthographiques très enrichissants.

Table des matières

Tables des figures	viii
Tables des tableaux	ix
1 Introduction	1
1.1 Contexte	3
1.2 Sujet traité et état de l'art	4
1.3 Approche proposée	5
1.4 Organisation du mémoire	5
1.5 Nos publications	6
2 Planification de mouvement dans un environnement dynamique	9
2.1 Problématiques	12
2.1.1 Espace de configuration	12
2.1.2 Reachable Sets (RS)	13
2.1.3 Chemins et trajectoires	13
2.1.4 Obstacles statiques, dynamiques et virtuels	14
2.1.5 Contraintes mécaniques	16
2.1.6 Contraintes et réglementation de la route	17
2.1.7 Perception partielle de l'environnement	17
2.2 État de l'art et approches existantes	19
2.2.1 Planification dans un environnement statique	19
2.2.2 Planification dans un environnement dynamique	27
2.2.3 Planification de mouvement pour les véhicules autonomes en milieu urbain	32
2.2.4 Tableaux récapitulatifs	36
2.3 Conclusion	40
2.3.1 Mode de représentation géométrique dans l'espace de configuration	40
2.3.2 Planification de vitesse le long d'un chemin	41
2.3.3 Planification de chemin	42

2.3.4	Planification en milieu urbain	43
3	Nos approches de planification	44
3.1	Introduction : approche géométrique	47
3.1.1	Espace-temps	47
3.1.2	Décomposition chemin-vitesse	52
3.1.3	Calculs géométriques	56
3.1.4	Conclusion	58
3.2	Planification de chemin à vitesse connue : méthode RIS et MMRIS	58
3.2.1	Définition des RIS (Reachable Interaction Sets)	58
3.2.2	Extensions des RIS pour la robotique	59
3.2.3	Extensions des RIS pour le milieu urbain	67
3.3	Planification de vitesse avec prise en compte des zones occultées : méthode SGSPA	76
3.3.1	Planification de vitesse dans l'espace chemin-temps	77
3.3.2	Prise en compte des zones occultées et des limites de perception	82
3.4	Planification de trajectoire en milieu urbain : méthode GPAD	86
3.4.1	Approche GPAD	87
3.4.2	Architecture	88
3.4.3	Planification de chemin par APF	92
3.4.4	Conclusion et attentes	94
4	Résultats et simulation	95
4.1	Scénarios et résultats de l'approche RIS	98
4.1.1	Implémentation de la méthode	98
4.1.2	Validation expérimentale	101
4.2	Scénarios et résultats de l'approche SGSPA	114
4.2.1	Traversée d'un carrefour en X	114
4.2.2	Traversée d'un carrefour en T avec visibilité limitée	117
4.2.3	Traversée d'un carrefour en X avec dix voitures	118
4.2.4	Conclusion	119
4.3	Scénarios et résultats de l'approche GPAD	120
4.3.1	Présentation de l'environnement de tests	120
4.3.2	Présentation des scénarios et résultats	121
5	Conclusion et perspectives	135
5.1	Approche géométrique	137
5.2	Planification de chemin	137
5.3	Planification de vitesse	138
5.4	Planification en alternant chemin et vitesse	138
5.5	Extensions possibles des approches	138

5.5.1	Extensions possibles de la méthode SGSPA	139
5.5.2	Extensions possibles des méthodes basées sur les RIS	139
5.5.3	Extensions possibles de la méthode GPAD	140
6	Annexes	I
	Glossaire	III
	Notations	V
6.1	Implémentation de la méthode géométrique	VII
6.1.1	Calcul des RS	VII
6.1.2	Occupation des obstacles	X
6.1.3	Implémentation du calcul des RIS et des TP obstacles	XI
6.2	Remplissage d'un rectangle par des disques	XIV
6.3	Implémentation d'un simulateur basé sur CARLA	XVI
	Références	XVII

Table des figures

2.1	Représentation schématique d'un environnement statique	12
2.2	Boîte englobante polygonale	14
2.3	Représentation géométrique de l'environnement statique	15
2.4	Représentation géométrique de l'espace de configuration	16
2.5	Graphique comparatif des niveaux de perception	18
2.6	Somme de Minkowski	20
2.7	Évolution des poids d'un graphe dans l'algorithme de Dijkstra	22
2.8	Représentation graphique d'une méthode par graphe de visibilité	23
2.9	Exemple de l'heuristique du rayon d'exploration	24
2.10	Exemple de recherche de chemin par A*	25
2.11	Représentation graphique d'une méthode par RRT	26
2.12	Représentation graphique d'une méthode par APF	26
2.13	Espace des vitesses dans l'approche dynamic window	29
2.14	Exemple de VO	30
3.1	Planification de chemin dans un environnement statique	49
3.2	Exemple d'orientation d'une trajectoire dans l'espace-temps	51
3.3	Planification dans un environnement dynamique	52
3.4	Planification de vitesse dans un environnement dynamique	54
3.5	Planification de chemin dans un environnement dynamique	55
3.6	Véhicule de Dubins	60
3.7	Maillage obtenu pour la surface de Dubins	61
3.8	Images représentant la probabilité de présence d'un obstacle	63
3.10	Exemples d'occupation d'obstacles	64
3.9	Seuil dynamique	64
3.11	Dessin représentant la marge des RIS	65
3.12	Différents cas d'urgence avec les RIS	67
3.13	Modèle bicyclette	68
3.14	Repère attaché à un chemin	69
3.15	Suivi de chemin par poursuite	70
3.16	Chemins candidats construits à partir des chemins cibles	71

3.17	Surface des RS pour le suivi de chemins candidats avec le modèle bicyclette	71
3.18	Exemple de chemins candidats libres de bout en bout	72
3.19	Exemple de chemins candidats occupés par les RIS	73
3.20	Profils de vitesse pour trois modes d'accélération différents	73
3.21	Représentation de l'élongation d'un obstacle	74
3.22	Schéma chronologique des pas de temps	77
3.23	Schéma des TP obstacles	79
3.24	Calcul de la courbe de la trajectoire	81
3.25	Scénario avec une zone occultée	83
3.26	Exemple où la zone occultée est vide	85
3.27	Présentation schématique de l'architecture de la planification alternative	89
4.2	Maillage des RS entre des couches n'ayant pas le même nombre de points	100
4.1	Maillage des RS entre des couches ayant le même nombre de points	100
4.3	RS dans l'espace-temps et RIS projetés à l'instant $t = 0$	101
4.4	Comparaison dans un environnement "door"	104
4.5	Résultats des approches RIS-path avec obstacles stochastiques . . .	107
4.6	Comparaison du taux de succès des deux versions de RIS-path . . .	108
4.7	Résultats dans "Open" "random"	109
4.8	Résultats dans "Open" "spread"	110
4.9	Résultats dans "Lanes" "random"	111
4.10	Résultats dans "Lanes" "spread"	111
4.11	Résultats dans "Door" "random"	112
4.12	Résultats dans "Door" "spread"	113
4.13	Résultats des approches RIS-path et DRT	113
4.14	Zone occultée vide	115
4.15	Zone occultée avec un véhicule	116
4.16	Carrefour en T avec une visibilité limitée	117
4.17	Carrefour en T : vitesse mesurée	118
4.18	Carrefour en X avec deux autres voitures arrivant à l'intersection .	119
4.19	Image vue du dessus de "Town02"	120
4.20	Scénario où l'ego-véhicule ne rencontre pas d'obstacle sur sa voie. Début de la simulation	122
4.21	Scénario où l'ego-véhicule ne rencontre pas d'obstacle sur sa voie. Arrêt à un feu rouge	123
4.22	Scénario où l'ego-véhicule ne rencontre pas d'obstacle sur sa voie. Le feu est passé au vert et un véhicule est positionné au loin	124
4.23	Scénario avec un véhicule arrêté sur la voie. Début de la simulation	125

TABLE DES FIGURES

4.24	Scénario avec un véhicule arrêté sur la voie. Début du dépassement	126
4.25	Scénario avec un véhicule arrêté sur la voie. Fin du dépassement	127
4.26	Scénario avec deux véhicules arrêtés sur la voie. Début de la simulation	128
4.27	Scénario avec deux véhicules arrêtés sur la voie. Début du dépassement	129
4.28	Scénario avec deux véhicules arrêtés sur la voie. Fin du dépassement	129
4.29	Scénario avec un véhicule arrêté sur la voie et un véhicule arrivant en face. Début de la simulation	130
4.30	Scénario avec un véhicule arrêté sur la voie et un véhicule arrivant en face. Début du dépassement	131
4.31	Scénario avec un véhicule arrêté sur la voie et un véhicule arrivant en face. Fin du dépassement	132
4.32	Scénario avec un véhicule arrivant trop rapidement en face. Début de la simulation	132
4.33	Scénario avec un véhicule arrivant trop rapidement en face. Début du dépassement	133
4.34	Scénario avec un véhicule arrivant trop rapidement en face. Échec du dépassement	133
4.35	Scénario avec un véhicule arrivant trop rapidement en face. Abandon du dépassement	134
6.1	Exemple de remplissage d'un rectangle par des disques	IX
6.2	Figure d'intersection des intervalles	XII
6.3	Polygones d'interpolation	XIII
6.4	Figure sans intersection des intervalles	XIII

Liste des tableaux

2.1	Tableau récapitulatif des approches pour obstacles fixes	37
2.2	Tableau récapitulatif des approches pour obstacles dynamiques . . .	38
2.3	Tableau récapitulatif des approches pour les véhicules autonomes en milieu urbain	39
4.1	Environnement "Open"	102
4.2	Environnement "Lanes"	102
4.3	Environnement "Door"	103
4.4	Scénarios "random"	103
4.5	Scénarios "spread"	103
4.6	Tableau comparatif des résultats de la méthode DRT	106
4.7	Tableau comparatif des résultats moyens de la méthode SES	107
6.1	Tableau des aires en fonction des paramètres	XV

Chapitre 1

Introduction

Table des matières

1.1	Contexte	3
1.2	Sujet traité et état de l'art	4
1.3	Approche proposée	5
1.4	Organisation du mémoire	5
1.5	Nos publications	6

1.1 Contexte

Le développement des transports est central dans le fonctionnement de nos sociétés modernes : les ressources nécessaires à l'épanouissement d'un groupe humain ne se trouvent pas toutes au même endroit. Dans les sociétés nomades, le groupe doit se déplacer pour améliorer son accès aux ressources. Le transport se fait donc vers celles-ci. Avec la sédentarisation, les groupes se spécialisent pour optimiser leur production, se fixant donc petit à petit autour des ressources nécessaires à leur activité. Cette concentration rend l'accès aux biens et aux services inégal et crée une interdépendance entre les groupes. Il devient alors nécessaire de garantir des flux d'échanges entre ceux-ci. Cette tendance à la concentration des activités n'a cessé de s'accélérer, au point que certaines métropoles présentent aujourd'hui des densités telles qu'il devient difficile d'y assurer le bon fonctionnement des transports. Par ailleurs, la densification de certaines zones entraîne la désertification d'autres zones. Apparaissent alors de nouveaux fonctionnements "néo-nomades" : dans les zones désertées, les groupes doivent à nouveau se déplacer vers les ressources. Au contraire, dans les zones denses, la qualité de vie des habitants peut se détériorer, accroissant leur désir d'en sortir régulièrement pour se "ressourcer".

Dans un contexte de demande croissante en transports, le secteur des véhicules autonomes dispose d'un potentiel majeur. Comme le précise l'entreprise [Renault Group, 2020] : *"Le bénéfice numéro un de la voiture autonome est son apport à la sécurité routière. En effet, 94 % des accidents sont causés par une erreur d'appréciation du conducteur. [...] Le véhicule autonome redonnerait par ailleurs mobilité et indépendance à des personnes handicapées (aveugles, sourds...).* Sa généralisation permettrait également d'augmenter fortement la capacité des axes routiers en réduisant l'interdistance entre des véhicules désormais pilotés par l'électronique. L'écoulement plus fluide du trafic permettrait au passage de réduire la consommation énergétique. Enfin, la voiture autonome ouvrirait la voie à des « robotaxis » partagés, sachant qu'un véhicule personnel passe 95 % de son temps... à l'arrêt."

Plus spécifiquement, nous pensons que des véhicules à niveau d'autonomie important (tel que défini par le rapport du [Comité d'ORAD, 2021]) permettraient de répondre au défi du "dernier kilomètre". Dans le réseau de transports, il représente en effet la partie la plus coûteuse du trajet. À l'échelle nationale, il pèse environ 20% du trafic, occupe 30% de la voirie et se trouve être à l'origine de 25% des émissions de gaz à effet de serre d'après [Rungis marché international, 2016]. Ce coût découle pour partie du manque de fluidité du trafic sur cette dernière portion de trajet et le recours aux véhicules autonomes pourrait constituer une réponse à ce problème : réduire l'importance des prises de décisions humaines faciliterait la fluidification du trafic. Celui-ci est en effet fortement dégradé par le caractère souvent erratique des décisions humaines. Les décisions individuelles

des conducteurs cherchant à améliorer leurs temps de trajet (changement de voie, dépassement, choix de son itinéraire) nuisent à la fluidité globale du trafic. Ce phénomène s'apparente au paradoxe de Braess [Braess, 1968] ou au "prix de l'anarchie" [Koutsoupias et Papadimitriou, 2009] (même s'il serait plus correct de parler de prix de l'anomie). En contraignant, voire supprimant, les décisions d'une grande partie des conducteurs, le trafic pourrait s'améliorer. L'accroissement de la part des véhicules autonomes dans la flotte automobile totale améliorerait la prévisibilité du comportement des agents : l'utilisation systématique des véhicules autonomes en remplacement des véhicules classiques limiterait donc le coût du dernier kilomètre. Par ailleurs, cette augmentation de la part des véhicules autonomes favoriserait le développement d'infrastructures adaptées, générant un cercle vertueux dans lequel l'intégration des véhicules autonomes serait de plus en plus aisée.

Pour autant, la multiplication des véhicules autonomes dans le milieu urbain n'est pas sans inconvénient. La Commission indique dans une communication récente [COM 789 final, 2020] que "*[l]a croissance du secteur des transports doit être subordonnée à l'écologisation de la mobilité. En Europe, la mobilité devrait reposer sur un système de transport multimodal efficace et interconnecté, tant pour les passagers que pour le fret*".

Cet objectif de multiplication et de diversification des différentes solutions de mobilité peut sembler incompatible avec une augmentation massive du nombre de véhicules autonomes. Elle devra se structurer autour d'une réelle prise en compte de toutes les catégories d'usagers de la route et nous pouvons espérer voir, en plus des piétons, de plus en plus de cyclistes, navettes, trams, etc. se parcourir nos villes. Pour les zones rurales, le problème est différent. "*[I]l est essentiel que la mobilité soit disponible et abordable pour tous, que les régions rurales et isolées soient mieux connectées*" ([COM 789 final, 2020]). Cela devra aussi se faire de manière multimodale, mais la faible densité de ces zones soulèvera d'autres types de problèmes d'interactions entre les différents moyens de mobilités.

1.2 Sujet traité et état de l'art

L'automatisation des véhicules a fait l'objet de nombreux travaux, dont beaucoup portent sur le milieu urbain. Les environnements urbains sont complexes et certains cas de figures sont encore mal étudiés. En effet, les méthodes de planification existantes prennent peu en compte les environnements où la topologie de la route n'est pas ou mal définie (voies de circulation non matérialisées, marquages au sol et signalisation inexistantes) et où la visibilité est limitée.

1.3 Approche proposée

Cette thèse développe une méthode permettant de définir un mouvement sûr pour un véhicule navigant dans un environnement urbain en présence d'autres véhicules. Ce calcul s'effectue à partir des informations captées par le véhicule autonome ainsi que d'autres informations disponibles sur l'environnement. La méthode doit résoudre des scénarios types rencontrés en milieu urbain.

Les apports de cette thèse portent sur les éléments suivants :

- La prise en compte des aspects incertains du mouvement des agents présents dans l'environnement ;
- La prise en compte des limites des systèmes de perception des véhicules autonomes et des occultations causées par l'environnement ;
- Les cas où la topologie de la route est connue mais est modifiée par un élément inattendu, par exemple par la présence d'un véhicule qui y serait arrêté.

Les travaux de cette thèse ont été menés au sein de l'équipe RITS (Robotics and Intelligent Transportation Systems) de l'Inria.

1.4 Organisation du mémoire

D'abord, nous présentons dans le Chapitre 2 les principales méthodes de planification de mouvement dans un environnement dynamique :

- La section 2.1 expose les problématiques liées au domaine ;
- La section 2.2 présente un état de l'art des recherches sur la planification de mouvement ;
- La section 2.3 conclut sur les méthodes existantes et se positionne par rapport à celles-ci.

Ensuite, nous présentons dans le Chapitre 3 les différentes approches développées au cours de la thèse :

- L'approche géométrique, détaillée dans la section 3.1, sert de fondement théorique à toutes les approches développées dans la thèse.
- La section 3.2 présente une méthode de planification de chemin basée sur l'approche RIS (Reachable Interaction Sets) introduite dans [de Beaucorps *et al.*, 2018]. Celle-ci prend en compte l'aspect incertain de la prévision faite sur le mouvement des agents de l'environnement. Elle est par ailleurs adaptée aux contraintes cinématiques propres au véhicule. Dans cette section, nous présentons aussi la méthode MMRIS (Multi Modal Reachable Interaction Sets) qui calcule différents RIS associés à différents profils de vitesse.
- La section 3.3 présente quant à elle l'approche SGSPA. Celle-ci permet

de résoudre un problème d'adaptation de vitesse en tenant compte des limitations des systèmes de perception ainsi que des occultations dues à l'environnement.

- Enfin, la section 3.4 introduit l'approche GPAD (Geometric Planning with Alternating Decomposition). Celle-ci permet une navigation plus autonome en milieu urbain : elle utilise la méthode MMRIS pour trouver un chemin quand cela est jugé nécessaire et adapte la vitesse le long du chemin avec la méthode SGSPA le reste du temps.

L'évaluation expérimentale des méthodes développées dans la thèse fait l'objet du Chapitre 4.

- La section 4.1 présente l'implémentation de la méthode RIS et de ses améliorations. Elle propose ensuite une validation expérimentale dans des environnements ouverts, c'est-à-dire sans que ne soient définies de chaussée, de voies de circulation ou de signalisation. Nous réalisons différents scénarios dans lesquels l'ego-véhicule (véhicule contrôlé par l'approche) doit se rendre d'un point à un autre de l'environnement en évitant toute collision avec les agents (d'autres véhicules se déplaçant dans l'environnement).
- La section 4.2 présente une évaluation expérimentale de la méthode SGSPA de planification de vitesse pour un ego-véhicule évoluant dans un milieu urbain. Celle-ci présente donc les caractéristiques réalistes de ce genre d'environnement tels qu'une chaussée bien définie, une signalétique, d'autres agents respectant les règles de la route... Cet environnement comporte un bloc opaque en dehors de la chaussée pour simuler l'occultation de la route.
- La section 4.3 valide la méthode combinée GPAD dans un environnement urbain plus complet. Celui-ci comporte ainsi des feux, des véhicules arrêtés sur la voie ou des véhicules ne respectant pas le code de la route.

Finalement, le Chapitre 5 conclut et propose des extensions possibles à nos travaux.

1.5 Nos publications

Ces travaux ont fait l'objet de deux publications :

- P. de Beaucorps, A. Verroust-Blondet, R. Poncelet et F. Nashashibi. "RIS : A framework for motion planning among highly dynamic obstacles." 15th International Conference on Control, Automation, Robotics and Vision, (ICARCV), 2018
- R. Poncelet, A. Verroust-Blondet et F. Nashashibi. "Safe Geometric Speed Planning Approach for Autonomous Driving through Occluded Intersections." 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2020

Une dernière publication est prévue sur l'approche présentée dans la section 3.4.

Chapitre 2

Planification de mouvement dans un environnement dynamique

Table des matières

2.1	Problématiques	12
2.1.1	Espace de configuration	12
2.1.2	Reachable Sets (RS)	13
2.1.3	Chemins et trajectoires	13
2.1.4	Obstacles statiques, dynamiques et virtuels	14
2.1.5	Contraintes mécaniques	16
2.1.6	Contraintes et réglementation de la route	17
2.1.7	Perception partielle de l'environnement	17
2.2	État de l'art et approches existantes	19
2.2.1	Planification dans un environnement statique	19
2.2.1.1	Extension des obstacles fixes dans l'espace de configuration	19
2.2.1.2	Méthodes de recherche de chemin dans un graphe	21
a)	Algorithme de Dijkstra	21
b)	Graphe de visibilité	22
c)	Algorithmes d'exploration	23
d)	A*	24
e)	RRT	24
2.2.1.3	APF	25
2.2.2	Planification dans un environnement dynamique	27
2.2.2.1	Résolution directe	27
a)	APF	27
b)	Intégration des obstacles dynamiques à l'espace de configuration	28
c)	Espace des vitesses	28
d)	Approches combinatoires	29
2.2.2.2	Planification par décomposition	31
a)	Planification avec un chemin prédéterminé	31
b)	Planification avec un profil de vitesse prédéterminé	31

2.2.3	Planification de mouvement pour les véhicules autonomes en milieu urbain	32
2.2.3.1	Contraintes cinématiques	32
2.2.3.2	Prise en compte des limitations des systèmes de perception	33
	a) Filtre à particules	33
	b) Obstacles virtuels	33
	c) POMDP : Partially Observable Markov Decision Process	35
2.2.3.3	Respect des règles de conduite	35
2.2.3.4	Prise en compte de l'aspect incertain des autres agents	36
2.2.4	Tableaux récapitulatifs	36
2.2.4.1	Récapitulatif des approches pour obstacles fixes	36
2.2.4.2	Récapitulatif des approches pour obstacles dynamiques	37
2.2.4.3	Récapitulatif des approches pour les véhicules autonomes en milieu urbain	38
2.3	Conclusion	40
2.3.1	Mode de représentation géométrique dans l'espace de configuration	40
2.3.2	Planification de vitesse le long d'un chemin	41
2.3.3	Planification de chemin	42
2.3.4	Planification en milieu urbain	43

2.1 Problématiques

Notre but est de faire naviguer sans collision un véhicule appelé ego-véhicule d'un lieu de départ vers un lieu d'arrivée dans un environnement urbain où circulent d'autres usagers de la route.

Il s'agit donc de trouver une trajectoire ou un chemin dans un environnement donné, réalisable par l'ego-véhicule, et qui évite les obstacles mobiles et fixes tout en respectant les contraintes de l'environnement.

Avant de présenter les approches existantes, nous détaillons des notions utilisées dans de nombreux travaux traitant de planification de mouvement.

2.1.1 Espace de configuration

La configuration ou l'état de l'ego-véhicule décrit la pose de l'ego-véhicule dans l'espace dans lequel il évolue. Nous considérons que l'état du véhicule est connu à chaque instant de manière précise. En général, nous utilisons la position orientée du centre géométrique de celui-ci. Un véhicule se déplace normalement sur une surface, donc l'espace de travail est \mathbb{R}^2 .

Nous notons X_{ego} l'état de l'ego-véhicule (cf. figure 2.1), avec :

$$X_{ego} = (x_{ego}, y_{ego}, \theta_{ego}) \quad (2.1)$$

où (x_{ego}, y_{ego}) est la position de centre de l'ego-véhicule et θ_{ego} est l'orientation de l'ego-véhicule.

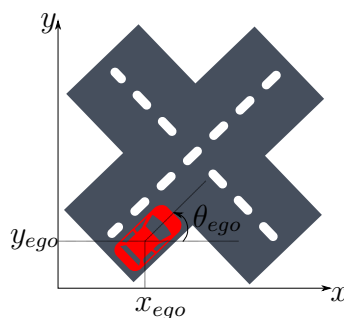


FIGURE 2.1 – Représentation schématique d'un environnement statique. L'ego-véhicule est en rouge, la chaussée en gris et les marquages au sol en blanc. La position orientée de l'ego-véhicule représente l'état de celui-ci.

L'ensemble des configurations possibles d'un véhicule évoluant à partir d'un état initial forme l'espace de recherche appelé espace de configuration. La notion d'espace de configuration pour la planification de mouvement a été introduite par [Lozano-Pérez et Wesley, 1979].

2.1.2 Reachable Sets (RS)

Les Reachable Sets [LaValle, 2006], notés RS dans la suite du manuscrit, sont l'ensemble des configurations atteignables par l'ego-véhicule à partir d'une configuration initiale. Cet espace dépend des contraintes cinématiques s'appliquant à l'ego-véhicule. Ces contraintes peuvent être mécaniques ou induites par l'environnement. En effet, en fonction de ces contraintes, seules certaines configurations sont atteignables. Des contraintes supplémentaires peuvent aussi être ajoutées pour trouver une solution. Quand l'espace de configuration est contraint, le terme de RS de l'ego-véhicule est aussi utilisé.

Notre objectif est de calculer un chemin ou une trajectoire pour l'ego-véhicule qui soit sans collision. Il est donc nécessaire de tenir compte des obstacles mobiles et des obstacles fixes présents dans la scène. L'ensemble des configurations des RS évitant une collision avec des obstacles est appelé espace libre et est noté \mathcal{E} . Le complément de \mathcal{E} dans les RS est l'ensemble des configurations où le robot serait en collision avec un obstacle. C'est ce que nous appellerons par la suite l'espace occupé. Il sera noté \mathcal{O} .

Nous avons donc la relation suivante :

$$\mathcal{O} + \mathcal{E} = RS \quad (2.2)$$

2.1.3 Chemins et trajectoires

Un chemin est une succession de configurations, de l'état initial de l'ego-véhicule à l'état final désiré. Un chemin sera noté \mathcal{P} .

La notion de chemin ne tient pas compte des instants auxquels les configurations sont atteintes. Si toutes les configurations du chemin \mathcal{P}_{ego} suivi par l'ego-véhicule sont dans l'espace libre, le chemin est libre :

$$\mathcal{P}_{ego} \text{ est libre} \iff \forall X_{ego} \in \mathcal{P}_{ego}, X_{ego} \notin \mathcal{O} \quad (2.3)$$

Quand l'environnement contient des obstacles mobiles, il faut intégrer le temps. Nous utilisons alors la notion de trajectoire. Une trajectoire est notée \mathcal{T} et correspond à une succession de configurations dans le temps.

De la même manière, une trajectoire peut être libre ou occupée. La trajectoire est libre si toutes les configurations qui la composent sont libres :

$$\mathcal{T}_{ego} \text{ est libre} \iff \forall X_{ego}(t) \in \mathcal{T}_{ego}, X_{ego}(t) \notin \mathcal{O}(t) \quad (2.4)$$

où $\mathcal{O}(t)$ est l'espace occupé à l'instant t .

Pour savoir si une configuration de l'ego-véhicule est en collision, il faut tenir compte de l'occupation de l'ego-véhicule en fonction de sa configuration, notée $\mathcal{O}_{ego}(X_{ego})$, et de l'occupation des obstacles dans l'environnement.

De nombreuses méthodes, comme celle de [Lozano-Pérez et Wesley, 1979], introduisent une boîte englobante pour effectuer un calcul approché de l'occupation du véhicule ou du robot. L'occupation du véhicule est remplacée par la plus petite boîte rectangulaire contenant entièrement le véhicule. Par exemple, dans la figure 2.2, la boîte englobante de l'ego-véhicule est un rectangle de longueur l_{ego} et de largeur L_{ego} orienté dans le sens du véhicule.

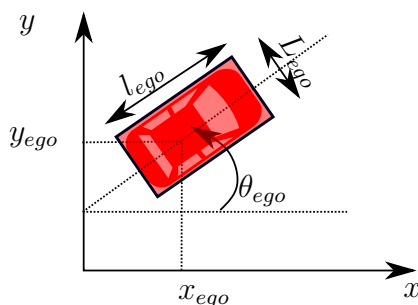


FIGURE 2.2 – Boîte englobante polygonale en rouge pour un véhicule de configuration $X_{ego} = (x_{ego}, y_{ego}, \theta_{ego})$.

2.1.4 Obstacles statiques, dynamiques et virtuels

Nous distinguons deux types d'obstacles dans l'environnement : les obstacles classiques et les obstacles virtuels.

Les obstacles classiques peuvent être statiques ou dynamiques. Ils ne dépendent du temps que dans le second cas.

Les obstacles statiques sont constitués :

- de zones interdites d'accès à l'ego-véhicule comme les bords de routes ou les voies en sens interdits,
- et des obstacles présents sur la chaussée comme des véhicules arrêtés ou des zones de travaux.

Les zones occupées par les obstacles statiques peuvent être connues via une carte détaillée de l'environnement et en utilisant les moyens de perception de l'ego-véhicule en connaissant la position précise de celui-ci sur la carte. Cette étape de perception ne sera pas traitée ici et nous considérons dans la suite de la thèse que l'occupation des obstacles statiques, notée $\mathcal{O}_{statique}$, est parfaitement connue.

Les autres usagers de la route en mouvement dans l'environnement constituent les obstacles dynamiques. Pour connaître l'occupation des autres usagers au cours du temps, il faut pouvoir prédire leurs états futurs. Ceux-ci sont généralement considérés comme étant partiellement connus et dépendent des hypothèses de travail.

Ces hypothèses seront discutées dans la suite de la thèse. À défaut d'indication contraire, nous considérerons que les autres usagers de la route sont des véhicules dont les positions et les vitesses sont connues à chaque instant. Plus précisément, nous considérerons que les véhicules gardent une vitesse constante dans le futur, suivent leur voie de circulation quand celle-ci existe et peuvent aller dans n'importe quelle direction aux embranchements. L'occupation des obstacles dynamiques est notée \mathcal{O}_{others} .

Les obstacles virtuels permettent d'ajouter dans la méthode des obstacles qui n'existent pas. Cela peut être utile pour transformer des règles de la route en obstacle. Par exemple, un feu tricolore, lorsqu'il est rouge, est équivalent à un obstacle statique barrant la voie. Si l'on sait à quel moment le feu repassera au vert, on peut même représenter l'interdiction de manière dynamique en utilisant des obstacles virtuels dynamiques. Nous noterons l'occupation des obstacles virtuels $\mathcal{O}_{virtuel}$.

L'occupation \mathcal{O}_{obs} de l'ensemble des obstacles de l'environnement est alors :

$$\mathcal{O}_{obs} = \mathcal{O}_{statique} \cup \mathcal{O}_{others} \cup \mathcal{O}_{virtuel} \quad (2.5)$$

Pour en déduire l'occupation dans l'espace de configuration, il faut tenir compte, comme dit précédemment, de l'occupation de l'ego-véhicule. Nous avons donc :

$$\forall X_{ego} \in RS, X_{ego} \in \mathcal{E} \iff \mathcal{O}_{ego}(X_{ego}) \cap \mathcal{O}_{obs} = \emptyset \quad (2.6)$$

Effectuer une navigation de l'ego-véhicule sans collision consiste à trouver un chemin libre ou une trajectoire libre selon que le cas est statique ou dynamique.

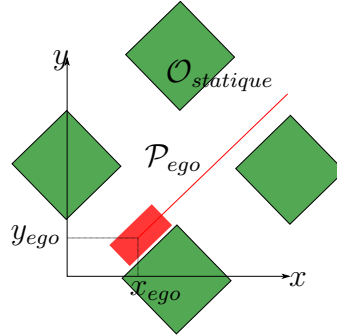


FIGURE 2.3 – Représentation géométrique de l'environnement statique.

Les zones interdites sont représentées par des polygones en vert, l'ego-véhicule par un polygone en rouge et le chemin \mathcal{P}_{ego} est tracé en rouge clair.

Pour calculer l'espace libre, de nombreux travaux considèrent que l'occupation de l'ego-véhicule est réduite à un point et agrandissent l'occupation des obstacles

avec la somme de Minkowski pour respecter l'équation (2.6). La figure 2.4 montre une représentation de l'espace de configuration où l'ego-véhicule de la figure 2.3 a été réduit à un point.

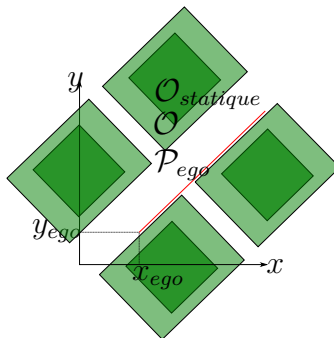


FIGURE 2.4 – Représentation géométrique de l'espace de configuration pour un environnement statique.

Les zones interdites sont représentées par des polygones en vert, l'ego-véhicule est réduit à un point et le chemin \mathcal{P}_{ego} est tracé en rouge clair. Les polygones en vert clair représentent les configurations dans \mathcal{O} .

2.1.5 Contraintes mécaniques

Nous avons vu que les contraintes cinématiques de l'ego-véhicule et les contraintes de l'environnement influent sur les RS. Différents modèles cinématiques peuvent être utilisés pour représenter la cinématique d'un véhicule. Nous considérerons que le modèle est connu et respecté par le véhicule.

Classiquement, nous distinguons les contraintes holonomes et les contraintes non-holonomes.

En utilisant les configurations de l'ego-véhicule $X_{ego}(t)$, les contraintes holonomes peuvent s'écrire :

$$\mathcal{C}_{holo}(X_{ego}(t)) \iff f(X_{ego}(t)) = 0 \quad (2.7)$$

Nous appelons véhicule holonome un véhicule dont le modèle cinématique n'est composé que de contraintes holonomes. Dans la plupart des cas, nous utiliserons le modèle à vitesse constante pour un véhicule holonome. Ce type de véhicule peut se déplacer dans n'importe quelle direction à vitesse constante, peu importe son orientation et sa vitesse initiale. Pour un véhicule allant à une vitesse v_{ego} , ce modèle peut s'écrire comme une contrainte holonome :

$$\mathcal{C}_{holo}(X_{ego}(t)) \iff \frac{dist(X_{ego}(t), X_{ego}(0))}{t} = v_{ego} \quad (2.8)$$

où $dist(X_{ego}(t_1), X_{ego}(t_2))$ est la fonction qui renvoie la distance entre la configuration de l'ego-véhicule à l'instant t_1 et celle à l'instant t_2 .

Les modèles de véhicules holonomes sont surtout utilisés dans les approches liées à la robotique car ils ne correspondent pas à la cinématique de la plupart des véhicules automobiles.

A l'inverse, un exemple simple de véhicule non-holonyme serait un véhicule pouvant aller dans n'importe quelle direction à une vitesse inférieure à une certaine vitesse v_{max} , peu important son orientation et sa vitesse initiale. Nous écririons ce modèle avec la contrainte non-holonyme suivante :

$$\mathcal{C}_{nholo}(X_{ego}(t)) \iff \frac{dist(X_{ego}(t), X_{ego}(0))}{t} \leq v_{max} \quad (2.9)$$

Ces contraintes permettent de représenter plus fidèlement des modèles de véhicules automobiles car ils prennent en compte des aspects mécaniques du véhicule qui jouent sur sa cinématique (l'angle de braquage maximal par exemple) ainsi que d'autres éléments qui jouent sur sa dynamique (comme l'accélération maximale).

2.1.6 Contraintes et réglementation de la route

Nous souhaitons souvent que l'ego-véhicule évoluant dans un environnement urbain ait un comportement proche de celui d'un conducteur interagissant avec les usagers de la route. Ce point fera l'objet d'une discussion dans la suite du manuscrit. Dans un premier temps, nous considérerons qu'une façon d'obtenir un comportement acceptable est de respecter les règles de circulation. Dans le cadre de cette thèse, nous utiliserons la réglementation francophone. Plus précisément, l'objectif est de prendre en compte les règles de priorité, les panneaux de circulation ("limitations de vitesse", "stops", "cédez-le-passage"), les feux de circulation et le respect des distances réglementaires entre les véhicules. Dans la suite de la thèse, nous supposerons que les règles que l'ego-véhicule doit appliquer sont toujours connues et dépendent de sa position sur une carte de l'environnement. Pour cela, la carte devra détailler l'ensemble des voies de circulations et les règles en vigueur sur chaque tronçon, les connections possibles entre les tronçons, la position et l'état des feux à chaque instant.

2.1.7 Perception partielle de l'environnement

Revenons sur l'hypothèse énoncée dans la section 2.1.4 : pour effectuer des prédictions sur les mouvements des autres véhicules, nous avons supposé que l'état courant de tous les autres véhicules présents dans l'environnement était connu. Ceci n'est pas toujours le cas quand nous ne disposons pas de système extérieur à l'ego-véhicule, comme un système de communication par exemple, qui nous

permettrait d'accéder à ces informations. En effet, sans moyen extérieur, seuls les véhicules perçus par l'ego-véhicule peuvent être considérés. Quel que soit le système de perception utilisé par l'ego-véhicule, nous pouvons considérer que certaines zones resteront occultées. Le problème de la prise en compte des zones occultées dans la navigation d'un véhicule a été soulevé en premier lieu dans [Sadou *et al.*, 2004]. Nous pouvons distinguer différents cas de figures (cf. figure 2.5) :

- celui où l'ego-véhicule est **omniscient** : il connaît l'état de tous les autres véhicules à chaque instant, y compris celui des véhicules occultés, soit l'hypothèse faite dans la section 2.1.4. Ceci peut par exemple être rendu possible grâce à un système de communication entre les véhicules ;
- celui où l'ego-véhicule est **clairvoyant** : il ne connaît que les états des véhicules visibles à chaque instant. Toutefois, la carte et les limites de perception de l'ego-véhicule lui permettent de calculer les zones d'où pourraient surgir des véhicules cachés ;
- enfin, celui où l'ego-véhicule est **naïf** : il ne connaît que l'état de tous les véhicules perçus à chaque instant et ignore l'existence de zones occultées d'où pourraient surgir d'autres véhicules.

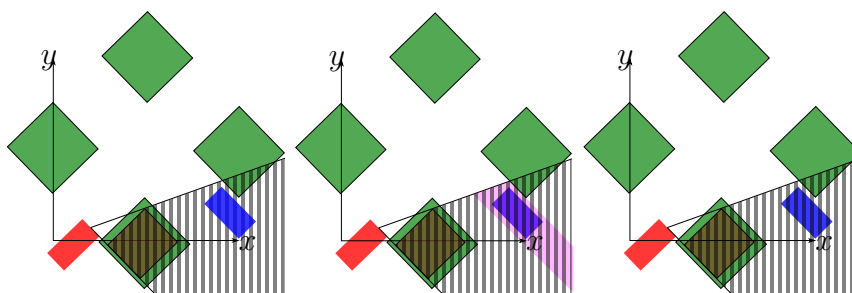


FIGURE 2.5 – Différents niveaux de perception pour l'ego-véhicule.

Nous avons un obstacle masquant une partie de la route pour l'ego-véhicule. L'obstacle est représenté en marron, l'ego-véhicule en rouge et la zone occultée par l'obstacle est hachurée. Un véhicule bleu est situé dans cette zone. À gauche, quand l'ego-véhicule est omniscient, l'état du véhicule bleu est connu de l'ego-véhicule, il n'est donc pas hachuré. Au centre, quand l'ego-véhicule est clairvoyant, le véhicule bleu est hachuré car il est occulté et son état n'est pas connu de l'ego-véhicule. Cependant, la zone violette où pourrait se situer un véhicule caché est connue de l'ego-véhicule grâce à la connaissance d'une carte des environs. Enfin, à droite, quand l'ego-véhicule est naïf, le véhicule bleu est hachuré car l'ego-véhicule ignore sa présence. Aucune information supplémentaire ne peut remédier à cette lacune.

La plupart des approches envisagent le cas d'un véhicule clairvoyant, l'objectif étant d'exploiter la connaissance de la provenance du risque. Les autres cas sont généralement utilisés à titre de comparaison dans les évaluations expérimentales.

2.2 État de l’art et approches existantes

Les problématiques liées à la planification de mouvement sans collision ont fait l’objet de nombreux travaux. Certaines approches ont été proposées pour des robots mobiles ([Latombe, 1991], [LaValle, 2006], [Minguez *et al.*, 2008], [Hoy *et al.*, 2015]) et d’autres pour des véhicules autonomes ([González *et al.*, 2016], [Paden *et al.*, 2016]).

Les premières méthodes de planification de véhicules sont directement issues de la robotique et ne considèrent pas d’obstacles dynamiques. Ces méthodes, centrales pour comprendre les suivantes, seront présentées dans la section 2.2.1.

Les approches qui tiennent compte des obstacles dynamiques seront décrites dans la section 2.2.2.

Ces méthodes, souvent issues de la robotique, ont été adaptées au cas de la conduite autonome dans la section 2.2.3 en tenant compte des aspects spécifiques au domaine .

2.2.1 Planification dans un environnement statique

Parmi les approches les plus utilisées en recherche de chemin, que ce soit en robotique ou en conduite autonome, nous distinguons : les approches formalisant l’intégration d’obstacles fixes dans l’espace de configuration (section 2.2.1.1) ; les approches de recherche de chemin dans des graphes (section 2.2.1.2) ; les approches basées sur une analogie physique avec l’électrostatique (section 2.2.1.3).

2.2.1.1 Extension des obstacles fixes dans l’espace de configuration

Certaines approches calculent un chemin évitant des obstacles fixes en calculant les configurations en collision avec ces obstacles fixes. Un chemin dans l’espace de configuration introduit dans [Lozano-Pérez et Wesley, 1979] représente les transformations possibles pouvant être appliquées au robot se déplaçant parmi des obstacles.

Pour construire l’espace de configuration d’un robot dans un environnement statique dans \mathcal{R}^2 , les obstacles sont modélisés par des polygones, notés P .

Cette représentation géométrique permet de calculer les zones occupées de l’espace de configuration \mathcal{O} que nous avons vu dans la section 2.1.4 grâce à la somme de Minkowski entre l’occupation des obstacles \mathcal{O}_{obs} , qui est une collection de polygones, et l’occupation de l’ego-véhicule \mathcal{O}_{ego} qui, en utilisant une boîte englobante, est aussi un polygone. Nous le notons P_{ego} .

La somme de Minkowski est une opération de somme géométrique sur les parties d’un espace vectoriel. Pour deux parties A et B , elle somme les ensembles formés

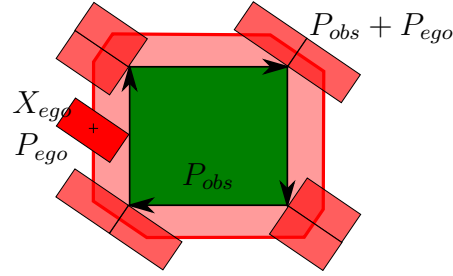


FIGURE 2.6 – Exemple en deux dimensions d’une somme de Minkowski. La somme est faite entre un ensemble P_{obs} (carré en vert) et un ensemble P_{ego} (rectangle en rouge). La somme de Minkowski $P_{obs} + P_{ego}$ correspond au contour rouge. Nous remarquons que si X_{ego} est sur le contour de $P_{obs} + P_{ego}$, alors P_{ego} intersecte P_{obs} en un point.

des sommes d’un élément de A et d’un élément de B . En pratique, elle est obtenue en balayant les limites de A avec un point de l’ensemble B .

$$A + B := \{a + b | a \in A, b \in B\} \quad (2.10)$$

Dans le cas qui nous intéresse, comme dit plus haut, nous souhaitons obtenir l’occupation dans l’espace de configuration \mathcal{O} en faisant la somme Minkowski de P_{ego} avec l’ensemble des polygones dans \mathcal{O}_{obs} . Pour l’explication et sans perte de généralité, nous ne considérons qu’un polygone de \mathcal{O}_{obs} que nous notons P_{obs} . Nous utiliserons toujours le centre du polygone P_{ego} pour réaliser la somme de Minkowski : dans notre cas, il correspondra toujours à la configuration du véhicule X_{ego} .

Nous balayons P_{ego} par son centre autour du polygone P_{obs} représentant l’occupation de l’obstacle (cf. figure 2.6).

Cette somme permet de réduire l’ego-véhicule à un point en grossissant les obstacles grâce à la propriété suivante :

$$X_{ego} \notin P_{ego} + P_{obs} \iff P_{ego} \cap P_{obs} = \emptyset \quad (2.11)$$

Le polygone obtenu correspondent bien à \mathcal{O} car d’après l’équation (2.11) nous respecterons bien l’équation (2.6).

Dans le cas géométrique, nous écrirons donc l’équation (2.6) comme ci-dessous :

$$\mathcal{O} = \mathcal{O}_{obs} + \mathcal{O}_{ego} \quad (2.12)$$

[Lozano-Pérez et Wesley, 1979] discutent aussi la prise en compte d’un robot rectangulaire pouvant tourner sur lui-même.

En effet, la figure 2.6 représente \mathcal{O} pour une seule orientation du robot. Une proposition des auteurs pour le cas où nous voudrions arriver avec un angle β différent de celui de départ α est de calculer trois espaces de configuration différents :

- \mathcal{O}_α avec le robot dans l'orientation α ,
- \mathcal{O}_β avec le robot dans l'orientation β ,
- $\mathcal{O}_{\alpha+\beta}$ où nous utilisons une boîte englobante plus grande qui puisse contenir le robot dans les deux orientations.

Cela permet de résoudre le problème en résolvant les sous-problèmes suivants :

- obtenir la configuration la plus proche de l'arrivée qui ne soit pas dans $\mathcal{O}_{\alpha+\beta}$,
- calculer un chemin du départ à ce point qui ne soit pas dans \mathcal{O}_α ,
- tourner le robot dans l'orientation β ,
- calculer un chemin jusqu'à l'arrivée qui ne soit pas dans \mathcal{O}_β .

Cette approche propose une simplification de l'espace de configuration en faisant une disjonction de cas en fonction de l'orientation du robot, puis en gérant le passage d'une orientation à une autre en tenant compte de l'occupation du robot lors de la rotation.

Cette approche est adaptée à des véhicules pouvant tourner sur place, ce qui n'est pas le cas d'un véhicule automobile.

Nous pouvons aussi représenter l'occupation dans l'espace de configuration pour plusieurs orientations suffisamment proches pour pouvoir négliger la rotation d'une orientation à l'autre. Pour ce faire, nous utilisons une discrétisation de l'orientation tous les $d\alpha$ et approximations l'occupation \mathcal{O} par l'ensemble des occupations à chaque orientation discrète. Nous utilisons alors l'occupation la plus proche de l'état actuel de l'ego-véhicule.

$$\mathcal{O} = \mathcal{O}_\alpha | \alpha \in \llbracket 0, d\alpha, 2\pi \rrbracket \quad (2.13)$$

où $\llbracket 0, d\alpha, 2\pi \rrbracket$ est la liste de tous les angles entre 0 et 2π distant de $d\alpha$.

2.2.1.2 Méthodes de recherche de chemin dans un graphe

Les approches de recherche de chemin dans un graphe sont très utilisées. Elles sont toutes des adaptations ou des extensions de l'algorithme de Dijkstra que nous allons détailler rapidement (a). Nous allons voir que dans un espace de configuration fini et connu, cet algorithme permet de trouver la meilleure solution. Cependant, d'autres approches l'ont adapté pour trouver une solution plus rapidement (b). D'autres approches encore se sont intéressées au problème d'exploration (c), soit pour trouver une solution plus rapidement, soit pour remédier au problème d'exploration d'un espace trop grand ou inconnu ((d) et (e)).

a) Algorithme de Dijkstra L'algorithme de Dijkstra [Dijkstra, 1959] permet de résoudre le problème du plus court chemin dans un graphe, chaque sommet du

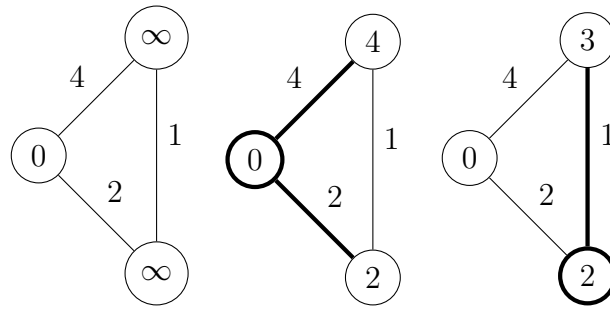


FIGURE 2.7 – Évolution des poids d'un graphe dans l'algorithme de Dijkstra étape par étape.

La mise à jour des poids est faite en fonction du sommet en gras. À gauche, le sommet de départ a un poids de 0, et les deux autres sommets n'ont pas encore été évalués, leur poids est donc infini. Au centre, en considérant le sommet de départ, nous déduisons le poids de chaque autre sommet à partir du coût de transition. Nous trouvons donc 4 pour le sommet du haut et 2 pour le sommet du bas. Enfin, à droite, en considérant le sommet du bas, nous obtenons un poids pour le sommet du haut de 2 (le poids du sommet considéré plus 1 le coût de transition). Nous obtenons donc un poids de 3 ce qui est inférieur au poids précédemment calculé. Nous mettons donc à jour le nouveau poids.

graphe étant un état possible. La transition entre deux états est un arc du graphe associé à un coût de transition. À partir d'un état de départ, la méthode consiste à calculer un poids pour chaque sommet du graphe. Celui-ci correspond au coût cumulé le plus petit, depuis le sommet de départ jusqu'à ce sommet. Ce poids est ensuite mis-à-jour par propagation (voir l'exemple de la figure 2.7).

En général, pour les approches, le graphe est construit en discrétisant l'espace de configuration et le coût est simplement la distance entre les configurations.

Dans ce cas, l'algorithme Dijkstra trouve la meilleure solution qui est le chemin le plus court.

b) Graphe de visibilité Les méthodes par graphe de visibilité se placent dans le même contexte sauf que la construction du graphe ne se fait pas en discrétisant l'ensemble de l'espace de configuration. En effet, une propriété permet d'accélérer la construction du graphe. Nous savons que le chemin le plus court est la ligne droite si cela est possible, sinon, le chemin le plus court est tangent aux obstacles : le chemin le plus court est donc une ligne brisée tangente aux obstacles. Si les obstacles sont des polygones, alors nous pouvons ajouter que le chemin le plus court est une ligne brisée passant par les sommets des obstacles. Nous allons donc maintenant détailler la construction de ce graphe de visibilité. Comme dit plus

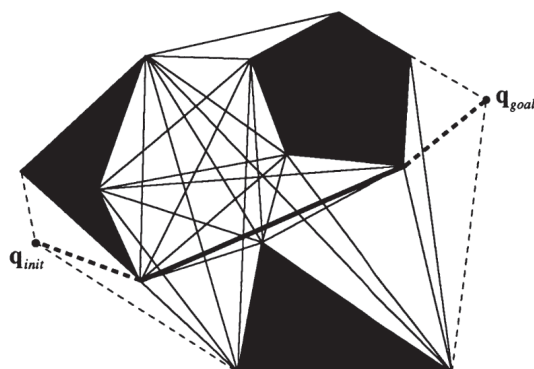


FIGURE 2.8 – Représentation graphique d’une méthode par graphe de visibilité dans [Latombe, 1991].

haut, les obstacles étant polygonaux, ils comptent un certain nombre de sommets. Le graphe est constitué de l’ensemble de ces sommets auxquels nous ajoutons la position de départ de l’ego-véhicule ainsi que l’objectif. En reliant tous les points du graphe deux à deux, nous obtenons deux types de segments : des segments qui n’intersectent pas les polygones et des segments qui les intersectent. Les connexions entre les points dans le graphe de visibilité sont les segments n’intersectant pas les polygones.

Dans le graphe, nous associons à chaque segment un coût. Comme dit précédemment, il s’agit en général de la distance du segment. Le chemin le plus court est celui qui minimise le coût total du départ à l’objectif et peut être trouvé grâce à l’algorithme de Dijkstra (a).

c) Algorithmes d’exploration Il est possible que l’ensemble des configurations de l’espace de configuration ne soit pas connu entièrement à l’avance. Il peut aussi arriver qu’il soit trop grand pour considérer de le discrétiser entièrement pour la recherche de chemin par souci de mémoire nécessaire ou de temps de calcul. Dans ce cas, il est généralement préférable de ne pas construire tout le graphe, et d’utiliser des algorithmes d’exploration. Les algorithmes d’exploration ne garantissent pas systématiquement de trouver la meilleure solution, mais ils permettent de trouver des solutions plus rapidement ou dans des cas plus généraux. Les algorithmes d’exploration sont basés sur des heuristiques. L’un des exemples d’heuristique d’exploration le plus simple pour l’algorithme de Dijkstra est basé sur le coût de transition. Si ce coût est la distance entre les sommets du graphe, alors l’heuristique sera de considérer un graphe constitué des états compris dans un cercle autour de la configuration de départ et que nous appelons le rayon d’exploration. Le rayon d’exploration augmentera jusqu’à ce que le graphe permette de trouver

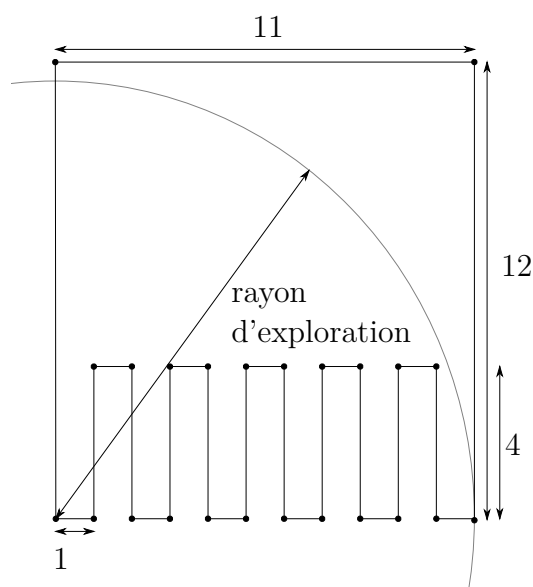


FIGURE 2.9 – Exemple où l’heuristique du rayon d’exploration ne trouve pas le chemin le plus court.

un chemin jusqu’à la configuration d’arrivée.

Nous voyons que l’heuristique permettra de trouver une solution, mais que le chemin trouvé n’est pas nécessairement le chemin le plus court. Nous pouvons en effet imaginer des scénarios où le chemin le plus court nécessite d’agrandir encore le rayon d’exploration (cf. figure 2.9).

d) A* L’heuristique la plus utilisée est l’heuristique de l’algorithme A* introduit par [Hart *et al.*, 1972]. [LaValle, 2006] résume bien l’utilisation de cet algorithme en robotique. Cette heuristique consiste à explorer en direction de l’objectif. Contrairement à l’heuristique précédente, A* permet de trouver la meilleure solution et d’accélérer la recherche dans la majorité des cas (cf. figure 2.10).

e) RRT Les méthodes basées sur les RRT sont une autre famille de méthodes de recherche dans des graphes. Ici, l’heuristique utilisée est de choisir aléatoirement des points à la façon de la méthode de Monte-Carlo [Metropolis et Ulam, 1949]. Introduite par [LaValle, 1998], la méthode permet l’exploration plus ou moins profonde de l’environnement, en fonction du nombre d’itérations, sans prendre en compte tous les sommets des obstacles. Nous reprendrons les notations introduites dans le papier pour exposer le principe de l’approche. Le graphe T est construit à partir de la position de l’ego-véhicule x_{init} . À chaque itération :

- Un point est choisi au hasard dans l’environnement : x_{rand} ,

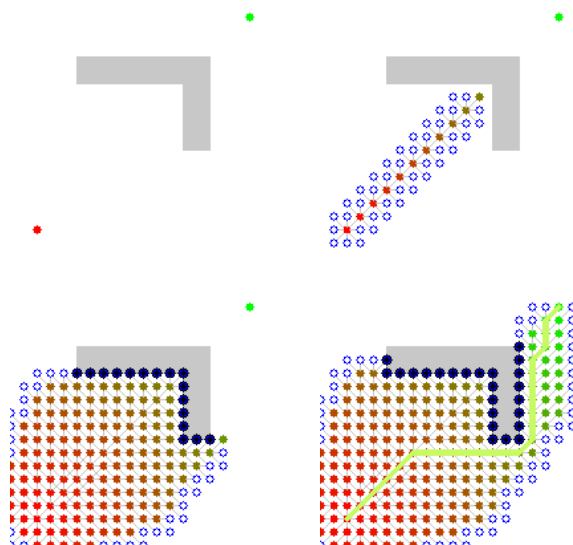


FIGURE 2.10 – Exemple de recherche de chemin par A*. Nous voyons que le graphe se construit par exploration vers l’objectif. Cette figure a été réalisée par Subhrajit Bhattacharya.

- puis relié au point le plus proche x_{near} dans le graphe par un segment.
- Nous prenons sur ce segment un point à une distance u que nous appelons x_{new} .
- Si x_{new} n’est pas dans un polygone, il est ajouté au graphe et connecté au point du graphe x_{near}

Nous pouvons voir différents exemples dans la figure 2.11.

Comme dans le graphe de visibilité nous pouvons trouver le chemin le plus court en donnant un coût aux branches du graphe.

2.2.1.3 APF

Les méthodes de champs de potentiel artificiels, appelées APF (artificial potential field), ont été introduites dans [Khatib, 1985, Latombe, 1991]. Ces méthodes se basent sur une analogie avec l’électrostatique pour calculer un chemin d’évitement d’obstacles jusqu’à un objectif. L’ego-véhicule et l’objectif sont des points, il existe des obstacles représentés par des polygones et l’objectif de l’approche est de trouver une ligne continue qui n’intersecte pas les polygones.

L’analogie fonctionne de la manière suivante : les éléments du problème initial sont transposés dans le problème électrostatique en les remplaçant par des charges électrostatiques (cf. figure 2.12 (a)).

En électrostatique, les charges induisent un champ électrostatique qui est un

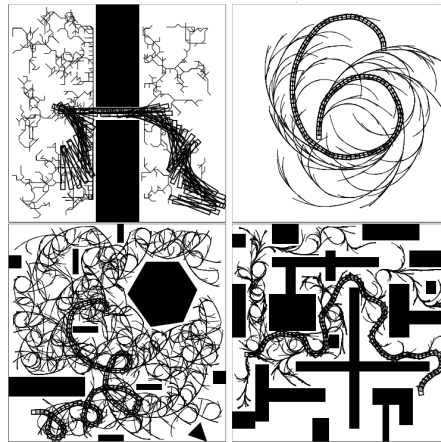


FIGURE 2.11 – Représentation graphique d’une méthode par RRT dans [LaValle, 1998].

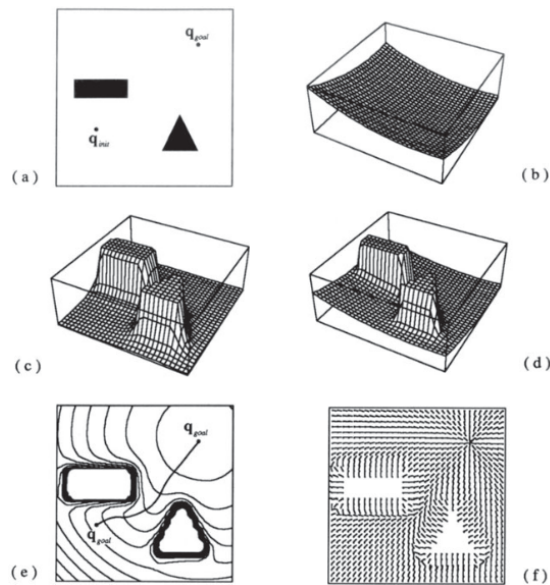


FIGURE 2.12 – Représentation graphique d’une méthode par APF dans [Latombe, 1991].

champ scalaire qu'il est possible de représenter par une surface. Le but induit un champ attractif, comme illustré dans la figure 2.12 (b), et les obstacles, un champ répulsif (cf. figure 2.12 (c)). La somme de deux champs est représentée dans la figure 2.12 (d). Dans ce champ résultant, une particule chargée subit une force opposée et proportionnelle au gradient du champ scalaire. Cette force résultante peut être représentée par un champ vectoriel, comme dans la figure 2.12 (f). Sous l'action de cette force, la particule se déplace jusqu'au point de potentiel le plus bas localement donnant la ligne de la figure 2.12 (e).

C'est ce que nous appelons une descente de gradient. Cette méthode permet dans beaucoup de cas de trouver un chemin rapidement, mais elle peut être compliquée à régler : l'analogie n'aide pas à choisir les ordres de grandeurs pour les charges équivalentes et peut ne pas donner la bonne solution dans des cas présentant plusieurs minima locaux.

2.2.2 Planification dans un environnement dynamique

Les approches existantes peuvent être scindées en deux grandes familles :

- celles qui effectuent une résolution directe, c'est-à-dire qui calculent simultanément le chemin et la vitesse du véhicule (section 2.2.2.1),
- et celles qui décomposent le problème en deux sous-problèmes comme dans [Kant et Zucker, 1986] : la planification du chemin et le calcul du profil de vitesse (section 2.2.2.2).

2.2.2.1 Résolution directe

Les principales familles de méthodes de planification par résolution directe sont : les méthodes utilisant les APF (a), celles qui adaptent l'espace de configuration à l'intégration d'obstacles dynamiques (b), celles se plaçant dans un espace intermédiaire (espace des vitesses) (c) et les approches combinatoires (d).

a) APF Les méthodes de champs de potentiel artificiels, appelées APF (artificial potential fields) [Khatib, 1985, Latombe, 1991], consistent à définir une fonction répulsive agissant sur le véhicule en fonction de l'environnement. Elles ont d'abord été introduites pour permettre au robot d'éviter les obstacles fixes, puis ont été étendues aux obstacles dynamiques dans [Ko et Lee, 1996, Ge et Cui, 2002] et [Huang, 2009].

Nous remarquons que [Ko et Lee, 1996] modifient les APF pour que la vitesse de l'obstacle soit incluse dans la définition de la fonction répulsive potentielle.

[Ge et Cui, 2002] vont plus loin : les fonctions répulsives potentielles tiennent compte à la fois de la position relative et de la vitesse relative du robot par rapport à la cible et aux obstacles.

Dans [Huang, 2009], les fonctions potentielles ne sont pas modifiées, mais une nouvelle formulation de la vitesse planifiée du robot est introduite : elle est déterminée par les vitesses relatives et les positions relatives entre le robot, les obstacles et la cible.

La combinaison d'un moteur à logique floue (fuzzy logic) avec un planificateur de trajet utilisant les champs de potentiels électrostatiques (electrostatic potential fields) est proposée dans [Tsourveloudis *et al.*, 2001].

b) Intégration des obstacles dynamiques à l'espace de configuration

Pour [van den Berg et Overmars, 2008], la vitesse maximale des obstacles en mouvement est supposée connue et leur mouvement est représenté par des disques qui grandissent avec le temps. Le but est de calculer une trajectoire entre ces disques en expansion.

Une autre façon de tenir compte des obstacles dynamiques dans l'espace de configuration est de considérer le temps comme une variable de l'espace de configuration. L'espace de configuration est alors spatio-temporel comme le proposent [Erdmann et Lozano-Pérez, 1987] et [Fujimura et Samet, 1990] dans leurs approches.

[Erdmann et Lozano-Pérez, 1987] approximent l'espace de configuration spatio-temporel par une liste de tranches d'espace de configuration par et le robot se déplace à une vitesse supérieure à celle de n'importe quel obstacle.

c) Espace des vitesses

D'autres méthodes résolvent le problème en utilisant l'espace des vitesses, comme l'approche par fenêtre dynamique (dynamic window) [Fox *et al.*, 1997, Seder et Petrović, 2007]. Dans ces approches, un ensemble de solutions possibles pour le robot sont exhibées à partir de la dynamique du robot et de trois règles :

- les trajectoires sont circulaires,
- la vitesse est admissible, c'est-à-dire que le robot peut s'arrêter avant un obstacle qui se trouve sur la courbure correspondante,
- et la vitesse est dans une fenêtre dynamique autour de la vitesse actuelle, c'est-à-dire que la vitesse peut être atteinte en une courte durée donnée (cf. figure 2.13).

Une solution est ensuite trouvée par optimisation.

Une famille d'approches utilise la notion de cône de collision (collision cone) introduite par [Chakravarthy et Ghose, 1998] ou celle d'obstacles de vitesse VO (velocity obstacles) introduite par [Fiorini et Shiller, 1993], puis reprise et étendue par [Fiorini et Shiller, 1998, van den Berg *et al.*, 2008, Wilkie *et al.*, 2009]. Ces méthodes utilisent la vitesse relative entre le robot et les agents de l'environnement pour construire des zones d'obstacles dans l'espace des vitesses qui permettront

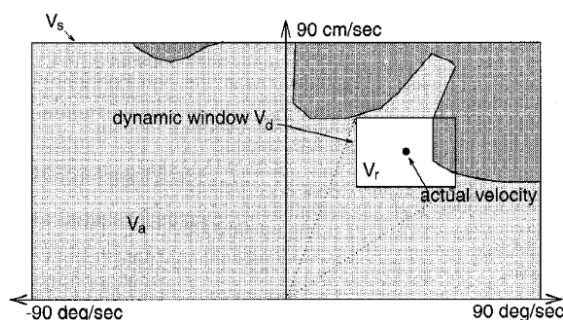


FIGURE 2.13 – Espace des vitesses dans l’approche dynamic window [Fox *et al.*, 1997].

V_s est l’espace des vitesses possibles, V_a l’ensemble des vitesses (en gris clair) qui permettent de s’arrêter avant un obstacle et V_d est l’ensemble des vitesses dans la fenêtre dynamique. Nous avons alors l’ensemble des vitesses admissibles $V_r = V_s \cap V_a \cap V_d$.

de tester si la vitesse du robot va mener à une collision avec les agents de l’environnement. L’espace des vitesses n’est pas représenté de la même manière que précédemment. Les vitesses sont représentées dans l’espace de configuration sous forme de vecteurs. La collision entre le robot et un agent risque de survenir si le vecteur vitesse se situe dans un obstacle de vitesse. Une fenêtre dynamique de vitesse est aussi utilisée pour trouver les vitesses admissibles par le robot (cf. figure 2.14).

d) Approches combinatoires Dans [Bender *et al.*, 2015], les différentes manœuvres possibles sont discrétisées et énumérées, puis regroupées comme faisant partie du résultat d’une décision combinatoire.

En effet, les différentes possibilités sont d’abord les différents chemins que peut emprunter le véhicule. Les classes de chemins désigneront tous les chemins découlant de la même décision. Par exemple, face à un obstacle, deux classes de chemins sont créées : les chemins passant par la droite et les chemins passant par la gauche.

Ensuite, sur chacune des classes de chemins, à chaque interaction avec un autre agent, le véhicule peut passer avant ou après cet autre agent, créant de la même façon deux classes de chemins.

La méthode sélectionnera le chemin le plus respectueux des contraintes qui lui auront été imposées.

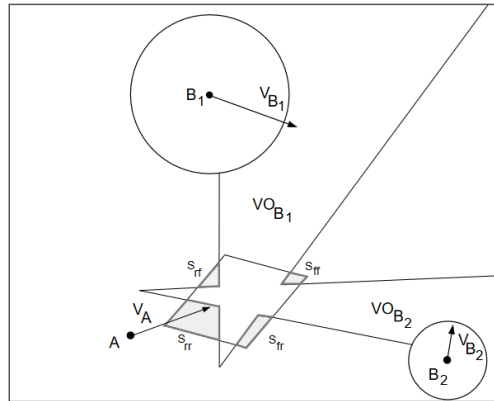


FIGURE 2.14 – Exemple de VO.

Un cône de collision est construit pour chaque obstacle, il correspond aux directions menant à une collision avec l'obstacle si celui-ci était fixe. Pour tenir compte de la vitesse de l'obstacle, le cône de collision est translaté du vecteur vitesse de l'obstacle. Par exemple, pour l'obstacle B_1 ayant une vitesse V_{B_1} , nous obtenons le cône VO_{B_1} . Nous avons un autre cône obtenu pour l'obstacle B_2 . Les zones qui ne mènent pas aux VO de la fenêtre dynamique sont représentées en gris. Nous obtenons quatre zones qui correspondent à des familles de solutions différentes : S_{rr} pour les vitesses qui font passer le robot après les deux obstacles, S_{fr} pour les vitesses qui font passer le robot avant B_1 mais après B_2 et inversement pour la zone S_{rf} et enfin S_{ff} pour les vitesses qui font passer le robot avant B_1 et B_2 . La figure est une simplification d'une figure de [Fiorini et Shiller, 1998].

2.2.2.2 Planification par décomposition

a) Planification avec un chemin prédéterminé Pour générer une trajectoire, certaines approches recherchent d'abord un chemin sans collision en ne considérant que les obstacles statiques ou utilisent un chemin préalablement déterminé. Elles recherchent ensuite un profil de vitesse pour calculer un mouvement sans collision avec les obstacles dynamiques.

Intéressons-nous ici aux approches qui se placent dans l'espace de configuration TP (Time-Path) où le chemin et le temps sont discrétisés et où les zones occupées par les obstacles mobiles sont calculées, comme dans [Kant et Zucker, 1986].

Dans cette approche, un graphe de visibilité est utilisé pour trouver un chemin correspondant au meilleur profil de vitesse en considérant les zones d'interactions possibles le long du chemin. Le profil obtenu n'est pas forcément dérivable.

Les travaux réalisés par [Johnson et Hauser, 2012] et leur extension [Johnson et Hauser, 2013] représentent les interactions le long d'un chemin connu dans l'espace de configuration TP. La méthode calcule ensuite l'ensemble des vitesses possibles en chaque sommet des zones d'interaction. Les vitesses possibles en un point dépendent des vitesses possibles aux points précédents. Le profil de vitesse peut ensuite être déduit par rétro-propagation.

[Krüger, 2019] propose que le chemin et le temps soient discrétisés sur des horizons donnés puis de relier les états possibles dans un graphe. Un coût est calculé pour chaque transition, et permet de trouver la meilleure suite d'états et de donner un profil de vitesse.

b) Planification avec un profil de vitesse prédéterminé Certaines approches fixent *a priori* un profil de vitesse pour résoudre la recherche de chemin.

Quand la vitesse des obstacles mobiles est faible, certaines approches utilisent la planification de mouvement dans un environnement considéré comme statique et recalculent la planification à intervalles réguliers.

C'est le cas pour [LaValle, 1998, Huang et Chung, 2004], où un graphe représentant des chemins candidats (n'intersectant pas les obstacles fixes) est calculé en utilisant les RRT (Rapidly exploring Random Tree) ou un graphe de visibilité. Le chemin le plus court est ensuite sélectionné dans ce graphe.

Pour les environnements fortement dynamiques, c'est-à-dire où les obstacles peuvent avoir une vitesse supérieure à la vitesse de l'ego-véhicule et où le mouvement des obstacles comporte des incertitudes, [Chiang *et al.*, 2015a] proposent une autre approche. Celle-ci intègre des positions atteignables stochastiques ou SR-sets (stochastic reachable sets). Les SR-sets sont les probabilités de présence d'obstacles au mouvement stochastique dans un futur proche. La méthode intègre des obstacles au mouvement stochastique se déplaçant soit sur des segments de ligne ou sur des arcs de cercles. Ensuite, une méthode APF calcule le chemin à partir de ces

SR-sets. Cette méthode a été étendue par [Malone *et al.*, 2017] pour des obstacles qui peuvent passer de l'un à l'autre type de trajectoire.

Dans les méthodes SES et DRT développées par [Chiang *et al.*, 2015b, Chiang *et al.*, 2017b] et dans beaucoup d'autres (voir l'aperçu réalisé par [Paden *et al.*, 2016]), les méthodes sont basées sur les RRT [Karaman et Frazzoli, 2011] et utilisent les SR-sets pour tester la probabilité de collision et s'assurer qu'elle soit inférieure à un seuil avant chaque ajout d'un nœud dans l'arbre.

Une autre approche consiste à représenter les zones d'interactions possibles avec les agents de l'environnement en considérant les RS du robot. Ces zones appelées RIS pour *Reachable Interaction Sets* ont été développées par [de Beaucorps *et al.*, 2018]. Un chemin peut alors être trouvé en évitant les RIS avec n'importe quelle méthode d'évitement d'obstacles statiques.

2.2.3 Planification de mouvement pour les véhicules autonomes en milieu urbain

Les méthodes que nous avons vues jusque-là sont en majorité des approches du domaine de la robotique, dans le sens où elles ne prennent pas en compte les contraintes liées spécifiquement aux véhicules et à leur environnement urbain. Nous verrons que des travaux portent sur la prise en compte des contraintes cinématiques des véhicules (section 2.2.3.1), sur la prise en compte des limitations des systèmes de perception (section 2.2.3.2), sur le respect des règles de conduites (section 2.2.3.3) et enfin sur la prise en compte de l'aspect incertain des autres agents (section 2.2.3.4).

2.2.3.1 Contraintes cinématiques

Certaines approches peuvent être adaptées pour prendre en compte les contraintes cinématiques du véhicule.

Le chemin de Dubins introduit dans [Dubins, 1957] est utilisé en planification de chemin [LaValle, 2006] pour trouver le chemin le plus court en considérant une courbure maximale du chemin. Le chemin ainsi trouvé peut être suivi par un véhicule non-holonome.

Les approches vues plus tôt basées sur les RRT : SES par [Chiang *et al.*, 2015b] et DRT par [Chiang *et al.*, 2017b] peuvent tenir compte des contraintes dynamiques lors de la construction de l'arbre.

[Fedotov *et al.*, 2011] définissent des RS (reachable sets) pour des véhicules non-holonomes. Ces RS ont été utilisées avec les RIS (Reachable Interaction Sets) par [de Beaucorps *et al.*, 2018] pour améliorer les résultats de la méthode lorsque le robot n'est pas holonome.

Dans les travaux de [Hubmann *et al.*, 2016], différentes actions correspondant à différentes valeurs d'accélération de l'ego-véhicule permettent de construire un arbre de recherche. Pour chaque nœud de l'arbre, différentes fonctions de coûts permettent de tenir compte de la sécurité, du confort et du temps de trajet. A^* est ensuite utilisé pour trouver le meilleur chemin.

2.2.3.2 Prise en compte des limitations des systèmes de perception

Certaines approches ont proposé des solutions pour remédier aux limitations dues à la portée des capteurs de perception et aux occultations causées par l'environnement.

Les techniques proposées peuvent être classifiées en plusieurs catégories :

a) Filtre à particules Des filtres à particules sont utilisés par [Narksri *et al.*, 2019] et [Yu *et al.*, 2019] pour évaluer les risques dûs à des véhicules qui seraient occultés.

Les particules représentent des véhicules occultés. [Narksri *et al.*, 2019] évaluent le temps mis par un véhicule occulté pour barrer la route à l'ego-véhicule. Cette approche tient aussi compte de la façon dont la visibilité dont disposent les autres véhicules influence leurs comportements. En effet, l'approche considère que si l'ego-véhicule est vu, les autres véhicules seront prêts à faire un écart, ce qui laisse plus de marge pour avancer sur l'intersection et donc augmenter sa visibilité. Le temps pour traverser le croisement est, lui aussi, évalué. La comparaison des deux temps permet la prise de décision.

Dans [Yu *et al.*, 2019], les risques sont calculés grâce à un filtre à particules : les particules se propagent dans les zones occultées. Une fonction de risque est calculée en fonction de la distance entre l'ego-véhicule et les particules. La fonction de coût à minimiser pour trouver la vitesse tient compte de la fonction de risque et d'une vitesse désirée.

Les approches utilisant des filtres à particules donnent une représentation probabiliste de l'environnement, mais ne considèrent pas le pire cas possible.

b) Obstacles virtuels Les zones d'où peuvent surgir des véhicules occultés sont identifiées comme étant une source de danger. Ce danger peut être pris en compte par des approches qui considère que des véhicules sont effectivement présents dans ces zones. Ces véhicules hypothétiques ajoutés par hypothèse sont appelés obstacles virtuels.

Bien que le terme d'obstacle virtuel ne soit pas utilisé, la méthode développée pour un robot dans un environnement d'intérieur par [Chung *et al.*, 2009], considère le risque des zones occultées pour modifier à la fois le chemin et la vitesse. Pour ce faire, il est considéré que des obstacles mobiles peuvent se trouver n'importe où

dans les zones occultées. En faisant des hypothèses sur la vitesse maximale de ces obstacles, des zones de risques sont construites. Un coût est calculé pour permettre le calcul d'une planification par descente de gradient.

[Orzechowski *et al.*, 2018, Naumann *et al.*, 2019, Hoermann *et al.*, 2017, Tas et Stiller, 2018, Damerow *et al.*, 2017] utilisent ces obstacles virtuels pour améliorer la prise en compte des zones occultées dans leurs approches.

En effet, pour évaluer les risques dus aux zones occultées, [Damerow *et al.*, 2017] positionnent une voiture virtuelle et lui affectent la vitesse maximale autorisée sur ce tronçon pour chaque voie concernée à l'endroit le plus proche de l'intersection à venir. Ils introduisent des obstacles virtuels qui permettent de modéliser tous ces véhicules potentiels. De même, [Orzechowski *et al.*, 2018] utilisent la visibilité pour connaître les zones d'où peuvent surgir de potentiels véhicules. La planification est ensuite réalisée par exploration dans l'espace de configuration spatio-temporel grâce à un RRT. Une trajectoire de secours est calculée à chaque nœud du RRT pour assurer la sûreté de la méthode.

[Hoermann *et al.*, 2017] considèrent trois représentations différentes de l'environnement. La première représentation concerne les objets dynamiques visibles ainsi qu'une prévision de leurs dynamiques. La seconde représentation complète la première avec un filtre à particules. Elle donne une estimation de la dynamique future des obstacles et permet de construire une carte de coûts. Enfin, la troisième tient compte des zones occultées : la dynamique dans les régions non visibles est donnée par un profil de vitesse. Ces représentations permettent de déterminer si une région est sûre. Lorsque c'est le cas, le véhicule tente d'atteindre la vitesse maximale. À défaut, le véhicule avance pas à pas à vitesse lente en ne contrôlant qu'une plus petite région.

[Tas et Stiller, 2018] utilisent la visibilité pour connaître les zones desquelles pourraient surgir des véhicules. Cela permet ainsi de décider si l'ego-véhicule doit ou non céder le passage.

Pour garantir la sécurité sans excès de prudence, [Naumann *et al.*, 2019] placent des véhicules hypothétiques dans les zones occultées. Pour limiter les cas possibles, le papier se concentre sur les intersections en considérant uniquement les véhicules hypothétiques avec priorité de passage sur l'ego-véhicule. La vitesse est trouvée en minimisant une fonction de coût. Le coût est calculé pour chaque action possible puis pondéré par la probabilité que l'action soit ensuite réalisée.

Des véhicules virtuels sont utilisés pour gérer des occultations par [Karanam *et al.*, 2022] dans des situations présentant des giratoires ou des insertions.

Dans le but de prendre en compte plus de types d'obstacles différents et avec toutes les orientations possibles, [Sánchez *et al.*, 2022] utilisent une modélisation d'éventuels obstacles cachés comme un ensemble d'états d'un modèle de masse

ponctuelle, puis calculent l'occupation future en se basant sur une analyse des zones accessibles et des observations précédentes.

c) POMDP : Partially Observable Markov Decision Process D'autres approches tiennent compte des zones occultées dans leur planification en traitant le problème comme un processus de décision de Markov partiellement observable (POMDP).

Dans les travaux de [Bouton *et al.*, 2018], les actions possibles de l'ego-véhicule sont relatives aux accélérations. La méthode est décomposée pour chaque agent : une optimisation de l'utilité de l'action (optimal belief action utility) est calculée comme si chaque agent était seul. Pour déterminer l'action optimale en considérant l'ensemble des agents, deux méthodes sont proposées. La première consiste à sommer les utilités trouvées pour tous les agents, la seconde consiste à ne considérer que l'utilité la plus faible, ce qui revient à ne considérer que l'agent jugé le plus dangereux.

[Hubmann *et al.*, 2019] considèrent des véhicules virtuels dans l'espace d'états. Le modèle d'observation est spécifique aux véhicules virtuels et tient compte du champ de vision sur la voie concernée. De même, les véhicules virtuels ont un modèle de transition spécifique. Sur cette base, une liste d'actions est définie, dans les deux cas les actions possibles sont différentes accélérations, ainsi qu'un modèle de récompense et le POMDP peut être résolu.

Pour [Schorner *et al.*, 2019], la probabilité de placer un véhicule virtuel dépend de la densité du trafic. Les véhicules virtuels arrivent sur le carrefour en même temps que l'ego-véhicule, à condition qu'ils gardent une vitesse réaliste. [Lin *et al.*, 2019] utilisent une recherche dans un arbre d'estimations. Les positions, vitesses et accélérations des autres véhicules sont considérées connues et tous les chemins possibles sont considérés. Des véhicules virtuels sont placés aux limites de visibilité de l'ego-véhicule et sont considérés comme des vrais véhicules.

2.2.3.3 Respect des règles de conduite

[Reichardt et Schick, 1994] adaptent en premier les APF pour les véhicules autonomes en introduisant dans le modèle la notion de voies de circulation.

Les APF sont adaptés par [Gerdes et Rossetter, 2001] pour tenir compte de la notion de voies de circulation.

Ils sont étendus par [Wolf et Burdick, 2008] pour prendre en compte la vitesse des véhicules sur les voies en étendant leurs champs artificiels vers l'avant en fonction de leurs vitesses.

Des obstacles virtuels sont utilisés par [Ajanovic *et al.*, 2018] pour représenter des règles de la route, comme les feux tricolores ou l'interdiction de doubler par la droite.

2.2.3.4 Prise en compte de l'aspect incertain des autres agents

La notion de bande élastique est introduite par [Quinlan et Khatib, 1993] et permet d'adapter un chemin calculé par une autre méthode de recherche de chemin aux changements de l'environnement. Cette notion est utilisée pour des véhicules autonomes dans [Hilgert *et al.*, 2003, Sattel et Brandt, 2005]. La dimension temporelle est prise en compte par l'introduction des bandes élastiques temporelles dans [Rösmann *et al.*, 2012]. [Keller *et al.*, 2014] utilisent cette notion pour traiter des cas où le risque de collision est imminent.

L'aspect incertain des obstacles est considéré par [Huang et Chung, 2004]. Ils créent une carte de risque en estimant la probabilité de présence de l'obstacle dans le temps SR-sets. La probabilité est alors seuillée en fonction du temps pour construire l'obstacle considéré.

2.2.4 Tableaux récapitulatifs

Parmi les approches que nous avons parcourues, nous allons voir quels aspects sont utiles à la résolution de notre problématique : planifier une trajectoire pour un véhicule autonome en milieu urbain parmi des obstacles aux comportements incertains.

2.2.4.1 Récapitulatif des approches pour obstacles fixes

Nous allons d'abord résumer les approches considérant uniquement des obstacles fixes dans le tableau 2.1. Ces approches ont été décrites dans la section 2.2.1.

Toutes ces méthodes utilisent l'intégration des obstacles fixes dans l'espace de configuration et utilisent une somme de Minkowski pour se ramener à un obstacle ponctuel. La somme de Minkowski permet de simplifier le problème dans le cas d'un robot holonome dont l'orientation serait modifiée à l'arrêt. Il est cependant plus complexe de considérer des robots ou véhicules qui tournent en avançant.

Nous comparons pour l'instant des méthodes générales qui n'ont pas forcément d'application directe comme l'algorithme de Dijkstra, mais qui sont utiles pour la mise en œuvre de nombreuses autres méthodes. Le graphe de visibilité fait référence aux méthodes basées sur les travaux de [Latombe, 1991], A* aux méthodes utilisant l'algorithme développé par [Hart *et al.*, 1972], RRT à celles utilisant la méthode définie par [LaValle, 1998] et les APF aux méthodes basées sur celle introduite par [Khatib, 1985, Latombe, 1991].

	Avantages	Inconvénients
Dijkstra <i>Recherche dans un graphe</i>	Solution optimale	Pas d'heuristique : peut être coûteux, long, voire irréalisable selon les cas
Graphe de visibilité <i>Construction d'un graphe avec les sommets des obstacles</i>	Solution optimale, temps de calcul, mémoire	Difficilement adaptable à un véhicule non-holonome, chemin tangent aux obstacles
A* <i>Construction d'un graphe avec heuristique</i>	Rapide, prise en compte de contraintes : coût adaptable	Difficilement adaptable à un véhicule non-holonome, peut perdre du temps dans des minimas locaux
RRT <i>Construction d'un graphe par la méthode de Monte-Carlo</i>	Rapide, facilement adaptable à un véhicule non-holonome, prise en compte de contraintes : coût adaptable	Difficile d'obtenir un chemin lisse, peut perdre du temps dans des minimas locaux
APF <i>Décente de gradient parmi des obstacles répulsifs</i>	Rapide facilement adaptable à un véhicule non-holonome, prise en compte de contraintes : fonction de répulsion adaptable	Peut être piégé dans des minimas locaux

TABLEAU 2.1 – Tableau récapitulatif des approches pour obstacles fixes.

2.2.4.2 Récapitulatif des approches pour obstacles dynamiques

Nous allons maintenant résumer les approches pour obstacles dynamiques présentées dans la section 2.2.2 dans le tableau 2.2. Les types d'approches comparés correspondent à des familles regroupant plusieurs méthodes fonctionnant sur les mêmes bases. Bien que chaque méthode ait ses spécificités, nous comparons ici les avantages et les inconvénients de ces familles de méthodes. Le terme d'APF dynamique regroupe les méthodes développées par [Ko et Lee, 1996, Ge et Cui, 2002, Huang, 2009, Tsourveloudis *et al.*, 2001]. L'intégration des obstacles dynamiques à l'espace de configuration fait référence aux travaux de [van den Berg et Overmars, 2008, Erdmann et Lozano-Pérez, 1987, Fujimura et Samet, 1990]. L'espace des vitesses englobe les méthodes développées par [Fox *et al.*, 1997, Seder et Petrović, 2007, Chakravarthy et Ghose, 1998, Fiorini et Shiller, 1993, Fiorini et Shiller, 1998,

van den Berg *et al.*, 2008, Wilkie *et al.*, 2009]. On parle d'approche combinatoire pour désigner l'approche de [Bender *et al.*, 2015]. On parle enfin, d'approche par décomposition chemin-vitesse pour les approches définies par [Kant et Zucker, 1986, Johnson et Hauser, 2012, Johnson et Hauser, 2013, Krüger, 2019, LaValle, 1998, Huang et Chung, 2004, Chiang *et al.*, 2015a, Malone *et al.*, 2017, Ko et Lee, 1996, Ge et Cui, 2002, Huang, 2009, Chiang *et al.*, 2017b, Karaman et Frazzoli, 2011, de Beaucorps *et al.*, 2018].

	Avantages	Inconvénients
APF dynamiques <i>Adaptation pour obstacles dynamiques</i>	Adaptation concluante aux obstacles dynamiques	Même défaut que les APF statiques
Espace de configuration <i>Intégration des obstacles dynamiques</i>	Permet de représenter des comportements complexes des agents	Occupation des obstacles surestimée, prend rapidement tout l'espace et empêche de trouver une solution
Espace des vitesses	Performant pour la coopération de plusieurs robots	Ne tient compte que de la vitesse instantanée des agents, difficilement adaptable aux véhicules non-holonomes
Approche combinatoire	Adaptation concluante des approches d'explorations	Explosion des possibles avec la dimension temporelle supplémentaire
Approches par décomposition chemin-vitesse	Permet de simplifier le problème	Hypothèses fortes

TABLEAU 2.2 – Tableau récapitulatif des approches pour obstacles dynamiques.

2.2.4.3 Récapitulatif des approches pour les véhicules autonomes en milieu urbain

Nous allons maintenant résumer les approches pour les véhicules autonomes en milieu urbain présentées dans la section 2.2.3 dans le tableau 2.3. De la même manière que dans la section précédente, la comparaison se fait par groupement d'approches. La prise en compte des contraintes cinématiques est implémentée par les méthodes développées par [LaValle, 2006, Chiang *et al.*, 2015a, Chiang *et al.*, 2017b, Fedotov *et al.*, 2011, de Beaucorps *et al.*, 2018, Hubmann *et al.*, 2016]. Les filtres à particules sont uti-

lisés dans les travaux de [Narksri *et al.*, 2019, Yu *et al.*, 2019], et les VOs dans les méthodes de [Chung *et al.*, 2009, Orzechowski *et al.*, 2018, Naumann *et al.*, 2019, Hoermann *et al.*, 2017, Tas et Stiller, 2018, Damerow *et al.*, 2017, Karanam *et al.*, 2022, Sánchez *et al.*, 2022]. Les travaux réalisés par [Bouton *et al.*, 2018, Hubmann *et al.*, 2019, Schorner *et al.*, 2019, Lin *et al.*, 2019] utilisent les POMDP. Le respect des règles de conduite est pris en compte et intégré dans les approches de [Reichardt et Schick, 1994, Gerdes et Rossetter, 2001, Wolf et Burdick, 2008, Ajanovic *et al.*, 2018], et la prise en compte de l'aspect incertain des autres agents apparait dans les approches de [Quinlan et Khatib, 1993, Hilgert *et al.*, 2003, Sattel et Brandt, 2005, Rösmann *et al.*, 2012, Keller *et al.*, 2014, Huang et Chung, 2004].

	Avantages	Inconvénients
Prise en compte des contraintes cinématiques	Augmente la sécurité et la faisabilité de la solution	Pas compatible avec toutes les approches
Filtres à particules	Probabiliste	Peut omettre un cas dangereux peu probable
VOs	Facilement intégrable, tient compte du pire cas	Peut être trop conservateur
POMDP	Probabiliste, bonne appréciation du risque	Très couteux en temps de calcul préalable
Respect des règles de conduites	Obligatoire pour certaines règles en milieu urbain, permet une conduite rassurante et se rapprochant du comportement humain	Contraignant et sous-optimal dans les cas où l'ego-véhicule est plus efficace que l'humain
Prise en compte de l'aspect incertain des autres agents	Permet d'absorber les incertitudes sur la prévision, permet d'utiliser des hypothèses moins fortes et d'adapter les approches à des prévisions plus réalistes	Calculs préalables ou modèles comportementales nécessaires

TABLEAU 2.3 – Tableau récapitulatif des approches pour les véhicules autonomes en milieu urbain.

2.3 Conclusion

Pour traiter le problème d'une planification de trajectoire pour un véhicule en milieu urbain, nous avons déjà choisi de décomposer le problème en sous parties comme le font certaines des approches que nous avons décrites. La décomposition peut être résumée par les points suivants :

- Mode de représentation commun (section 2.3.1),
- Planification de vitesse le long d'un chemin (section 2.3.2),
- Planification de chemin (section 2.3.3),
- Planification autonome (section 2.3.4).

2.3.1 Mode de représentation géométrique dans l'espace de configuration

Pour décomposer le problème en sous problèmes, il fallait déjà choisir un mode de représentation commun permettant de traiter différents cas de figure avec différentes approches tout en gardant la possibilité de combiner les différentes approches entre elles quand cela est nécessaire. Nous avons vu dans les sections 2.2.1 et 2.2.2 que la représentation des obstacles dans l'espace de configuration était l'approche la plus générale. Elle requiert que nous puissions représenter les obstacles statiques et dynamiques par des polygones. Cette hypothèse a peu d'influence sur le résultat final. Pour les obstacles statiques, nous choisissons de confondre toutes les zones que l'ego-véhicule doit éviter. Pour connaître ces zones, il est nécessaire de les calculer au préalable à l'aide d'une carte détaillée contenant des informations sur les éléments permanents de la route et sur les règles de circulation dans la zone. D'éventuelles informations supplémentaires sur des éléments temporaires, comme la présence de travaux ou de signalisations exceptionnelles, pourraient aussi être intégrées. Il faut aussi que le véhicule puisse percevoir des agents fixes à l'aide de capteurs, par exemple une voiture garée qui empièterait sur la chaussée. Ce calcul préalable sera discuté dans le Chapitre 4, nous considérons, en général, ces informations comme parfaitement connues.

Pour intégrer les obstacles dynamiques, il faut être capable de connaître la position et la vitesse des agents visibles, ainsi que leur taille. Nous considérons aussi que le système de perception de l'ego-véhicule se charge de recueillir suffisamment d'informations à l'aide de capteurs afin de connaître son environnement de manière précise. Nous discuterons des incertitudes dans la section 3.2, mais nous supposons en général que les positions et vitesses des autres agents visibles sont connues. Il faut ensuite être capable de donner une prévision sur leurs positions au cours du temps, et ce durant un temps suffisamment long pour garantir la sécurité de la navigation. Pour cela, nous utilisons différents types de modèles comportementaux simples que nous détaillons au fur et à mesure. Pour simplifier le problème, nous

considérons que tous les agents de la route sont des véhicules. Nous pouvons alors reporter ces occupations dans un espace de configuration spatio-temporel. Exprimer l'occupation des obstacles de manière analytique serait extrêmement complexe. Une solution viable est donc de discrétiser l'espace de configuration comme le font [Erdmann et Lozano-Pérez, 1987]. Cela implique aussi d'avoir un espace de configuration qui soit limité en taille. Nous choisissons donc une limite spatiale et une limite temporelle. Ainsi, tous les objets utilisés dans la suite seront sujets à ces discrétisations temporelles et spatiales ainsi qu'aux limites de l'espace de configuration. Par exemple, les occupations seront représentées par des collections de polygones et les chemins et trajectoires par des lignes polygonales.

Nous n'utilisons pas la somme de Minkowski pour réduire l'ego-véhicule à un point dans l'espace de configuration, préférant calculer son occupation propre en fonction de sa configuration. En effet, dans le cas contraire, l'occupation dans l'espace de configuration des autres véhicules dépend de l'orientation de l'ego-véhicule, ce qui rend son calcul très complexe quand le véhicule non-holonyme change d'orientation en avançant.

Le choix de l'espace de configuration spatio-temporel étant une représentation très générale, il n'apporte pas de solution pour réduire la complexité du problème. Nous avons donc décidé d'utiliser la décomposition chemin-vitesse pour se ramener à des problèmes moins complexes.

Du fait de cet ensemble de choix et d'hypothèses, une approche géométrique nous semble pertinente. En effet, celle-ci est adaptée à la réalisation de raisonnements et de calculs dans l'espace de configuration spatio-temporel discrétisé que nous avons choisi. De plus, les objets de cet espace sont polygonaux : cela renforce encore l'utilité d'une approche géométrique.

2.3.2 Planification de vitesse le long d'un chemin

Nous avons précisé précédemment que la décomposition chemin-vitesse impliquait une hypothèse forte sur le chemin ou sur la vitesse. Dans le cas de la planification de vitesse le long d'un chemin prédéterminé, l'hypothèse est en effet contraignante, mais elle est tout à fait acceptable, voire désirable dans la majorité des situations. Les manœuvres sont nécessaires en milieu urbain, mais il arrive souvent que l'ego-véhicule suive de longues portions de route sans manœuvrer au cours du trajet. Il est donc acceptable de considérer séparément la planification de vitesse le long d'un chemin et la planification de chemin. Pour planifier la vitesse le long d'un chemin, il est nécessaire de supposer un chemin de référence sur lequel se trouve toujours l'ego-véhicule. Nous utiliserons pour cela la notion de voies de circulation. Nous supposons que la carte fournie est découpée en différents tronçons de route. Chacun de ces tronçons est subdivisé en plusieurs voies de circulation. Ces voies seront constituées de suites ordonnées de points de passage orientés dans

le sens de la marche. À chaque point sera associée la valeur de la largeur de la voie. Nous supposons, en général, que l’ego-véhicule a toujours une voie bien définie en fonction de sa position. Le chemin de référence sera construit à partir de la position de départ de l’ego-véhicule puis en suivant ces points de passage. Dans certains cas particuliers que nous verrons par la suite, l’ego-véhicule pourra avoir besoin d’un chemin de référence qui ne sera pas construit à partir du centre de sa voie de circulation. Dans ce cas, nous supposons qu’il n’y a pas d’obstacle mobile, nous calculerons un chemin parmi les obstacles statiques puis nous l’utiliserons comme chemin de référence. Pour le calcul d’un chemin évitant les obstacles statiques, nous avons choisi d’utiliser la méthode APF et de l’adapter. En effet, comme nous l’avons dit plus haut, cette méthode peut s’adapter aux véhicules non-holonomes et permet d’intégrer d’autres contraintes, ce qui peut être utile pour obtenir un comportement satisfaisant vis-à-vis de l’environnement urbain.

Nous choisissons d’intégrer des contraintes cinématiques pour obtenir une vitesse réalisable par l’ego-véhicule qui soit à la fois sûre et confortable pour l’utilisateur. Nous pouvons calculer, en fonction du modèle physique de l’ego-véhicule, une vitesse et une accélération maximales pour que celui-ci parvienne à rester sur le chemin. Nous supposons que, pour tous les chemins utilisés, cette vitesse maximale est inférieure à la vitesse maximale autorisée et que l’accélération maximale est inférieure à une accélération maximale de confort. Nous n’autoriserons pas le dépassement de la vitesse maximale autorisée. En revanche, nous pourrions autoriser une accélération plus grande que l’accélération de confort uniquement pour des freinages d’urgence. Ce point sera discuté dans la section 3.3.

Pour la prise en compte des zones occultées, nous supposons que les systèmes de perception de l’ego-véhicule lui permettent de connaître avec précision les zones de la carte qui sont occultées. À partir de ces zones, nous pourrions générer des véhicules virtuels pour simuler la présence de véhicules. Cette solution a été choisie, car elle permet de tenir compte du pire cas, ce qui est nécessaire pour garantir la sécurité. Les situations de blocage occasionnées par cette considération pourront être traitées à part.

2.3.3 Planification de chemin

Pour la planification de chemin, les RIS ont été étendus aux véhicules non-holonomes. Pour ce faire, un modèle cinématique pour l’ego-véhicule a été utilisé : le modèle bicyclette (bicycle model). La cinématique de ce modèle est en général suffisante pour bien représenter le comportement cinématique d’un véhicule. C’est d’ailleurs le modèle cinématique le plus détaillé utilisé dans les approches que nous avons présentées.

2.3.4 Planification en milieu urbain

Pour obtenir une approche qui soit la plus autonome possible en milieu urbain, nous pensons qu'il est nécessaire d'alterner entre une planification du chemin et une planification de la vitesse en fonction des interactions avec l'environnement, ses agents et ses règles. Pour tenir compte des règles de conduite utilisées pour la planification en milieu urbain, nous considérerons que les règles s'appliquant à l'ego-véhicule dépendent de sa position et sont toujours connues. Cette information peut être obtenue par la carte ou grâce aux systèmes de perception de l'ego-véhicule. Pour les feux tricolores, par exemple, nous supposons connaître leurs états à chaque instant, ainsi que les durées des différents états (vert, orange et rouge).

Chapitre 3

Nos approches de planification

Table des matières

3.1	Introduction : approche géométrique	47
3.1.1	Espace-temps	47
3.1.1.1	Planification dans un environnement statique . . .	47
3.1.1.2	Planification dans un environnement dynamique . .	49
3.1.1.3	Conclusion	52
3.1.2	Décomposition chemin-vitesse	52
3.1.2.1	Planification de la vitesse	53
3.1.2.2	Planification d'un chemin	54
3.1.3	Calculs géométriques	56
3.1.3.1	Planification de la vitesse	56
3.1.3.2	Planification d'un chemin	57
3.1.4	Conclusion	58
3.2	Planification de chemin à vitesse connue : méthode RIS et MMRIS	58
3.2.1	Définition des RIS (Reachable Interaction Sets)	58
3.2.2	Extensions des RIS pour la robotique	59
3.2.2.1	Intégration de contraintes de mouvement pour les véhicules non-holonomes	59
a)	Modèle de véhicule : véhicule de Dubins .	59
b)	Construction de RS : angle de braquage constant	60
3.2.2.2	Obstacles stochastiques	61
3.2.2.3	Gestion des cas d'urgences	65
a)	Cas de l'ego-véhicule dans un RIS	65
b)	Cas de l'ego-véhicule à l'arrêt	66
3.2.3	Extensions des RIS pour le milieu urbain	67
3.2.3.1	Intégration des contraintes routières aux mouve- ments de l'ego-véhicule	68
a)	Modèle de véhicule : modèle bicyclette . .	68
b)	Construction de RS : suivi de chemins candidats	69
3.2.3.2	Utilisation des chemins candidats dans la planification	72

3.2.3.3	MMRIS (Multi Modal Reachable Interaction Sets)	72
3.2.3.4	Sécurité et confort	73
a)	Marge de sécurité	73
b)	Time-gap	74
c)	Marge de manœuvre	74
d)	Cooldown	75
3.2.3.5	Règles d'usage et du code de la route	75
3.3	Planification de vitesse avec prise en compte des zones occultées : méthode SGSPA	76
3.3.1	Planification de vitesse dans l'espace chemin-temps	77
3.3.1.1	TP obstacles	78
3.3.1.2	Construction d'un chemin candidat	78
3.3.1.3	Contrainte d'accélération	78
3.3.1.4	État sûr avant une intersection	80
3.3.1.5	Modélisation de la voiture	82
3.3.2	Prise en compte des zones occultées et des limites de perception	82
3.3.2.1	Véhicules virtuels et horizon virtuel	82
3.3.2.2	Calcul de la planification de vitesse	84
3.4	Planification de trajectoire en milieu urbain : méthode GPAD	86
3.4.1	Approche GPAD	87
3.4.1.1	Hypothèses	87
3.4.1.2	Éléments principaux	87
3.4.1.3	Obstacles virtuels et obstacles conditionnels	88
3.4.2	Architecture	88
3.4.2.1	Planification alternative	88
3.4.2.2	Décision du calcul d'une manœuvre	90
3.4.2.3	Chemin de référence	91
3.4.2.4	Calcul du contrôle	91
3.4.3	Planification de chemin par APF	92
3.4.3.1	Grille de calcul	92
3.4.3.2	But local	92
3.4.3.3	Obstacles	92
3.4.3.4	Les voies et le chemin précédent	93
3.4.3.5	Calcul de la charge totale	93
3.4.3.6	Calcul du chemin	93
3.4.4	Conclusion et attentes	94

3.1 Introduction : approche géométrique

Avant de décrire les méthodes développées dans la thèse, nous allons détailler les pré-requis nécessaires pour l'ensemble des approches.

Comme nous l'avons vu dans le Chapitre 2, rechercher une trajectoire sans collision avec l'environnement est un problème qui peut être résolu par une approche géométrique en représentant le véhicule dans l'espace approprié [Latombe, 1991].

Nous allons tout d'abord présenter dans la section 3.1.1 les différentes notions et objets utilisés pour les approches géométriques et spécifier les opérations qui nous seront utiles. En particulier, nous verrons que certaines de ces opérations se ramènent à des calculs d'intersections géométriques.

Puis, dans la section 3.1.2, nous verrons ce qu'implique l'utilisation de la décomposition chemin-vitesse dans une approche géométrique. Ceci nous permettra de spécifier les calculs d'intersections géométriques propres à chacune des hypothèses de la décomposition chemin-vitesse dans la section 3.1.3.

3.1.1 Espace-temps

Le calcul du mouvement de l'ego-véhicule peut se représenter dans un espace à trois dimensions : l'espace-temps, où les deux premières dimensions représentent le plan sur lequel l'ego-véhicule évolue et où le temps est la troisième dimension. Les approches diffèrent selon que l'environnement de l'ego-véhicule est statique ou dynamique. Pour plus de simplicité, nous allons commencer par décrire le formalisme et les objets utilisés pour la planification dans un environnement statique dans la section 3.1.1.1 pour ensuite les étendre au cas où l'environnement de l'ego-véhicule est dynamique dans la section 3.1.1.2.

3.1.1.1 Planification dans un environnement statique

Plaçons-nous tout d'abord dans le cas où l'environnement de l'ego-véhicule est statique, *i.e.* il ne contient pas d'autres objets en mouvement (cf. figure 2.1).

Nous considérons que l'environnement de l'ego-véhicule est représenté par un plan $P(x, y)$ et que l'ego-véhicule ne peut se déplacer que sur la chaussée. Nous avons donc un ensemble de zones inaccessibles délimitées par les bords de la chaussée qui peuvent être considérées comme des obstacles fixes représentés par des polygones dans le plan $P(x, y)$ (cf. figure 2.3).

Nous notons cette collection de polygones :

$$\mathcal{O}_{statique} = \{P_k | k \in [0, N_{statique}]\} \quad (3.1)$$

où $N_{statique}$ est le nombre de polygones représentant les zones inaccessibles.

Étant donné la configuration de l'ego-véhicule :

$$X_{ego} = (x_{ego}, y_{ego}, \theta_{ego}) \quad (3.2)$$

où (x_{ego}, y_{ego}) est la position du centre de l'ego-véhicule et θ_{ego} est l'orientation de l'ego-véhicule, nous voulons prévoir si l'ego-véhicule est en collision avec au moins l'un des polygones de l'environnement.

Pour ce faire, nous utiliserons la boîte englobante de l'ego-véhicule, comme dans [Lozano-Pérez et Wesley, 1979] pour représenter son occupation. L'occupation de l'ego-véhicule est donc représentée par un polygone orienté noté P_{ego} dans l'espace (x, y) en fonction de la configuration du véhicule X_{ego} (cf. figure 2.2) :

$$\mathcal{O}_{ego}(X_{ego}) = \{P_{ego}(X_{ego})\} \quad (3.3)$$

Un chemin est libre si l'ego-véhicule peut le suivre sans entrer en collision avec un des obstacles statiques.

Dans les approches présentées par la suite, nous représenterons les chemins par des lignes polygonales \mathcal{P} [Gomes et al., 2012], formées par une suite de segments consécutifs décrite par une liste de points, extrémités des segments :

$$\mathcal{P} = \{p_i = (x_i, y_i) | i \in [0, N_{\mathcal{P}}]\} \quad (3.4)$$

où $N_{\mathcal{P}}$ est le nombre de points dans la ligne polygonale.

\mathcal{P} pourra aussi être représenté par la liste des segments la constituant :

$$\mathcal{P} = \{\mathcal{S}_{\mathcal{P},i} | i \in [0, N_{\mathcal{P}} - 1]\} \quad (3.5)$$

où $\mathcal{S}_{\mathcal{P},i}$ est le segment d'extrémités p_i et p_{i+1}

Pour prédire si un chemin est libre, il faut calculer la zone occupée par le véhicule le long du chemin suivi \mathcal{P}_{ego} . Elle correspond à la surface balayée par la boîte englobante de l'ego-véhicule le long des segments formant le chemin \mathcal{P}_{ego} . Cette zone peut être calculée en effectuant des sommes de Minkowski entre l'occupation de l'ego-véhicule \mathcal{O}_{ego} et les segments de la ligne polygonale formant son chemin [Lozano-Pérez, 1983]. Pour chaque segment $\mathcal{S}_{\mathcal{P}_{ego},i}$ de la ligne polygonale :

$$\mathcal{O}_{ego}(\mathcal{S}_{\mathcal{P}_{ego},i}) = P_{ego}(p_i, \theta_i) + \mathcal{S}_{\mathcal{P}_{ego},i} \quad (3.6)$$

où θ_i est l'orientation du segment i .

L'union des polygones obtenus sur tous les segments de la ligne polygonale donnent l'occupation totale le long du chemin (cf. figure 3.1) :

$$\mathcal{O}_{ego}(\mathcal{P}_{ego}) = \bigcup_{i=0}^{N_{\mathcal{P}}-1} \mathcal{O}_{ego}(\mathcal{S}_{\mathcal{P}_{ego},i}) \quad (3.7)$$

Si le polygone obtenu et les polygones représentant les zones inaccessibles n'ont pas d'intersection commune, alors le chemin pourra être emprunté par l'ego-véhicule

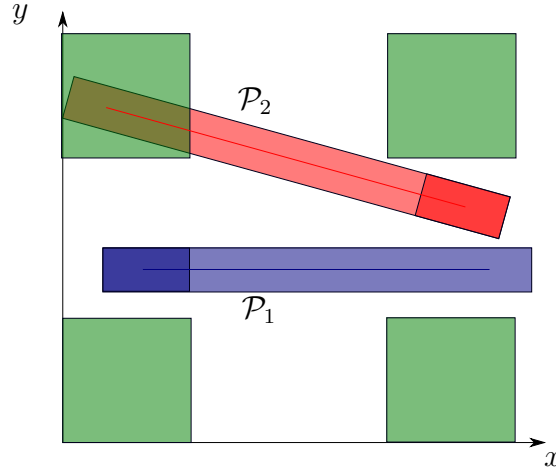


FIGURE 3.1 – Planification de chemin dans un environnement statique. Les deux petits rectangles sont les boîtes englobantes des véhicules alors que les longs rectangles représentent les occupations totales des véhicules le long de leurs chemins respectifs. Le chemin \mathcal{P}_1 est libre alors que le chemin \mathcal{P}_2 est en collision avec l’environnement.

sans collision : le chemin est libre. Le chemin \mathcal{P}_{ego} doit donc vérifier l’équation suivante pour être libre :

$$\mathcal{O}_{ego}(\mathcal{P}_{ego}) \cap \mathcal{O}_{statique} = \emptyset \quad (3.8)$$

3.1.1.2 Planification dans un environnement dynamique

Plaçons nous maintenant dans un cadre plus général où l’environnement peut contenir d’autres objets en mouvement. Pour considérer les mouvements respectifs de l’ego-véhicule et des autres véhicules en mouvement, il est nécessaire d’ajouter à la représentation précédente une dimension temporelle que l’on notera t . Nous représentons alors l’environnement par un espace à trois dimensions (x, y, t) appelé espace-temps (cf. figure 3.3).

La configuration de l’ego-véhicule dépendant du temps, on redéfinit l’occupation de l’ego-véhicule en fonction du temps $\mathcal{O}_{ego}(X_{ego}(t))$ pour une configuration $X_{ego}(t) = (x_{ego}(t), y_{ego}(t), \theta_{ego}(t))$:

$$\mathcal{O}_{ego}(X_{ego}(t)) = \{P_{ego}(X_{ego}(t))_{(t)}\} \quad (3.9)$$

où l’indice (t) du polygone signifie que celui-ci appartient au plan d’espace d’instant t .

Pour prédire si un chemin est libre, il faut maintenant considérer la trajectoire de l’ego-véhicule, *i.e.* la position de l’ego-véhicule sur le chemin au cours du temps.

3. Nos approches de planification

Nous considérons que les trajectoires sont des lignes polygonales dans l'espace-temps (x, y, t) . Elles sont notées :

$$\mathcal{T} = \{q_j = (x_j, y_j, t_j) | j \in [0, N_{\mathcal{T}}]\} \quad (3.10)$$

où $N_{\mathcal{T}}$ est le nombre de points dans la ligne polygonale \mathcal{T} .

Nous utiliserons aussi la notation :

$$\mathcal{T} = \{\mathcal{S}_{\mathcal{T},j} | j \in [0, N_{\mathcal{T}} - 1]\} \quad (3.11)$$

où $\mathcal{S}_{\mathcal{T},j}$ est le segment d'extrémités q_j et q_{j+1} .

De la même manière que précédemment, pour déterminer si la trajectoire est libre, nous pouvons calculer la zone occupée par le véhicule le long de la trajectoire en effectuant les sommes de Minkowski entre l'occupation de l'ego-véhicule et les segments de la ligne polygonale représentant sa trajectoire \mathcal{T}_{ego} .

Chaque point q_j de la trajectoire projeté sur le plan d'espace ($t = 0$) appartient au chemin \mathcal{P}_{ego} .

Pour chaque segment $\mathcal{S}_{\mathcal{T}_{ego},j}$ de la trajectoire \mathcal{T}_{ego} :

$$\mathcal{O}_{ego}(\mathcal{S}_{\mathcal{T}_{ego},j}) = \{P_{ego}(q_j, \theta_j)_{(t=t_j)} + \mathcal{S}_{\mathcal{T}_{ego},j}\} \quad (3.12)$$

où θ_j est l'orientation du segment $\mathcal{S}_{\mathcal{T}_{ego},j}$.

L'orientation θ_j du $j^{\text{ème}}$ segment dans l'espace-temps correspond à l'orientation du segment projeté sur le plan ($t = 0$).

Nous aurons alors :

$$\theta_j = \theta_i \iff i, j | q_j \cdot E(t = 0) \in \mathcal{S}_{\mathcal{P}_{ego},i} \quad (3.13)$$

où $E(t = 0)$ est un plan de l'espace-temps tel que $t = 0$, et où l'opérateur \cdot est une projection orthogonale (cf. figure 3.2).

L'union des polygones obtenus sur tous les segments de la ligne polygonale donne l'occupation totale le long de la trajectoire :

$$\mathcal{O}_{ego}(\mathcal{T}_{ego}) = \bigcup_{j=0}^{N_{\mathcal{T}_{ego}}-1} \mathcal{O}_{ego}(\mathcal{S}_{\mathcal{T}_{ego},j}) \quad (3.14)$$

L'occupation est un volume dans l'espace-temps (x, y, t) .

Les zones inaccessibles sont supposées ne pas dépendre du temps, leur occupation reste donc la même pour toutes les valeurs de t .

De manière analogue à ce qui a été fait pour l'ego-véhicule, nous pouvons maintenant considérer d'autres véhicules ayant chacun une position connue dans le temps (*i.e.* une trajectoire future connue) et un polygone d'occupation connu (*i.e.* une occupation connue dans le temps).

Pour N_{dyn} véhicules, nous notons l'état du $l^{\text{ème}}$ véhicule en fonction du temps t :

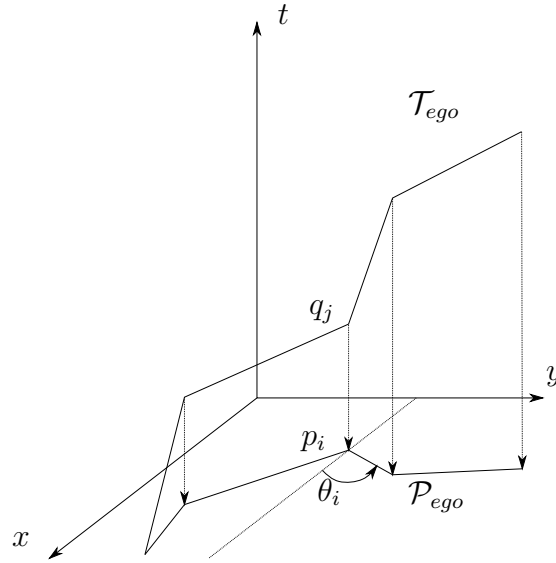


FIGURE 3.2 – Exemple d'orientation d'une trajectoire dans l'espace-temps.

$$X_l(t) = (x_l(t), y_l(t), \theta_l(t)) \quad (3.15)$$

L'occupation du $l^{\text{ème}}$ véhicule est notée :

$$\mathcal{O}_l(X_l(t)) = \{P_l(X_l(t))\} \quad (3.16)$$

Soit une occupation sur l'ensemble de la trajectoire \mathcal{T}_l du $l^{\text{ème}}$ véhicule :

$$\mathcal{O}_l(\mathcal{T}_l) = \cup_{j=0}^{N_{\mathcal{T}_l}-1} \mathcal{O}_l(\mathcal{S}_{\mathcal{T}_l,j}) \quad (3.17)$$

pour une trajectoire $\mathcal{T}_l = \{\mathcal{S}_{\mathcal{T}_l,j} | j \in [0, N_{\mathcal{T}_l} - 1]\}$

L'occupation totale des N_{dyn} véhicules est donc :

$$\mathcal{O}_{others} = \cup_{l=0}^{N_{dyn}} \mathcal{O}_l(\mathcal{T}_l) \quad (3.18)$$

S'il n'y a pas d'intersection entre l'occupation prévue pour l'ego-véhicule et l'occupation totale des N_{dyn} véhicules ainsi que l'occupation représentant les zones inaccessibles, alors la trajectoire peut être empruntée sans collision : elle est libre. Nous notons l'occupation totale des obstacles de l'environnement :

$$\mathcal{O}_{obs} = \mathcal{O}_{statique} \cup \mathcal{O}_{others} \quad (3.19)$$

La trajectoire \mathcal{T}_{ego} doit donc vérifier l'équation suivante pour être libre :

$$\mathcal{O}_{ego}(\mathcal{T}_{ego}) \cap \mathcal{O}_{obs} = \emptyset \quad (3.20)$$

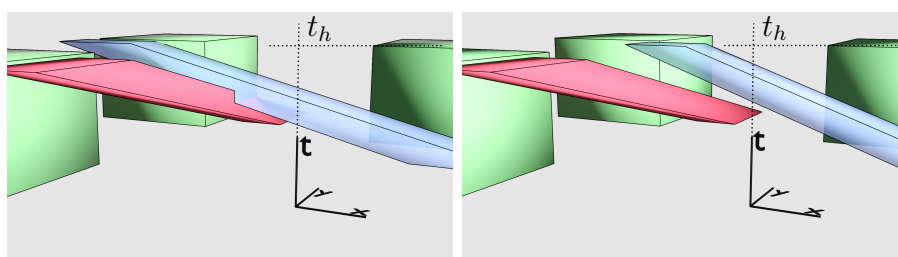


FIGURE 3.3 – Planification dans un environnement dynamique. Vues dans l’espace-temps.

Les occupations de l’ego-véhicule et de l’autre véhicule sont respectivement dessinées en rouge et en bleu et les obstacles fixes sont en vert. Les deux véhicules ont un mouvement rectiligne uniforme. À gauche, la vitesse de l’autre véhicule est trop rapide et conduit à une collision avec l’ego-véhicule. À droite, l’autre véhicule se déplace plus lentement, les occupations des deux véhicules ne s’intersectent pas et aucun de risque de collision n’est détecté.

3.1.1.3 Conclusion

Il est donc possible d’utiliser une représentation géométrique dans l’espace-temps pour vérifier si une trajectoire est libre en connaissant l’environnement statique et en ayant une prévision de l’évolution de l’ensemble des autres agents voisins de l’ego-véhicule au cours du temps.

Nous envisagerons dans la partie 3.3 les façons de pallier les incertitudes de prévision du mouvement des autres véhicules, et les contraintes à introduire pour qu’une trajectoire soit bien suivie par l’ego-véhicule.

3.1.2 Décomposition chemin-vitesse

La recherche d’une trajectoire dans l’espace-temps sans collision est connue pour être un problème NP-difficile [Canny et Reif, 1987].

Pour simplifier le problème de recherche de trajectoire dans l’espace-temps, nous pourrions diminuer la taille de l’espace de recherche en ajoutant des contraintes qui limitent les configurations atteignables par l’ego-véhicule (RS). L’objectif est de réduire l’espace de recherche à un espace à deux dimensions (une surface) pour résoudre le problème.

Pour un ensemble de contraintes $\mathcal{C}(X_{ego})$:

$$X_{ego} \in RS \iff \mathcal{C}(X_{ego}) \quad (3.21)$$

Pour rappel, il s’agit de décomposition chemin-vitesse, car la méthode introduite dans [Kant et Zucker, 1986] propose de contraindre l’ego-véhicule à suivre un

chemin de référence et de calculer la vitesse le long de celui-ci. C'est ce que nous allons décrire dans la section 3.1.2.1. Il est aussi possible de contraindre la vitesse de l'ego-véhicule et de calculer un chemin atteignable. C'est ce que nous verrons par la suite dans la section 3.1.2.2.

3.1.2.1 Planification de la vitesse

Plaçons nous dans le cas où un chemin de référence est fixé. L'espace de recherche est ici réduit à la surface de l'espace-temps : "chemin de référence-temps" (TP) par deux contraintes.

La première contrainte est le suivi du chemin de référence. Nous notons le chemin de référence \mathcal{P}_{ref} composé des segments :

$$\mathcal{P}_{ref} = \{\mathcal{S}_{ref,i} | i \in [O, N_{\mathcal{P}_{ref}} - 1]\} \quad (3.22)$$

où $N_{\mathcal{P}_{ref}}$ est le nombre de points dans le chemin \mathcal{P}_{ref} . Nous notons $\theta_{ref,i}$ l'orientation du segment i . Sous cette contrainte, une configuration de l'ego-véhicule $X_{ego}(t)$ projeté sur le plan ($t = 0$) est forcément sur le chemin de référence \mathcal{P}_{ref} . L'ego-véhicule avec la configuration $X_{ego}(t)$ doit aussi avoir la même orientation que le chemin de référence. Nous pouvons écrire cette contrainte $\mathcal{C}_{ref}(X_{ego}(t))$ de la manière suivante :

$$\mathcal{C}_{ref}(X_{ego}(t)) \iff \exists i | ((x_{ego}(t), y_{ego}(t)) \in \mathcal{S}_{ref,i} \ \& \ \theta_{ego}(t) = \theta_{ref,i}) \quad (3.23)$$

La deuxième contrainte est le respect de la vitesse maximale par l'ego-véhicule. Nous noterons cette vitesse maximale v_{max} . De plus, l'ego-véhicule ne peut pas revenir en arrière, et donc ne peut pas avoir une vitesse négative. Cette contrainte s'écrit $\mathcal{C}_{max}(X_{ego}(t))$ et porte sur la vitesse de l'ego-véhicule $v_{ego}(t)$. Celle-ci est construite à partir de la dérivée des composantes spatiales de la configuration de l'ego-véhicule.

$$v_{ego}(t) = \sqrt{\dot{x}_{ego}^2 + \dot{y}_{ego}^2} \quad (3.24)$$

$$\mathcal{C}_{max}(X_{ego}(t)) \iff 0 \leq v_{ego}(t) \leq v_{max} \quad (3.25)$$

Les RS de l'ego-véhicule correspondent dans ce cas à la surface en rose de la figure 3.4.

Pour réellement réduire la dimension, nous utilisons une variable du chemin de référence appelée abscisse curviligne et notée s_{ref} . Cette variable correspond à la longueur du chemin par rapport à son premier point. Il est nécessaire pour réduire la dimension qu'une abscisse curviligne corresponde à un unique point du chemin.

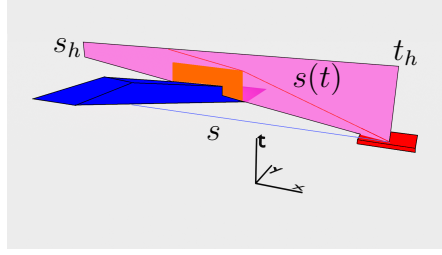


FIGURE 3.4 – Planification de vitesse dans un environnement dynamique. Vue dans l'espace-temps.

La boîte englobante de l'ego-véhicule n'est représentée qu'au temps $t = 0$. Nous avons dessiné en violet l'espace de recherche qui est ici l'espace chemin-temps. Les TP obstacles sont dessinés en marron. Ils représentent l'occupation des obstacles dans le sous-espace.

Ainsi un point des RS peut être défini par son abscisse curviligne le long du chemin de référence et par le temps. Il existe donc une application bijective entre (s_{ref}, t) et les RS. Cela implique de faire l'hypothèse que le chemin de référence \mathcal{P}_{ref} ne reboucle pas sur lui-même. D'un point de vue mathématique, nous voulons que le sous-espace TP respecte :

$$\exists f | f((s, t)) \in RS \quad (3.26)$$

où f est une application bijective de l'espace-temps vers le sous-espace.

Cette espace de recherche est bien un sous-espace de l'espace-temps qui a pour dimensions (s_{ref}, t) .

Nous pouvons représenter les configurations de l'ego-véhicule dans ce plan :

$$X_{ego} = (s_{ego}, t_{ego}) \quad (3.27)$$

La méthode détaillée dans la section 3.3 utilise ces RS.

3.1.2.2 Planification d'un chemin

L'ensemble des positions atteignables par l'ego-véhicule à vitesse donnée peut aussi être décrit comme une surface dans l'espace-temps. La surface dépendra des hypothèses faites sur la cinématique de l'ego-véhicule.

Sans perte de généralité et pour plus de clarté, nous considérons dans cette section que l'ego-véhicule est un véhicule holonome qui se déplace à vitesse constante v_{ego} .

Nous pouvons écrire cette contrainte $\mathcal{C}_{holo}(X_{ego}(t))$ en fonction de la distance au point de départ, *i.e.* $(x_{ego}(0), y_{ego}(0))$:

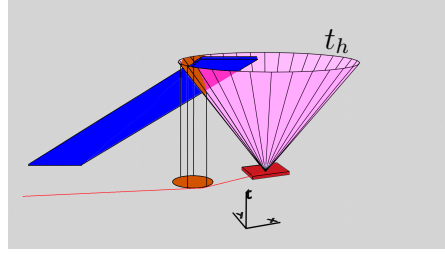


FIGURE 3.5 – Planification de chemin dans un environnement dynamique. Vue dans l'espace-temps.

La boîte englobante de l'ego-véhicule n'est représentée qu'au temps $t = 0$. Nous avons dessiné en violet l'espace de recherche qui est ici l'espace des RS pour un véhicule holonome pouvant se déplacer dans n'importe quelle direction à vitesse constante. L'espace de recherche est donc un cône ayant pour sommet la position initiale de l'ego-véhicule et pour rayon $v_{ego} * t$. L'occupation des obstacles représentée dans le sous-espace est dessinée en marron. Cette représentation de l'occupation des obstacles est projetée sur le plan $E(t = 0)$ pour donner les RIS (Reachable Interaction Sets) [de Beaucorps *et al.*, 2018].

$$\mathcal{C}_{holo}(X_{ego}(t)) \iff \frac{dist((x_{ego}(t), y_{ego}(t)), (x_{ego}(0), y_{ego}(0)))}{t} = v_{ego} \quad (3.28)$$

où $dist(p_1, p_2)$ est la fonction qui renvoie la distance entre p_1 et p_2 .

Les RS de l'ego-véhicule correspondent dans ce cas au cône en rose de la figure 3.5. Cet espace de recherche est bien un sous-espace de l'espace-temps à deux dimensions. De même que précédemment, pour réduire la dimension, il est nécessaire qu'à un point $p = (x, y)$ corresponde un point des RS. Il est donc nécessaire que la surface des RS ne reboucle pas sur elle-même. Cela n'est pas le cas pour un véhicule holonome, mais ce point sera discuté pour des véhicules non-holonomes dans la section 3.2.

Cela peut aussi s'écrire par l'existence d'une application bijective entre (x, y) et les RS :

$$\exists f | f((x, y)) \in RS \quad (3.29)$$

où f est une application bijective de l'espace-temps vers le sous-espace.

Nous pouvons alors représenter les configurations de l'ego-véhicule dans le plan (x, y) :

$$X_{ego} = (x_{ego}, y_{ego}) \quad (3.30)$$

Cela revient à projeter la surface des RS sur le plan $(t = 0)$. Cette représentation sera utilisée par la suite dans la section 3.2 qui utilise ces RS.

3.1.3 Calculs géométriques

Pour trouver une trajectoire libre dans ces sous-espaces, nous avons besoin de connaître l'occupation des différents obstacles dans ces sous-espaces.

L'occupation totale des obstacles dans l'espace-temps est \mathcal{O}_{obs} .

L'occupation des obstacles dans le sous-espace des RS correspond à l'intersection de l'occupation des obstacles dans l'espace \mathcal{O}_{obs} et de l'occupation de l'ego-véhicule cumulée en chaque point du sous-espace (*i.e.* en chaque point des RS) :

$$\mathcal{O}_{obs(RS)} = \mathcal{O}_{obs} \cap \mathcal{O}_{ego}(RS) \quad (3.31)$$

où :

$$\mathcal{O}_{ego}(RS) = \bigcup_{X_{ego}(t) \in RS} \mathcal{O}_{ego}(X_{ego}(t)) \quad (3.32)$$

En utilisant le fait que $\mathcal{O}_{obs} = \mathcal{O}_{statique} \cup \mathcal{O}_{others} \cup \mathcal{O}_{virtuel}$, et en ne considérant pas pour l'instant les obstacles virtuels ($\mathcal{O}_{virtuel} = \emptyset$), nous obtenons :

$$\mathcal{O}_{obs(RS)} = (\mathcal{O}_{statique} \cap \mathcal{O}_{ego}(RS)) \cup (\mathcal{O}_{others} \cap \mathcal{O}_{ego}(RS)) \quad (3.33)$$

Les calculs d'intersection seront différents selon le type de planification envisagé pour l'ego-véhicule : planification de vitesse sur un chemin de référence ou planification de chemin en connaissant un profil de vitesse.

3.1.3.1 Planification de la vitesse

Le chemin de référence est libre vis-à-vis des obstacles statiques. Nous avons donc :

$$\mathcal{O}_{statique} \cap \mathcal{O}_{ego}(RS) = \emptyset \quad (3.34)$$

Il n'est donc pas nécessaire de prendre en compte l'occupation des obstacles statiques lors de la recherche de vitesse, il suffit de considérer l'occupation des autres véhicules en mouvement. Nous appelons TP obstacles [Johnson et Hauser, 2012] les obstacles dans le sous-espace TP, c'est-à-dire les configurations $X_{ego}(t)$ de l'ego-véhicule des RS telles qu'il y ait une intersection entre l'occupation de l'ego-véhicule dans cette configuration et l'occupation d'un obstacle. Nous pouvons définir les TP obstacles comme suit :

$$X_{ego}(t) \in TP_{obstacles} \iff \mathcal{O}_{others} \cap \mathcal{O}_{ego}(RS) \neq \emptyset \quad (3.35)$$

3.1.3.2 Planification d'un chemin

Plaçons nous dans le cas où le profil de vitesse est connu. Comme nous l'avons dit précédemment, les RS pour la planification de chemin peuvent être représentés projetés sur le plan $E(t = 0)$. Nous allons donc projeter l'occupation des obstacles dans le sous-espace puis le projeter sur le plan $(t = 0)$. Nous appelons RIS (Reachable Interaction Sets) [de Beaucorps *et al.*, 2018] les obstacles dans le sous-espace des RS à vitesse déterminée qui ont été projetés sur le plan $E(t = 0)$. Ce sont les zones d'interactions potentielles de l'ego-véhicule avec les obstacles mobiles, connaissant la vitesse de l'ego-véhicule.

$$X_{ego} \in RIS \iff (\mathcal{O}_{obs(RS)} \cap \mathcal{O}_{ego}(RS)).E(t = 0) \neq \emptyset \quad (3.36)$$

Or, en décomposant avec l'équation (3.33) nous obtenons :

$$\begin{aligned} X_{ego} \in RIS \iff & (\mathcal{O}_{statique} \cap \mathcal{O}_{ego}(RS)).E(t = 0) \\ & \cup (\mathcal{O}_{others} \cap \mathcal{O}_{ego}(RS)).E(t = 0) \neq \emptyset \end{aligned} \quad (3.37)$$

Comme les obstacles statiques ne dépendent pas du temps, nous avons :

$$\begin{aligned} (\mathcal{O}_{statique} \cap \mathcal{O}_{ego}(RS)).E(t = 0) = & \mathcal{O}_{statique}.E(t = 0) \\ & \cap \mathcal{O}_{ego}(RS).E(t = 0) \neq \emptyset \end{aligned} \quad (3.38)$$

La projection de l'occupation des obstacles statiques ou de l'occupation de l'ego-véhicule sur les RS ne nécessite pas vraiment de calcul supplémentaire. Il suffit en effet d'omettre la dimension temporelle. Pour simplifier le problème, nous considérons donc les obstacles statiques à part. Ils sont ensuite ajoutés aux RIS calculées avec les obstacles dynamiques.

Nous avons donc :

$$X_{ego} \in RIS \iff X_{ego} \in RIS_{statique} \cap RIS_{others} \quad (3.39)$$

où $RIS_{statique}$ s'écrit :

$$RIS_{statique} = \mathcal{O}_{statique}.E(t = 0) \cap \mathcal{O}_{ego}(RS).E(t = 0) \quad (3.40)$$

et RIS_{others} s'écrit :

$$RIS_{others} = (\mathcal{O}_{others} \cap \mathcal{O}_{ego}(RS)).E(t = 0) \quad (3.41)$$

3.1.4 Conclusion

Une approche géométrique peut permettre le calcul d'une trajectoire sans collision avec l'environnement quand nous contraignons les positions atteignables de l'ego-véhicule *a priori*. Deux méthodes sont envisagées :

- le calcul de la vitesse en fixant un chemin de référence ;
- le calcul d'un chemin en fixant un profil de vitesse.

Ces méthodes transforment le problème à trois dimensions en un problème à deux dimensions.

3.2 Planification de chemin à vitesse connue : méthode RIS et MMRIS

Dans ce qui suit, nous supposons connaître la carte de l'environnement. L'environnement est composé d'obstacles statiques représentant des zones interdites pour l'ego-véhicule. L'environnement comprend également d'autres véhicules en mouvement. L'ego-véhicule est supposé circuler avec un profil de vitesse connu. L'objectif est de calculer un chemin admissible pour l'ego-véhicule tel que celui-ci puisse aller d'une position de départ jusqu'à un point d'arrivée sans entrer en collision avec les différents obstacles statiques ou dynamiques de l'environnement. La méthode présentée dans cette section est une extension des RIS rapidement définis dans la section 3.1.3.2 et que nous allons détailler dans la section 3.2.1.

La section 3.2.2 sera consacrée à une utilisation des RIS pour la navigation d'un robot mobile. Nous présenterons dans la section 3.2.2.1 l'adaptation des RIS au mouvement des véhicules non holonomes. Nous supposerons en premier lieu que les positions et les vitesses des autres véhicules sont connues pendant une courte période de temps. Puis, dans la section 3.2.2.2, nous étendrons notre approche à d'autres types d'obstacles dont la vitesse n'est pas parfaitement connue, que nous appellerons obstacles stochastiques.

Enfin nous présenterons dans la section 3.2.3 les extensions faites sur les RIS pour le milieu urbain et les MMRIS.

Les zones représentées par les RIS ou les MMRIS sont considérées comme des obstacles statiques par les algorithmes de planification de chemin. Les sections 4.1 et 4.3 présentent les résultats de différentes méthodes de planification sur les RIS ou les MMRIS.

3.2.1 Définition des RIS (Reachable Interaction Sets)

Comme nous l'avons déjà décrit dans la section 3.1, la décomposition chemin-vitesse peut se résumer comme suit :

- nous posons une hypothèse sur le chemin ou sur la vitesse de l’ego-véhicule ;
- nous en déduisons un espace de recherche ;
- nous représentons l’occupation des obstacles dans cet espace de recherche ;
- nous calculons enfin une planification dans cet espace de recherche réduit en évitant les obstacles.

Ici, nous posons l’hypothèse d’un profil de vitesse pour l’ego-véhicule comme nous l’avons vu dans la section 3.1.2.2. En fonction d’un modèle cinématique et/ou d’une ou plusieurs contraintes sur le mouvement de l’ego-véhicule, nous en déduisons des RS ou positions atteignables. Celles-ci sont représentées dans l’espace-temps. Ce sont ces RS qui composent le sous-espace de recherche. Nous avons vu l’exemple d’un véhicule holonome dans la section 3.1.2.2 où les RS forment une surface (un cône) dans l’espace de configuration.

Les RIS sont une représentation des zones d’interactions futures possibles avec les obstacles. Elles sont représentées comme des obstacles statiques dans le plan d’espace-temps $E(t = 0)$. Elles sont associées aux configurations dans lesquelles l’ego-véhicule serait en collision avec au moins un obstacle de l’environnement. Ces configurations de collision sont ensuite projetées sur le plan $E(t = 0)$ pour former les RIS. On obtient les configurations de collision en calculant les intersections entre l’occupation de l’ego-véhicule en chaque point des RS ($\mathcal{O}_{ego}(RS)$) et l’occupation prévue des obstacles (\mathcal{O}_{obs}). Il existe une bijection entre les configurations de collision et les RIS (équation (3.29)). Nous appellerons donc aussi RIS l’ensemble des configurations de collision (avant projection).

3.2.2 Extensions des RIS pour la robotique

3.2.2.1 Intégration de contraintes de mouvement pour les véhicules non-holonomes

Pour un véhicule holonome se déplaçant à vitesse constante, les RS sont sur un cône de rayon $v_{ego}t$ où v_{ego} est la vitesse du véhicule et t le temps (cf. figure 4.3, ① et ②). Cette surface peut être définie pour d’autres types de véhicules ([Fedotov *et al.*, 2011]).

a) Modèle de véhicule : véhicule de Dubins Pour construire les RIS en robotique, nous utiliserons un modèle de véhicule appelé le véhicule de Dubins.

Le véhicule de Dubins a été choisi pour sa simplicité et permet de bien décrire les contraintes de déplacement d’un véhicule non-holonome dans une première approche. Le modèle considère uniquement le rayon de braquage du véhicule. Le véhicule est ramené à un point qui est contraint d’évoluer sur un cercle dont le rayon varie en fonction de l’angle de braquage des roues que nous noterons $u(t)$. Les RS sont définies par le système suivant (cf. figure 3.6) :

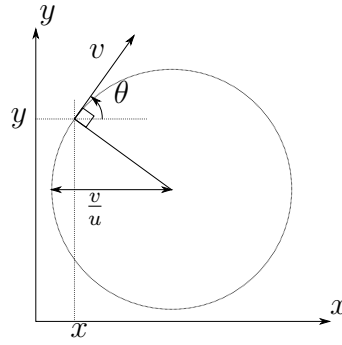


FIGURE 3.6 – Véhicule de Dubins.

Le véhicule a une vitesse v . La commande $u(t)$ fait varier le rayon de courbure $r_c(t)$ de la trajectoire avec la relation $r_c(t) = v(t)/u(t)$.

$$\begin{cases} \dot{\theta}(t) &= u(t) \\ \dot{x}(t) &= v(t) \sin(\theta(t)) \\ \dot{y}(t) &= v(t) \cos(\theta(t)) \end{cases} \quad (3.42)$$

En plus de ce modèle cinématique, une contrainte physique non-holonome est ajoutée au modèle pour représenter l'angle de braquage maximal de l'ego-véhicule. L'angle de braquage $u(t)$ du véhicule doit respecter :

$$-u_{max} < u(t) < u_{max} \quad (3.43)$$

u_{max} étant un paramètre dépendant du véhicule représentant l'angle de braquage maximal.

b) Construction de RS : angle de braquage constant Le modèle permet de calculer les RS en fonction des variables de commandes du modèle sur un horizon de temps t_h donné. Cet horizon temporel est la durée considérée par l'approche pour la planification. Toutes les prévisions s'arrêtent au-delà de cet horizon. Pour les deux modèles, les variables de commande sont la vitesse $v(t)$ et l'angle de braquage des roues $u(t)$.

Comme nous l'avons déjà vu, la méthode fixe la vitesse par hypothèse. Pour la suite, nous utiliserons la loi suivante :

pour n'importe quelle vitesse inférieure à la vitesse maximale autorisée, l'ego-véhicule accélère jusqu'à la vitesse maximale. Nous avons donc le comportement suivant pour la vitesse :

$$v(t) = \min(v(0) + a_{max}t, v_{max}) \quad (3.44)$$

où a_{max} et v_{max} sont des paramètres du véhicule représentant respectivement l'accélération et la vitesse maximales. Nous pouvons alors obtenir les RS pour une commande $u(t)$ qui dépend de la vitesse initiale de l'ego-véhicule $v(0)$.

Pour l'angle de braquage $u(t)$, nous souhaitons que les RS décrivent bien l'ensemble des positions atteignables par l'ego-véhicule tout en gardant une complexité acceptable.

Le compromis que nous avons choisi est de faire une hypothèse supplémentaire sur le rayon de braquage $u(t)$: il sera constant en robotique sur tout l'horizon temporel t_h .

Nous considérons que le véhicule se déplace avec une commande u constante sur l'ensemble de l'horizon temporel. Pour une commande u constante donnée, nous pouvons calculer, à partir de la position initiale du véhicule, la trajectoire résultante $\mathcal{T}_{ego}(u)$. L'ensemble des trajectoires pour toutes les valeurs de commandes entre u_{max} et $-u_{max}$ forme les RS.

$$X_{ego} \in RS \iff X_{ego} \in \int_{-u_{max}}^{u_{max}} \mathcal{T}_{ego}(u) \quad (3.45)$$

avec, pour rappel, $X_{ego} = (x, y, t)$.

La surface des RS est représentée dans la figure 3.7.

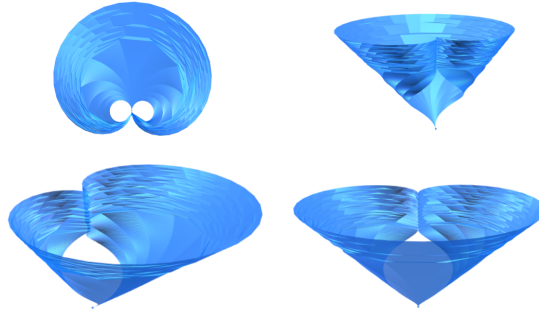


FIGURE 3.7 – Maillage obtenu pour la surface de Dubins.

3.2.2.2 Obstacles stochastiques

Nous avons vu comment construire les RIS à partir d'occupations d'obstacles sous forme de collection de polygones. Pour construire l'occupation de l'obstacle, il nous faut soit connaître avec certitude son comportement futur, soit faire une hypothèse forte (par exemple, supposer qu'il va garder une vitesse constante). Pour pallier les erreurs de prédiction, une solution est d'ajouter des marges de sécurité. Toutefois, la différence entre la prédiction et la réalité dérive très rapidement si l'obstacle ne respecte pas l'hypothèse faite sur son comportement. Pour limiter

cette dérive, il faudra alors planifier à nouveau régulièrement. Une autre solution pour limiter cette dérive serait d'utiliser un modèle comportemental plus approprié. Quand le mouvement des obstacles mobiles n'est pas connu avec certitude, celui-ci peut être modélisé par un modèle stochastique. La construction des RIS peut être étendue pour tenir compte de ce type de modélisation.

Pour prédire l'occupation d'un obstacle dans le futur, nous allons lui associer un modèle comportemental. Nous supposons que nous pouvons associer à chaque obstacle un modèle stochastique d'obstacle adapté à son comportement. Nous utilisons une prédiction des positions futures basée sur la méthode de Monte-Carlo comme dans l'approche de [Chiang *et al.*, 2015b]. Nous posons un modèle d'obstacles stochastiques comme suit :

- l'obstacle se déplace sur un chemin défini $\mathcal{P}_{obstacle}$;
- à chaque itération, nous sélectionnons une vitesse de façon aléatoire parmi κ vitesses possibles.

Cela correspond au système suivant :

$$\begin{cases} v_{obstacle}(m, t) = random(v_1, \dots, v_\kappa) \\ s_{obstacle}(m, t) = s_{obstacle}(m, t - \Delta t) + v_{obstacle}(m, t) \times \Delta t \end{cases} \quad (3.46)$$

où $v_{obstacle}(m, t)$ est la vitesse de l'obstacle à l'instant t lors de la $m^{\text{ème}}$ itération de Monte-Carlo, (v_1, \dots, v_κ) sont les κ vitesses possibles de l'obstacle.

Connaissant par ailleurs l'abscisse curviligne $s_{obstacle}(m, t)$ et le chemin $\mathcal{P}_{obstacle}$ nous déduisons $X_{obstacle}(m, t)$.

$X_{obstacle}(m, t)$ est la position de l'obstacle à l'instant t lors de la $m^{\text{ème}}$ itération de Monte-Carlo et Δt est le pas de temps.

Nous en déduisons alors l'occupation de l'obstacle à la $m^{\text{ème}}$ itération de Monte-Carlo :

$$\mathcal{O}_{obstacle}(m, t) = \mathcal{P}_{obstacle}(m, t) \quad (3.47)$$

L'objectif est de construire une occupation pour l'obstacle stochastique qui approxime son occupation réelle. Nous noterons cette occupation stochastique $obstacle(t)$.

Tout d'abord, nous redéfinissons le polygone d'occupation pour chaque itération de Monte-Carlo par une fonction d'occupation. C'est une fonction qui vaut 1 dans le polygone et 0 à l'extérieur. Nous l'écrivons :

$$\mathcal{P}_{obstacle}(x, y, m, t) = (x, y) \in \mathcal{P}_{obstacle}(m, t) \quad (3.48)$$

Ensuite, nous prenons la moyenne de la fonction d'occupation de toutes les itérations. Pour N_{mc} itérations, nous obtenons une approximation de la probabilité



FIGURE 3.8 – Images représentant la probabilité de présence d’un obstacle à trois instants différents dans le cas d’un mouvement le long d’une ligne droite. Plus un pixel tend vers le noir, plus sa probabilité d’être occupé par l’obstacle est élevée.

de présence de l’obstacle stochastique ayant pour modèle le système d’équations (3.46) (cf. figure 3.8) :

$$\rho_{occupation}(x, y, t) = \frac{1}{N_{mc}} \sum_{m=1}^{N_{mc}} P(x, y, m, t) \quad (3.49)$$

Pour intégrer des obstacles stochastiques respectant l’équation (3.46) dans la méthode RIS, nous décidons de seuiller la probabilité de présence pour être capable de définir un intérieur et un extérieur aux obstacles. Nous utilisons un seuil dynamique inspiré de celui utilisé dans DRT [[Chiang et al., 2017b](#)]. Le seuil est défini comme ci-dessous :

$$\rho_{seuil}(t) = \begin{cases} \rho_{min}, & t \leq t_A \\ \alpha(t), & t_B > t > t_A \\ \rho_{max}, & t \geq t_B \end{cases} \quad (3.50)$$

avec :

$$\alpha(t) = \frac{\rho_{max} - \rho_{min}}{t_B - t_A} \times (t - t_A) + \rho_{min} \quad (3.51)$$

Cela donne la courbe suivante :

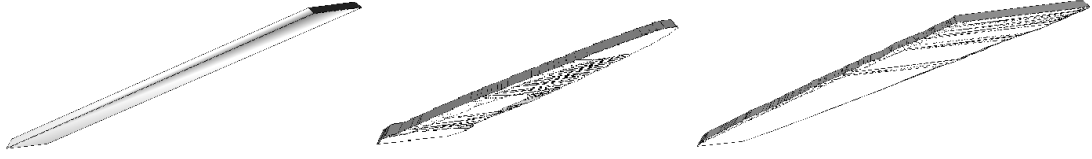


FIGURE 3.10 – Occupation d’un obstacle à vitesse constante à gauche et occupations de deux obstacles stochastiques à droite dans l’espace spatio-temporel. *Remarque :* La forme de l’obstacle stochastique dépend du seuil défini dans l’équation (3.51). En effet, nous avons un seuil dynamique au centre et un seuil constant à droite.

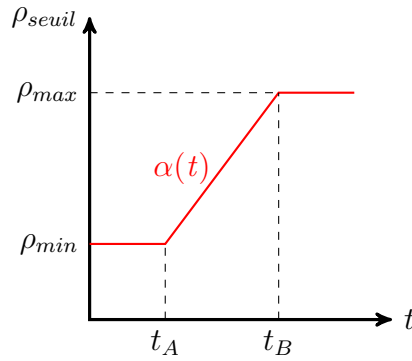


FIGURE 3.9 – Seuil dynamique permettant de définir à quelle probabilité se trouve la limite de l’obstacle.

$\rho_{seuil}(t)$ est le seuil dynamique défini comme une fonction du temps ayant pour paramètres : ρ_{min} le seuil minimal, ρ_{max} le seuil maximal, t_A le temps durant lequel le seuil est au minimum et $t_B - t_A$ le temps de montée du seuil (cf. figure 3.2.2.2).

Nous pouvons appliquer le seuil $\rho_{seuil}(t)$ à $\rho_{occupation}(x, y, t)$ pour retrouver une fonction d’occupation.

$$\mathcal{P}_{obstacle}(x, y, t) = \rho_{occupation}(x, y, t) \geq \rho_{seuil}(t) \quad (3.52)$$

Sous certaines conditions, nous pouvons construire des polygones d’occupation qui représente l’occupation probable de l’obstacle stochastique à partir de cette fonction d’occupation.

Nous pouvons donc construire l’équivalent stochastique de l’occupation $\tilde{\mathcal{O}}_{obstacle}$ en reliant les polygones obtenus de la même manière que pour les obstacles non stochastiques. En calculant l’intersection de ceux-ci avec les RS de l’ego-véhicule, nous obtenons une version stochastique des RIS.



FIGURE 3.11 – Dessin représentant le RIS sans marge en bleu et la marge du RIS en marron. Le RIS sans marge est inclus dans le RIS avec marge.

Remarque : les obstacles stochastiques présentés dans cette section peuvent être adaptés pour intégrer d'autres types de comportements. Pour ce faire, il suffit de modifier l'équation (3.46) introduite pour pouvoir comparer notre approche à celle de [Chiang *et al.*, 2015b].

3.2.2.3 Gestion des cas d'urgences

L'utilisation des RIS dans une méthode de planification peut mener à des situations que nous appelons cas d'urgence. Nous avons donc introduit des heuristiques pour résoudre ces cas particuliers :

- lorsque l'ego-véhicule est dans un RIS, le risque de collision est plus important qu'en dehors d'un RIS. Il faut donc en sortir au plus vite.
- lorsque aucun chemin n'est trouvé, l'ego-véhicule s'arrête. Or si l'ego-véhicule est à l'arrêt, les RS calculés ne sont plus valables car l'ego-véhicule ne respecte plus le comportement prévu par l'équation (3.44). Il faut donc recalculer les RS à l'arrêt.

a) Cas de l'ego-véhicule dans un RIS La méthode de planification de mouvement vise à créer un chemin qui ne passe pas par les RIS, mais il arrive que l'ego-véhicule se retrouve à l'intérieur d'un RIS. Cela est dû au fait que l'horizon de l'ego-véhicule est limité et que certains obstacles se déplacent à grande vitesse. Il est donc possible que l'ego-véhicule commence dans un RIS lors de la planification. Or, nous avons vu qu'il était nécessaire d'ajouter aux RIS une marge pour pallier l'incertitude de la prévision. Nous calculons donc deux ensembles de RIS différents : d'abord, les RIS classiques qui possèdent une marge (cf. figure 3.11), puis les RIS "d'urgence", sans marge.

Nous avons hiérarchisé les zones de RIS pour éviter les comportements dangereux. Cette hiérarchisation est illustrée par la figure 3.12.

Pour trouver le chemin le plus court pour sortir d'un RIS avec ou sans marge, nous cherchons le point libre le plus proche de l'ego-véhicule, c'est-à-dire le point

p_{libre} situé sur le bord du RIS le plus proche du centre de l'ego-véhicule. Le segment de droite qui relie le centre de l'ego-véhicule et p_{libre} représente le chemin de sortie.

Cette méthode n'est valable que pour un véhicule holonome. Dans le cas d'un véhicule non-holonome, nous traçons un certain nombre de trajectoires à braquage constant u respectant les contraintes défini par le système d'équations (3.42). La trajectoire choisie est celle qui sort le plus rapidement du RIS.

Plusieurs cas sont alors possibles (voir la description dans l'algorithme 1).

Algorithme 1 : Construction du chemin.

```
if ego-véhicule dans un RIS avec marge then
  if ego-véhicule dans un RIS sans marge then
    Trouve le chemin le plus court pour sortir du RIS sans marge
  else
    Trouve le chemin le plus court pour sortir du RIS avec marge
    if Le chemin pour sortir du RIS avec marge passe par un RIS sans marge
      then
         $PATH \leftarrow []$ 
      end if
    end if
     $PATH \leftarrow [Chemin\ de\ sortie\ du\ RIS]$ 
  else
    Trouve le chemin le plus court pour atteindre le but sans passer par les RIS
     $PATH \leftarrow$ 
    [Chemin le plus court pour atteindre le but sans passer par les RIS]
  end if
```

b) Cas de l'ego-véhicule à l'arrêt Lorsque aucun chemin n'a été trouvé, nous décidons que l'ego-véhicule doit s'arrêter. Cela peut correspondre au comportement voulu, mais, il faut s'assurer que la position d'arrêt est sûre, car les RS de l'ego-véhicule à l'arrêt ne correspondent pas à ceux qui ont été calculés. Nous avons vu que la vitesse était prédéterminée par l'équation (3.44). En effet, quand l'ego-véhicule est à l'arrêt, les RS sont sur une droite verticale passant par sa position actuelle dans l'espace-temps.

Pour vérifier que la position est sûre, nous vérifions que cette droite n'intersecte pas l'occupation des obstacles. Si l'intersection est vide, la position est sûre. L'ego-véhicule peut donc rester immobile jusqu'à l'instant suivant où un calcul de chemin sera effectué.

Dans le cas contraire, la position n'est pas sûre. Il faut alors générer un chemin pour quitter cette position. Une heuristique doit être introduite et nous en avons



FIGURE 3.12 – Différents cas d’urgence avec les RIS.

De gauche à droite, nous avons :

- Chemin (en rouge) pour l’ego-véhicule (en rouge foncé) en dehors d’un RIS avec marge.
- Chemin (en rouge) pour l’ego-véhicule (en rouge foncé) dans la marge d’un RIS.
- Chemin (en rouge) passant dans un RIS sans marge pour l’ego-véhicule (en rouge foncé) dans la marge d’un RIS. Le chemin n’est pas valide
- Chemin (en rouge) pour l’ego-véhicule (en rouge foncé) dans un RIS sans marge.

identifié deux qui semblent satisfaisantes :

Projection : la première est de lister les obstacles dangereux dans le cas où l’ego-véhicule resterait à l’arrêt. Pour tous ces obstacles, nous projetons l’ensemble de leurs occupations sur le plan à $t = 0$ et nous ajoutons ces projections aux RIS. Sachant que la position n’est pas sûre, l’ego-véhicule se retrouve alors dans un RIS dont il va sortir au plus vite en respectant l’heuristique du paragraphe (a) présentée ci-dessus (cf. figure 3.12).

Horizon : la deuxième solution est de réduire l’horizon de prédiction dans le but de réduire les RIS jusqu’à trouver un chemin. La position actuelle n’étant pas sûre, l’ego-véhicule ne peut pas rester à l’arrêt. L’objectif de cette heuristique est de trouver un chemin en réduisant les exigences de planification au court terme.

Les deux approches ont leurs avantages et leurs inconvénients. La première permet de forcer l’ego-véhicule à sortir de la zone de danger sans réduire l’horizon de prédiction, mais le chemin choisi est moins bon que le chemin trouvé en réduisant l’horizon. Cependant, une réduction trop forte de l’horizon rend l’ego-véhicule aveugle à l’environnement. Les deux versions ont été développées puis testées sur un certain nombre de scénarios dans le cas d’un véhicule holonome comme dans le cas d’un véhicule non-holonome. Cette évaluation est présentée en section 4.1.2.4.

3.2.3 Extensions des RIS pour le milieu urbain

L’ego-véhicule évoluant dans un milieu urbain est soumis à certaines contraintes liées aux mouvements plausibles d’un véhicule, aux voies de circulation, au confort

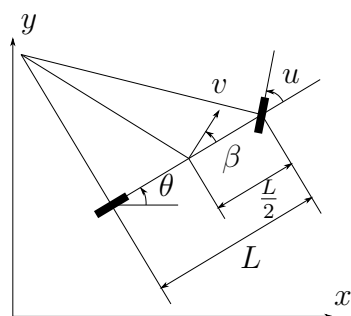


FIGURE 3.13 – Modèle bicyclette.

des utilisateurs, etc. En tenant compte de ces éléments, les RIS présentés dans la section précédente peuvent conduire à des trajectoires non réalistes. Nous avons donc décidé de les adapter en nous appuyant entre autres sur les chemins candidats générés à partir de chemins de référence (quand ils existent) ou de considérer plusieurs hypothèses de vitesse pour l’ego-véhicule avec les MMRIS.

3.2.3.1 Intégration des contraintes routières aux mouvements de l’ego-véhicule

a) **Modèle de véhicule : modèle bicyclette** Pour le milieu urbain, nous utilisons un modèle plus fidèle au véhicule réel.

La différence principale avec le modèle vu dans le paragraphe (a) de la section 3.2.2.1 est la prise en compte de la distance entre les roues avant et les roues arrière du véhicule. Cette distance appelée l’empattement sera notée L . Pour simplifier le modèle, nous considérons que le centre de l’ego-véhicule est situé au centre de gravité de l’ego-véhicule, entre les roues avant et les roues arrière (cf. figure 3.13).

En revanche, l’écartement latéral des roues n’est pas pris en compte dans le modèle, d’où le nom de modèle bicyclette. Ce modèle est très fidèle à la cinématique du véhicule, car la conception de la direction dans un véhicule est en général faite pour respecter le modèle bicyclette.

Dans la section 3.4 nous présentons le modèle complet et les simplifications que nous avons faites pour son utilisation.

$$\begin{cases} \dot{x}(t) &= v(t) \sin(\theta(t) + \beta(t)) \\ \dot{y}(t) &= v(t) \cos(\theta(t) + \beta(t)) \\ \dot{\theta}(t) &= v(t) \frac{\tan(u(t)) \cos(\beta(t))}{L} \\ \beta(t) &= \arctan\left(\frac{\tan(u(t))}{2}\right) \end{cases} \quad (3.53)$$

Ce modèle aussi tient compte de l’angle de braquage maximal de l’ego-véhicule. Nous y ajoutons donc l’équation (3.43).

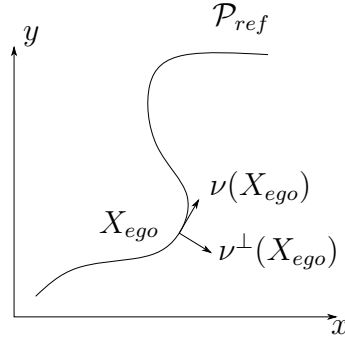


FIGURE 3.14 – Repère attaché à un chemin.

b) Construction de RS : suivi de chemins candidats Dans un environnement urbain, nous pouvons contraindre les RS pour ne conserver que des comportements raisonnables.

Nous choisissons de baser les RS sur un chemin de référence \mathcal{P}_{ref} créé sans tenir compte des obstacles dynamiques.

Pour l'approche urbaine, nous utilisons des chemins candidats pour construire les RS. Les chemins candidats suivent un faisceau construit par transformation du chemin de référence \mathcal{P}_{ref} . Pour former le faisceau, nous allons utiliser un repère lié au chemin. Nous posons :

$$\nu(X_{ref}) = \frac{\dot{X}_{ref}}{|\dot{X}_{ref}|} | X_{ref} \in \mathcal{P}_{ref} \quad (3.54)$$

et $\nu^\perp(X_{ref})$ qui est perpendiculaire à $\nu(X_{ref})$ (cf. figure 3.14). Nous obtenons une ligne du faisceau \mathcal{P}_λ en fonction d'un paramètre de longueur λ tel que :

$$X_\lambda \in \mathcal{P}_\lambda \iff X_\lambda = X_{ref} + \lambda \nu^\perp(X_{ref}) | X_{ref} \in \mathcal{P}_{ref} \quad (3.55)$$

À partir de ce faisceau, nous voulons obtenir un autre faisceau de chemins respectant toutes les équations (3.53) du modèle utilisé. Pour une ligne du faisceau \mathcal{P}_λ , nous souhaitons trouver un chemin \mathcal{T}_λ qui parte de la position initiale de l'ego-véhicule X_0 et qui respecte le modèle cinétique. Pour ce faire, nous utilisons un modèle de suivi de chemin. Nous utilisons pour le suivi le modèle de poursuite. Nous calculons un point devant le véhicule :

$$p_{poursuite}(t) = (x(t), y(t)) + L_{poursuite} \frac{v(t)}{|v(t)|} \quad (3.56)$$

où $L_{poursuite}$ est une distance de réglage de l'approche. Nous trouvons ensuite le point le plus proche de $p_{poursuite}(t)$ sur $\mathcal{P}_{faisceau}(\lambda)$: $p_{cible}(t)$. Nous obtenons une

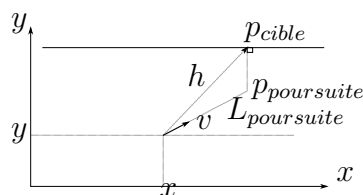


FIGURE 3.15 – Suivi de chemin par poursuite.

direction à suivre $h(t) = p_{cible}(t) - (x(t), y(t))$ (cf. figure 3.15) et donc un angle cible pour l'ego-véhicule $\theta_h(t)$.

Nous remontons à l'angle de braquage des roues cibles en fonction du modèle cinétique. Tout d'abord, nous notons l'écart entre l'angle cible et l'angle de l'ego-véhicule : $\Delta\theta(t) = \theta_h(t) - \theta(t)$. Nous choisissons pour trouver la commande d'approximer $\theta(t)$ par $\Delta\theta(t)$. Nous trouvons donc, pour le modèle de Dubins (équations (3.42)) :

$$u(t) = \Delta\theta(t) \quad (3.57)$$

et pour le modèle bicyclette (équations (3.53)) :

$$\Delta\theta(t) = \frac{v(t)}{L} \frac{\tan(u(t))}{\sqrt{\frac{\tan^2(u(t))}{4} + 1}} \quad (3.58)$$

que nous approximations par :

$$\Delta\theta(t) = \frac{v(t)}{L} \tan(u(t)) \quad (3.59)$$

Nous avons donc :

$$u(t) = \arctan\left(\frac{L(\Delta\theta(t))}{v(t)}\right) \quad (3.60)$$

De la même manière que précédemment, nous finissons par trouver les RS en intégrant l'ensemble des possibilités. Ici, nous restreignons les possibilités en fixant l'épaisseur du faisceau : $-\lambda_{max} \leq \lambda \leq \lambda_{max}$ où λ_{max} est une valeur fixée empiriquement en fonction du scénario et du comportement désiré de l'ego-véhicule.

Nous avons donc :

$$X_{ego} \in RS \iff X_{ego} \in \int_{-\lambda_{max}}^{\lambda_{max}} \mathcal{T}_{ego}(\lambda) \quad (3.61)$$

En pratique, nous créons un certain nombre de chemins parallèles cibles de manière discrète pour avoir une approximation du faisceau.

Nous pouvons alors calculer pour chacun de ces chemins une trajectoire de suivi respectant les contraintes dynamiques de l'ego-véhicule.

L'ensemble des trajectoires constitue une bonne approximation des RS de l'ego-véhicule en tenant compte des contraintes de la route (cf. figure 3.17).

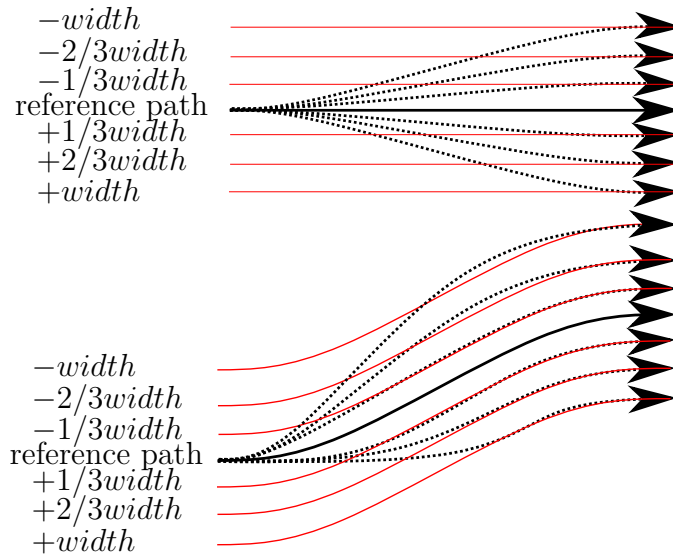


FIGURE 3.16 – Chemins candidats construits à partir des chemins cibles dessinés en rouge.

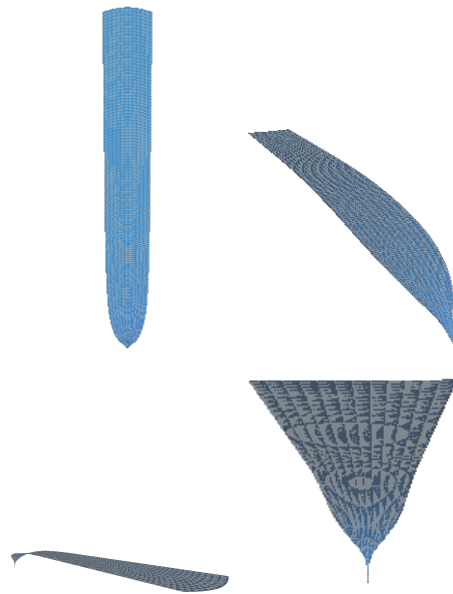


FIGURE 3.17 – Surface des RS pour le suivi de chemins candidats avec le modèle bicyclette.

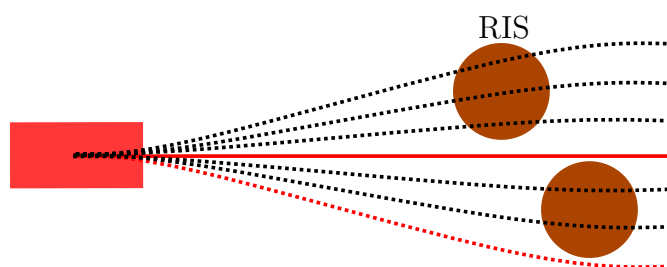


FIGURE 3.18 – Exemple de chemins candidats libres de bout en bout. Les RIS dessinés en marron intersectent les chemins candidats dessinés en noir. Les chemins candidats rouges sont libres et réalisables par l’ego-véhicule.

3.2.3.2 Utilisation des chemins candidats dans la planification

Pour obtenir un comportement plus humain de l’ego-véhicule en milieu urbain, nous avons remarqué qu’il n’était pas utile de calculer une planification avec les RIS si au moins un des chemins candidats servant à constituer les RS était libre (cf. figure 3.18).

S’il existe plusieurs chemins candidats libres, alors nous mettons en place une stratégie de choix adaptée au milieu urbain. Nous sélectionnons un des chemins comme suit :

- si des voies de circulation sont définies dans l’environnement proche de l’ego-véhicule, les chemins candidats les plus proches des voies sont pré-sélectionnés. Parmi ceux-ci, le chemin sur la voie la plus à droite est choisi, ce qui incitera l’ego-véhicule à se rabattre dès que possible ;
- si aucun des chemins candidats n’est suffisamment proche d’une voie ou s’il n’y a pas de voie bien définie, le plus gros groupe de chemins candidats adjacents et libres est pré-sélectionné. Le chemin le plus central du groupe est choisi pour éloigner l’ego-véhicule des obstacles.

En revanche, si aucun chemin candidat n’est libre de bout en bout, nous pouvons trouver une manœuvre à l’aide des RIS en utilisant une méthode de planification (cf. figure 3.19).

3.2.3.3 MMRIS (Multi Modal Reachable Interaction Sets)

Nous avons développé une extension aux RIS en combinant différentes hypothèses de vitesse. Nous avons remarqué que les manœuvres étaient liées à l’intention d’accélération. Trois modes d’action ont été distingués :

- le mode d’accélération où $a_{cible} = a_{max}$ et $v_{cible} = v_{max}$
- le mode de vitesse constante où $a_{cible} = 0$
- le mode de freinage où $a_{cible} = a_{brake}$ et $v_{cible} = \frac{v(0)}{2}$

De chaque mode découle une accélération et une vitesse cible. Ainsi, en connaissant

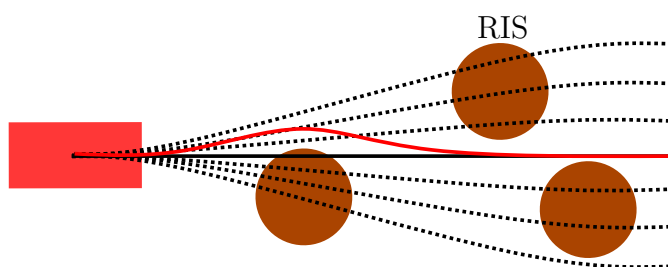


FIGURE 3.19 – Exemple de chemins candidats occupés par les RIS. Aucun chemin candidat n'est libre. Les RIS dessinés en marron intersectent tous les chemins candidats dessinés en noir. Il faut alors intégrer les RIS dans une méthode de planification de chemin.

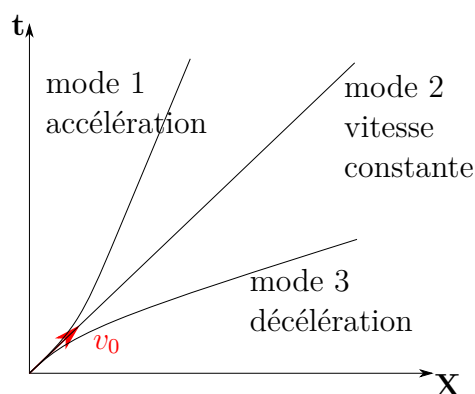


FIGURE 3.20 – Profils de vitesse pour trois modes d'accélération différents.

la vitesse actuelle de l'ego-véhicule, un profil de vitesse peut être obtenu comme illustré dans la figure 3.20.

En utilisant les RIS correspondant aux différents modes d'action, trois planifications de chemins correspondant aux trois modes d'action (accélération, vitesse constante, freinage) sont calculées.

Comme nous l'avons vu dans la section 3.2.3.2, le chemin correspondant à chaque mode peut être un des chemins candidats qui a servi à construire les RIS ou bien le résultat d'une méthode de planification de chemin utilisant les RIS lorsque aucun des chemins candidats n'est libre de bout en bout.

Enfin, la manœuvre donnée par le mode d'action le plus rapide est sélectionnée quand plusieurs solutions sont possibles.

3.2.3.4 Sécurité et confort

a) **Marge de sécurité** Il faut s'assurer qu'une zone autour de la trajectoire de l'ego-véhicule est sûre pour pallier les potentielles erreurs de prévision liées aux

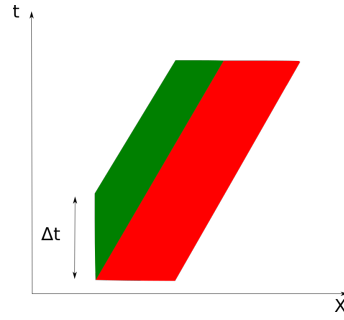


FIGURE 3.21 – Représentation de l'élongation d'un obstacle.

incertitudes et aux approximations de calcul. Pour cela, nous introduisons une marge de sécurité pour agrandir les zones occupées par les obstacles.

Le calcul de la marge de sécurité doit aussi tenir compte du confort des usagers de la route. En effet, il est important de laisser des distances rassurantes entre le véhicule autonome et les autres agents. Nous prendrons la plus grande marge entre la marge de sécurité et la marge de confort.

b) Time-gap De plus, pour des raisons de sécurité et de confort, un écart temporel entre les utilisateurs doit être respecté. Cet écart est appelé time-gap. Légalement d'une valeur de deux secondes en France, il consiste à laisser un temps minimal entre le passage de deux utilisateurs successif sur la même route. En considérant la représentation de l'occupation des obstacles dans l'espace-temps utilisé, cela revient à allonger l'occupation des obstacles de la valeur time-gap vers le futur et le passé.

L'élongation (cf. figure 3.21) d'une durée Δt de l'occupation $V_{obstacle}(t)$ d'un obstacle sera notée :

$$V_{obstacle}^*(t) = stretch(V_{obstacle}(t), \Delta t) \quad (3.62)$$

Cette opération équivaut à :

$$\forall t \in [0, t_{horizon}], V_{obstacle}^*(t) = \sum_{t^*=t}^{t+\Delta t} V_{obstacle}(t^*) \quad (3.63)$$

c) Marge de manœuvre Enfin, pour éviter les blocages dans les mouvements de l'ego-véhicule, nous laissons aussi une marge supplémentaire autour des agents arrêtés permettant à l'ego-véhicule d'effectuer la manœuvre de dépassement.

d) Cooldown Un comportement oscillant de l'ego-véhicule peut aussi nuire à la sécurité ou au confort. Pour réduire les risques d'oscillation, lorsqu'une manœuvre est effectuée, nous limitons les prochaines recherches durant une certaine période de temps. Nous parlons de période de cooldown.

3.2.3.5 Règles d'usage et du code de la route

Pour garantir la sécurité et le respect des lois, l'ego-véhicule doit respecter certaines règles d'usage et le code de la route.

Nous pouvons représenter un certain nombre de règles en modifiant ou contraignant certains éléments de l'approche. Certaines de ces modifications ne sont effectives que sous conditions.

- La première modification concerne la modification des RS à l'approche d'une intersection. Nous limitons les chemins candidats au chemin statique de sorte qu'aucune manœuvre ne soit possible dans une intersection sauf quand l'ego-véhicule est bloqué depuis longtemps par un agent qui n'est pas un feu de signalisation.
- La seconde modification est de contraindre la prévision que nous faisons des autres agents pour débloquer les situations de dépassement. Nous considérons que les usagers peuvent changer de voie pour se rabattre, ce qui évite de les doubler par la droite. En revanche, si nous considérons qu'un usager peut toujours se déporter sur la gauche, nous ne pouvons jamais envisager de le doubler. Pour remédier à cela, le déportement à gauche est contraint dans les prévisions. Il n'est considéré que lorsque nous détectons que l'agent commence sa manœuvre à gauche. Quand l'agent est déjà légèrement déporté, nous estimons que la manœuvre a commencé.
- La troisième modification est l'adaptation du time-gap entre les agents à la situation. Le time-gap de deux secondes peut être ignoré ou réduit dans certains cas :
 - tout d'abord, le time-gap n'est appliqué qu'aux agents prioritaires, *i.e.* aux agents qui se trouvent sur une route prioritaire. Pour savoir si une route est prioritaire, nous tenons compte de la carte (qui pour rappel contient les informations relatives au régime de priorité à droite, aux régimes de priorité temporaires indiqués par des panneaux et aux marquages aux sol). Ainsi, un agent non prioritaire ne sera considéré qu'en cas de danger imminent lors des croisements ;
 - si un agent qui est situé derrière l'ego-véhicule ne respecte pas le time-gap fixé par les règles de la route, cet agent est dangereux et il faudrait que l'ego-véhicule se déporte pour le laisser passer. Si l'ego-véhicule peut se déporter sur la droite et se replier, il le fait (voir 3.2.3.2). En revanche, si le repli est impossible, se déporter sur la gauche pour laisser passer

un agent sur sa droite est un cas d'urgence. Il est préférable alors de garder ses distances avec l'agent dangereux, quitte à réduire le time-gap avec l'ensemble des autres agents. La situation de compromis la moins dangereuse semble être de laisser l'agent dangereux imposer sa valeur de time-gap. Pour éviter d'être responsable d'un accident en réduisant excessivement le time-gap, il doit a minima correspondre au temps de réaction maximal de l'approche. Nous choisissons un temps de réaction d'une seconde ;

- si l'ego-véhicule intersecte un volume d'occupation d'obstacle, l'approche ne peut pas trouver de solution. Ceci survient quand une collision semble inévitable sur la base des prévisions faites. Nous pouvons donc prévoir la collision si le time-gap est trop réduit ou si la position de l'ego-véhicule se retrouve dans un volume d'occupation. Nous pourrions alors utiliser d'autres approches pour calculer une manœuvre d'urgence, mais cette partie ne sera pas traitée.

3.3 Planification de vitesse avec prise en compte des zones occultées : méthode SGSPA

Pour rappel, dans ce qui suit, une carte de l'environnement de l'ego-véhicule comprenant toutes les informations sur la topologie de la route est supposée connue. Un chemin de référence \mathcal{P}_{ref} est calculé hors ligne à partir de la position initiale $X_{ego}(t = 0)$ et de la destination globale X_{goal} de l'ego-véhicule.

De plus, nous supposons :

- que l'ego-véhicule navigant sur le chemin ne peut entrer en collision avec les obstacles statiques, c'est-à-dire que le chemin de référence \mathcal{P}_{ref} est libre ;
- que les positions et les vitesses des autres véhicules sont connues pendant une courte période de temps dès qu'ils sont visibles depuis l'ego-véhicule ;
- que les autres véhicules proches d'une intersection peuvent indifféremment aller tout droit ou tourner à gauche ou à droite, lorsque cela est possible.

Notre objectif ici est d'adapter la vitesse de l'ego-véhicule afin d'obtenir une trajectoire sûre et confortable lorsqu'il suit son chemin en présence d'obstacles dynamiques tout en considérant les potentiels obstacles mobiles présents dans les zones occultées. La méthode décrite dans cette section, appelée SGSPA (pour Safe Geometric Speed Planning Approach) a fait l'objet d'une publication : [Poncelet *et al.*, 2020].

L'approche SGSPA sera présentée en premier lieu dans le cas où l'environnement de l'ego-véhicule ne contient pas de zones carrossables occultées, puis une extension tenant compte des véhicules potentiellement cachés par des obstacles fixes sera proposée dans la section 3.3.2.

3.3.1 Planification de vitesse dans l'espace chemin-temps

Supposons ici que tous les véhicules présents dans le voisinage de l'ego-véhicule sont visibles.

Deux processus sont exécutés en parallèle, avec différents pas de temps et différentes fréquences d'images :

1. pour chaque pas de temps τ_i , notre algorithme calcule le mouvement longitudinal $s(t)$ de l'ego-véhicule, en considérant les véhicules en interaction, c'est-à-dire la liste N_{dyn} véhicules environnants qui coupent la partie de la trajectoire de l'ego-véhicule limitée à $[s(\tau_i), s_h]$ pendant la période $\Delta T = [\tau_i, \tau_i + t_h]$, où t_h est un horizon temporel et s_h l'horizon de visibilité;
2. pour chaque pas de temps t_o , le mouvement longitudinal est simulé par un modèle de voiture décrit dans la section 3.3.1.5. L'intervalle de temps entre deux pas de temps consécutifs t_o et t_{o+1} est plus petit que celui de deux pas de temps consécutifs τ_i et τ_{i+1} , comme l'illustre la figure 3.22.

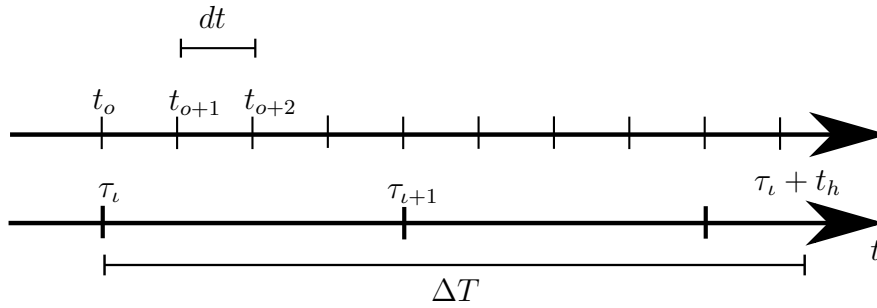


FIGURE 3.22 – Les différents pas de temps utilisés dans notre approche.

Un mouvement longitudinal est planifié à chaque pas de temps τ_i en considérant la période ΔT et la commande de l'ego-véhicule est calculée tous les t_o .

Suivons [Kant et Zucker, 1986] et calculons un mouvement longitudinal $s(t)$ pour l'ego-véhicule pendant la période $\Delta T = [\tau_i, \tau_i + t_h]$ le long du chemin \mathcal{P}_{ref} dans le domaine TP (s, t) . Le mouvement longitudinal $s(t)$ doit être une fonction monotone croissante de t avec des contraintes sur la pente et sur la courbure de $s(t)$ déduites des contraintes sur la vitesse $v(t)$ et l'accélération $a(t)$ de l'ego-véhicule. De plus, pour que le mouvement soit sûr, $s(t)$ peut être déduit du graphe de visibilité construit dans TP comme dans [Kant et Zucker, 1986], en considérant l'ensemble des N_{dyn} véhicules qui peuvent croiser \mathcal{P}_{ref} pendant ΔT .

Le calcul de $s(t)$, $v(t)$ et $a(t)$, $t \in \Delta T$ est effectué en trois étapes :

1. une zone polygonale, appelée TP obstacle, est insérée dans TP pour chaque véhicule (section 3.3.1.1) ;

2. une ligne polygonale $\tilde{s}(t)$, candidate pour $s(t)$, est construite dans TP (section 3.3.1.2);
3. $\tilde{s}(t)$ est lissée afin de respecter la contrainte sur l'accélération longitudinale (section 3.3.1.3).

Les étapes 2 et 3 sont illustrées dans la figure 3.24.

3.3.1.1 TP obstacles

Nous supposons que les N_{dyn} véhicules ont une vitesse constante pendant la période ΔT . Pour chacun d'eux, un TP obstacle est construit (défini de manière générale dans la section 3.1.3.1) : il correspond à la zone polygonale occupée dans l'espace TP par le véhicule pendant la période ΔT . Pour assurer la sécurité, le confort et prendre en compte les éventuelles variations de vitesse pendant la période ΔT , la zone occupée dans l'espace TP par un véhicule est élargie suivant la dimension temporelle d'un écart d'une valeur appelée time-gap (t_{gap}). Les TP obstacles sont aussi élargis suivant la dimension du chemin d'une marge de sécurité que nous définirons. Afin de prendre en compte les incertitudes liées aux manœuvres, nous considérons que l'ego-véhicule ignore la manœuvre choisie par un autre véhicule lorsqu'il s'engage dans un carrefour routier avant d'être totalement engagé sur une voie. Ainsi, pour le scénario de la figure 3.23 dans lequel l'ego-véhicule va traverser une intersection et où un autre véhicule arrive par la droite, nous considérons les chemins a , b et c comme étant potentiellement empruntés par l'autre véhicule et la région qui peut être occupée par le véhicule durant l'intervalle de temps ΔT est représentée par l'union des TP obstacles correspondant aux manœuvres a , b et c .

3.3.1.2 Construction d'un chemin candidat

Une fois les TP obstacles construits, un chemin candidat $\tilde{s}(t)$ pour $s(t)$ est calculé dans l'espace TP. Comme mentionné précédemment, c'est un chemin composé de segments de droite ayant une pente positive délimitée par la valeur de vitesse maximale v_{max} , joignant le point $(0, 0)$ et un point appartenant à la ligne verticale correspondant au temps $\tau_l + t_h$ ou appartenant à la ligne horizontale correspondant à s_h . Ce chemin relie un sous-ensemble de sommets des TP obstacles et il est construit par une méthode basée sur le graphe de visibilité comme dans [Kant et Zucker, 1986].

3.3.1.3 Contrainte d'accélération

Le chemin candidat $\tilde{s}(t)$ construit précédemment respecte les limites de vitesse mais il doit être lissé car il peut ne pas satisfaire les bornes d'accélération. Ceci

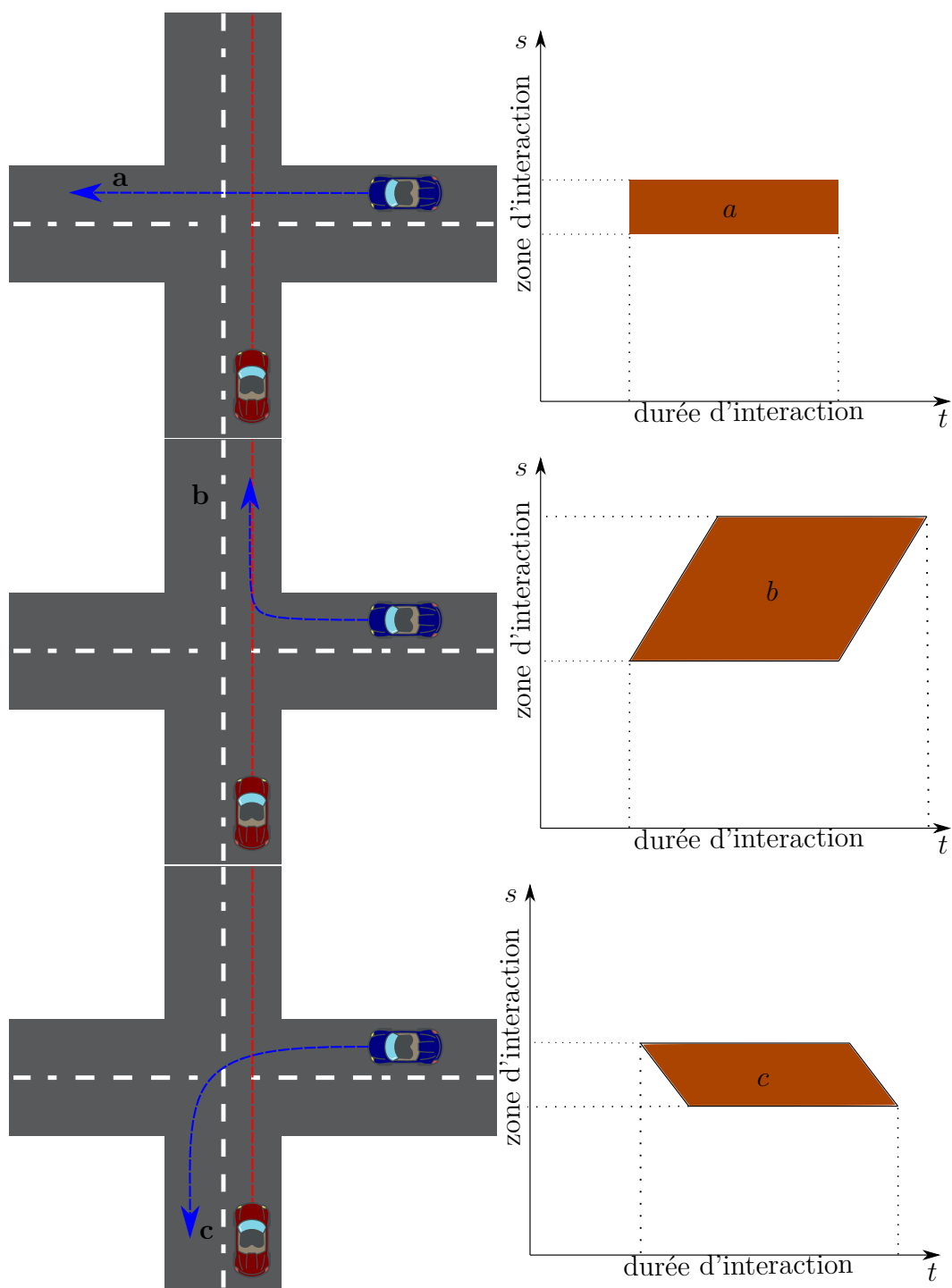


FIGURE 3.23 – Schéma des TP obstacles.
Les trois manœuvres possibles pour le véhicule bleu venant de la droite et leurs TP obstacles correspondants dans l'espace TP.

est effectué en lissant la vitesse $\tilde{v}(t)$ avec une saturation sur l'accélération $\tilde{a}(t)$ du chemin candidat $\tilde{s}(t)$, avec $\tilde{v}(\tau_l) = v_l$:

$$\tilde{a}(\tau_l) = \frac{\tilde{s}(\tau_l + dt) - \tilde{s}(\tau_l)}{dt^2} - \frac{v_l}{dt} \quad (3.64)$$

et, quand $\tau_l < t < \tau_l + t_h$:

$$\tilde{v}(t) = \frac{\tilde{s}(t + dt) - \tilde{s}(t)}{dt}, \quad (3.65)$$

$$\tilde{a}(t) = \frac{\tilde{v}(t + dt) - \tilde{v}(t)}{dt} \quad (3.66)$$

Le chemin lissé $s(t)$ est alors défini par :

$$s(\tau_l) = s_0, \quad (3.67)$$

$$v(\tau_l) = v_l \quad (3.68)$$

et, quand $\tau_l \leq t < \tau_l + t_h$:

$$a(t) = \min(\max(\tilde{a}(t), a_{Min}), a_{Max}), \quad (3.69)$$

$$v(t) = \min(\max(\tilde{v}(t) + a(t) \times dt, 0), v_{Max}), \quad (3.70)$$

$$s(t + dt) = s(t) + dt \times v(t) \quad (3.71)$$

Une fois $s(t)$ obtenu, un calcul de collision est réalisé entre le $s(t)$ et les TP obstacles, comme représenté dans la figure 3.24.

S'il y a collision avec un TP obstacle, un nouveau candidat $\tilde{s}(t)$ est construit et un nouveau chemin lissé $s(t)$ est calculé : le nouveau chemin candidat $\tilde{s}(t)$ correspond à une ligne polygonale passant à droite du polygone en collision (cf. figure 3.24).

3.3.1.4 État sûr avant une intersection

Avant de valider $\tilde{s}(t)$ comme une planification sûre, nous nous assurons que la planification ne se termine pas dans une intersection. En effet, dans ce cas, il sera difficile de lever les incertitudes de sécurité lors du franchissement de l'intersection. Si le domaine TP comprend une zone d'intersection, nous décidons de laisser l'ego-véhicule rouler aussi vite que possible jusqu'à la jonction tout en préservant une distance de sécurité pour lui permettre de freiner avant l'intersection. Afin de modéliser ce comportement dans le domaine TP, nous réduisons la limite de visibilité de l'ego-véhicule en déplaçant l'horizon de visibilité s_h juste avant l'intersection.

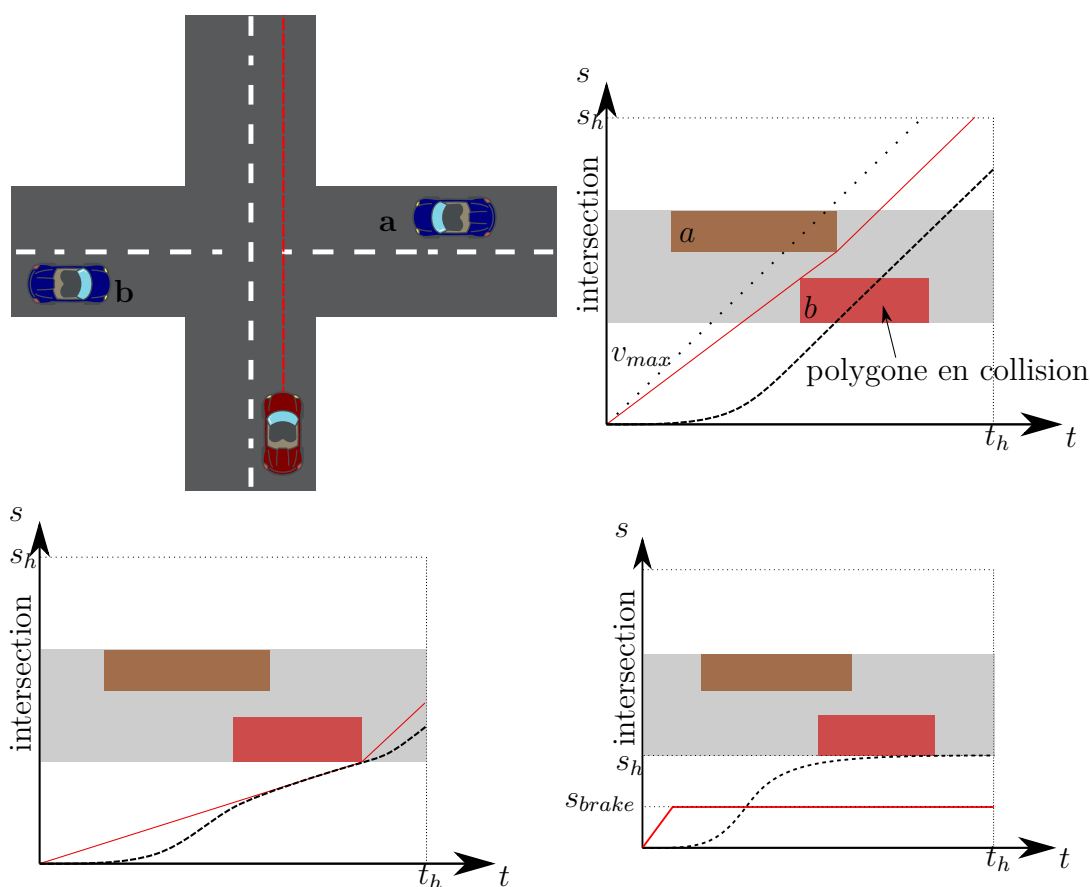


FIGURE 3.24 – Calcul de la courbe de la trajectoire dans l'espace TP.

En haut, à gauche, une vue d'ensemble de la scène. En bas, à gauche, la trajectoire candidate \tilde{s} (ligne rouge) obtenue par la méthode par graphe de visibilité comme dans [Kant et Zucker, 1986]. La version lissée (ligne pointillée noire) de \tilde{s} intersecte un TP obstacle (en rouge). À $t = 0$ l'ego-véhicule est stationnaire et sa vitesse est égale à 0. En haut, à droite, un nouveau chemin candidat \tilde{s} est construit, qui passe après le polygone en collision. En bas, à droite : l'horizon visuel (s_h) est virtuellement réduit avant la zone d'intersection (en gris). Un nouveau chemin candidat \tilde{s} est construit en considérant un état sûr avant l'intersection grâce à la contrainte $\tilde{s} < s_{brake}$. La dernière version lissée s est sûre. Nous remarquons qu'une solution peut toujours être trouvée sous deux conditions : La situation initiale doit être sûre, *i.e* l'ego-véhicule peut freiner sans entrer dans une intersection, les obstacles ne peuvent pas heurter l'ego-véhicule lorsqu'il est arrêté.

De plus, pour obliger le véhicule à s'arrêter lorsqu'il atteint la limite de visibilité, nous prévoyons un stop dans le graphe à la distance :

$$s_{brake} = s_h - d_{brake} \quad (3.72)$$

où d_{brake} est la distance requise pour freiner lorsque la vitesse de l'ego-véhicule est égale à la vitesse maximale v_{max} : si $\tilde{s}(t) > s_{brake}$ alors $\tilde{s}(t) = s_{brake}$ comme dans la figure 3.24.

3.3.1.5 Modélisation de la voiture

En parallèle des calculs précédemment décrits, à chaque pas de temps t_o , un contrôleur longitudinal adapte la commande de l'ego-véhicule pour modifier son accélération. Il utilise comme entrée $\hat{s}(t_o)$, $\hat{v}(t_o)$, les valeurs mesurées de $s(t)$ et $v(t)$ à $t = t_o$ et $s(t_o)$, $v(t_o)$. Puis, il calcule $\hat{a}(t_{o+1})$, $\hat{v}(t_{o+1})$ et $\hat{s}(t_{o+1})$ pour en déduire une commande pour l'ego-véhicule. La vitesse initiale considérée au temps τ_{i+1} pour calculer le mouvement longitudinal pour la période suivante est $v_{i+1} = \hat{v}(\tau_{i+1})$, comme précisé dans la section 3.3.1.3 .

3.3.2 Prise en compte des zones occultées et des limites de perception

Étendons maintenant notre approche pour prendre en compte l'existence de portions de voies occultées sur le chemin de l'ego-véhicule. Pour nous inspirer du comportement d'un conducteur humain dans cette situation, nous introduisons des véhicules virtuels et une ligne d'horizon virtuelle (section 3.3.2.1) et adaptons le mouvement longitudinal $s(t)$ pour pouvoir réagir en toute sécurité sans être bloqué (section 3.3.2.2).

3.3.2.1 Véhicules virtuels et horizon virtuel

Pour représenter tous les risques des zones occultées appartenant aux voies rencontrant la trajectoire de l'ego-véhicule, nous choisissons d'introduire des obstacles virtuels sur les frontières de visibilité, comme dans [Damerow *et al.*, 2017, Hoermann *et al.*, 2017, Orzechowski *et al.*, 2018, Naumann *et al.*, 2019, Tas et Stiller, 2018, Karanam *et al.*, 2022, Sánchez *et al.*, 2022]. Afin de prévoir le pire cas de figure, nous supposons que l'obstacle virtuel est d'une longueur infinie et qu'il roule à la vitesse maximale possible, sauf lorsqu'il existe un véhicule se déplaçant sur la même voie devant la frontière de visibilité. Dans ce cas, nous supposons que l'obstacle virtuel a la même vitesse que ce véhicule. De plus, lorsque la limite de visibilité est proche d'une intersection, nous supposons que le véhicule peut suivre n'importe quelle trajectoire, en tournant à gauche, à droite ou en

allant tout droit lorsque cela est possible. Nous créons donc les obstacles TP de la façon suivante :

au pas de temps τ_ι , si une frontière d'une zone occultée appartient à une voie croisant la trajectoire \mathcal{T}_{ego} de l'ego-véhicule durant ΔT , et si cette frontière est très proche de la jonction routière, un TP obstacle est inséré dans l'espace TP pour représenter la région occupée par un obstacle virtuel positionné à la limite de visibilité et roulant à la vitesse maximale pendant ΔT ou à la même vitesse que le véhicule le précédant lorsqu'un tel véhicule existe. Comme un obstacle virtuel est supposé être de longueur infinie, le polygone correspondant est délimité par la limite droite de l'espace TP, c'est-à-dire la ligne verticale $t = \tau_\iota + \Delta T$, comme dans la figure 3.25. Les zones situées en dehors de la portée du capteur peuvent également contenir des obstacles cachés. Ainsi, si nous voulons assurer un comportement sûr, l'ego-véhicule doit s'arrêter lorsqu'il atteint la valeur d'horizon de visibilité s_h dans sa trajectoire. La ligne horizontale $s = s_h$ sera donc considérée comme un obstacle lors du calcul de la trajectoire candidate.

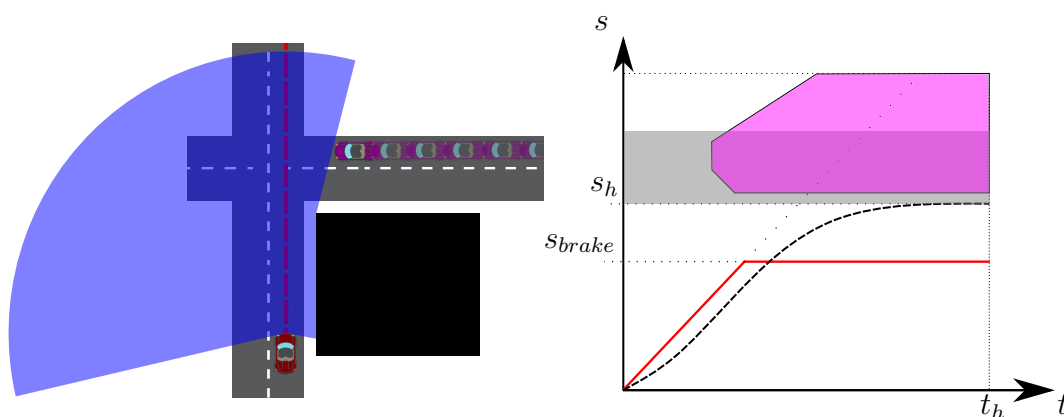


FIGURE 3.25 – Scénario avec une zone occultée.

À gauche : l'ego-véhicule suit une trajectoire (ligne pointillée rouge) et un obstacle noir cache une partie de la route (la zone visible correspond à la zone bleue transparente). Un véhicule virtuel a donc été créé (dessiné en violet). À droite : le TP obstacle correspondant (notez que sa forme correspond à l'union des TP obstacles relatifs aux trois manœuvres possibles a, b et c de la figure 3.23), le chemin candidat $\tilde{s}(t)$ et le chemin sûr $s(t)$ pour l'ego-véhicule dans l'espace TP.

3.3.2.2 Calcul de la planification de vitesse

Une fois que les TP obstacles correspondant aux véhicules virtuels et aux véhicules visibles et que la ligne d'horizon de visibilité ont été insérés dans l'espace TP, nous devons calculer un chemin candidat $\tilde{s}(t)$ pour l'ego-véhicule. Nous avons deux cas à considérer :

- lorsque la visibilité n'est pas suffisante, notre solution doit être sûre mais pas trop prudente. Les TP obstacles correspondants seront étendus. L'ego-véhicule n'aura pas suffisamment de temps pour passer. L'horizon de visibilité s_h est alors déplacé vers la plus petite valeur de s du premier TP obstacle associé à un véhicule virtuel pour permettre à l'ego-véhicule de freiner si un autre véhicule vient de la zone occultée (comme dans le cas (a) de la figure 3.26) ;
- si les limites des zones occultées sont suffisamment éloignées de l'intersection avec la trajectoire de l'ego-véhicule, les TP obstacles correspondants seront petits. Dans ce cas, l'ego-véhicule aura suffisamment de visibilité pour pouvoir traverser le carrefour en toute sécurité et une ligne polygonale $\tilde{s}(t)$ construite comme précédemment passera par la partie gauche de la zone correspondant au carrefour dans l'espace TP (comme dans le cas (b) de la figure 3.26).

Ensuite, une version lissée $s(t)$ de $\tilde{s}(t)$ est calculée comme dans la section 3.3.1.3.

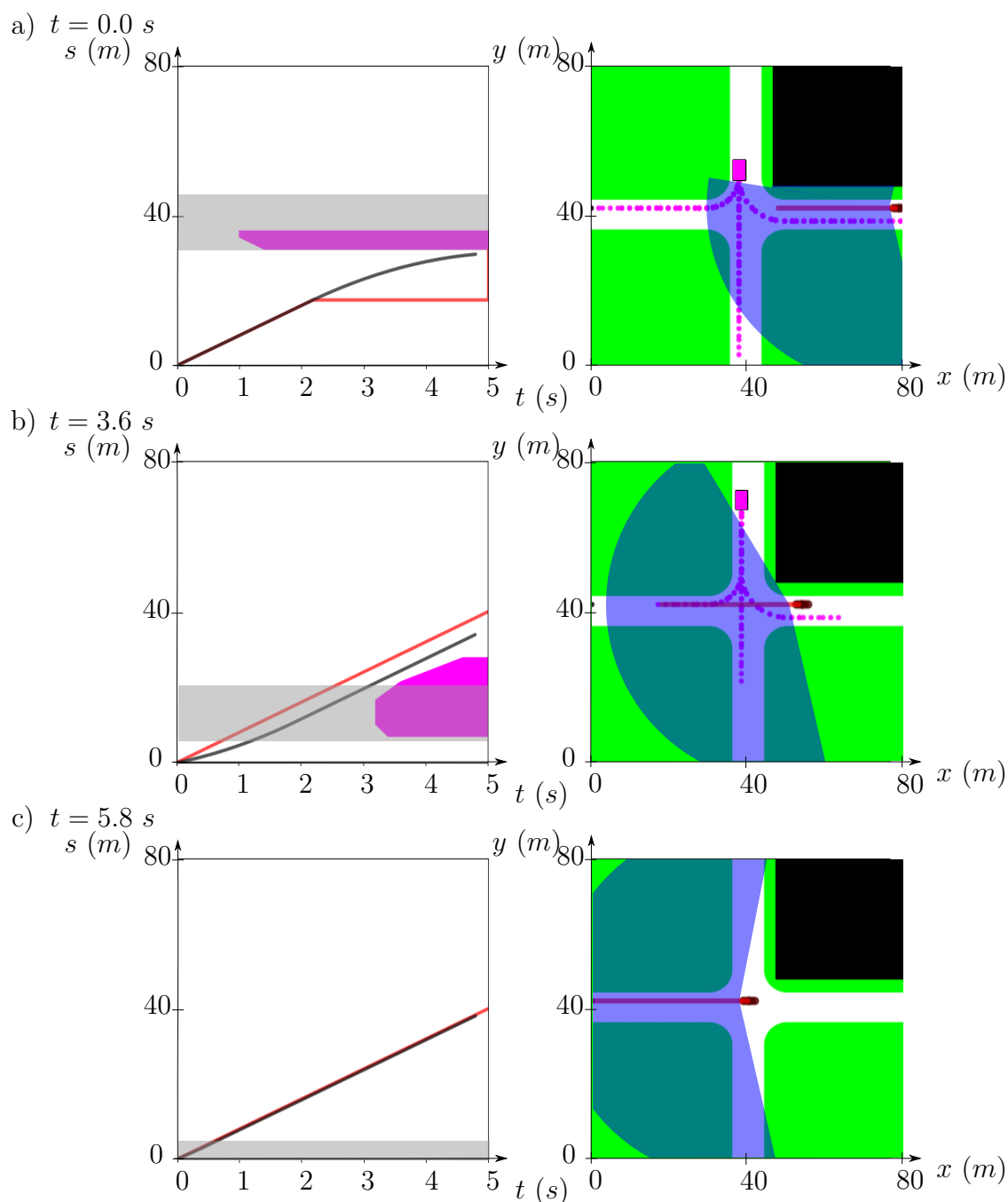


FIGURE 3.26 – Exemple où la zone occultée est vide.

L'égo-véhicule circule de droite à gauche et rencontre une voie à sens unique (sens de circulation vers le bas). Différents pas de temps de la méthode ($t = 0$, $t = 3.6s$, $t = 5.8s$). Véhicules virtuels : les trajectoires possibles et les TP obstacles sont dessinés en rose. Ego-véhicule : $\tilde{s}(t)$ est dessiné en rouge et $s(t)$ en noir.

3.4 Planification de trajectoire en milieu urbain : méthode GPAD

Dans les deux sections précédentes, nous avons présenté deux méthodes géométriques de planification de trajectoire pour un ego-véhicule évoluant dans un milieu urbain comportant d'autres véhicules en interaction avec lui : une planification de chemin connaissant le profil de vitesse de l'ego-véhicule et une planification de vitesse quand l'ego-véhicule circule sur un chemin de référence.

Notre objectif ici est de calculer des trajectoires sûres en milieu urbain ouvert ou semi-contraint. L'ego-véhicule doit se déplacer de sa position de départ jusqu'à un objectif défini dans un environnement comportant des interactions typiques d'un environnement urbain telles que d'autres usagers de la route, des situations de croisement avec ou sans feu de signalisation et des obstacles fixes comme des véhicules arrêtés sur la chaussée.

De nombreuses approches de planification de trajectoire d'un ego-véhicule (cf. [González *et al.*, 2016], [Paden *et al.*, 2016] et [Claussmann *et al.*, 2020]) présupposent une topologie du réseau de routes connue et planifient le mouvement par adaptation de vitesse quand le véhicule circule dans une voie, ou génèrent des manœuvres de changement de voie. Il est cependant nécessaire dans certains cas de figure de faire naviguer l'ego-véhicule dans un environnement où les voies de circulation sont parfois mal définies ou partiellement, voire complètement, inconnues. L'objectif de notre approche est donc de pouvoir calculer également une planification de trajectoire pour l'ego-véhicule dans ces cas-là, plus précisément :

- l'ego-véhicule qui suit une voie ou un chemin donné doit pouvoir adapter sa vitesse lorsqu'il croise d'autres usagers de la route pour obtenir une trajectoire sûre, c'est-à-dire sans collision, tout en adoptant un comportement acceptable vis-à-vis des règles de circulation et des autres usagers de la route ;
- l'approche doit pouvoir détecter le fait que le suivi de voie mène à une situation de blocage ou à une collision pour l'ego-véhicule. Dans ce cas, une trajectoire alternative doit être proposée. Cette trajectoire doit tenir compte du mouvement des autres usagers, mais aussi de l'environnement proche de l'ego-véhicule et, en particulier, de la présence ou non de voie définie, de la chaussée, etc.

Une solution est de combiner les deux méthodes de planification de chemin et de vitesse présentées dans les sections précédentes, comme nous allons le voir dans la suite de ce chapitre. L'approche présentée ici sera nommée approche GPAD pour Geometric Planning with Alternating Decomposition.

3.4.1 Approche GPAD

3.4.1.1 Hypothèses

Nous considérons que l’ego-véhicule évolue dans des environnements urbains ou péri-urbains où sont connus :

- des portions de voies représentées par des lignes polygonales ;
- la chaussée représentée par un ensemble de polygones ;
- des feux de circulation.

La cinématique de l’ego-véhicule est supposée régie par le modèle bicyclette détaillé dans le paragraphe (a) de la section 3.2.3.1.

Les autres véhicules présents dans cet environnement sont supposés totalement observables et partiellement prévisibles, c’est-à-dire que l’état (position et orientation du véhicule) et la dérivée de l’état (vitesse du véhicule) sont connus à chaque instant. Nous supposons de plus que les autres véhicules se déplacent à vitesse constante, qu’ils peuvent suivre n’importe quelle voie et qu’ils respectent les règles de conduite.

3.4.1.2 Éléments principaux

Avant de décrire la façon dont sont combinées les différentes méthodes de planification de vitesse ou de chemin dans la planification, listons les éléments utilisés par notre approche et mentionnés dans sa description schématique (cf. figure 3.27) :

- avant chaque planification, nous devons disposer d’un *chemin de référence* \mathcal{P}_{ref} pour l’ego-véhicule. Ce chemin est calculé en ne tenant compte que des obstacles statiques de la scène.
L’étape de calcul de ce chemin est appelé le *renouvellement du chemin* ;
- dans le cas où le chemin de référence \mathcal{P}_{ref} ne mènerait qu’à des situations de blocage ou de fort ralentissement de l’ego-véhicule, un nouveau chemin pour l’ego-véhicule \mathcal{P}_{ego} est calculé en tenant compte du mouvement des obstacles mobiles.
Les tests permettant de déclencher le calcul d’un nouveau chemin \mathcal{P}_{ego} pour l’ego-véhicule sont appelés *événement de vitesse et changement de voie nécessaire* ;
- les *données de l’environnement* correspondent aux informations (position, vitesse) relatives aux N_{dyn} véhicules en mouvement présents dans l’environnement immédiat de l’ego-véhicule à l’instant t considéré ;
- le module MMRIS-APF (Multi Modal Reachable Interaction Sets - Artificial Potential Field) calcule le chemin \mathcal{P}_{ego} et donne en sortie la vitesse $v_{ego}(t)$ sélectionnée dans le calcul. Le chemin peut être calculé de différentes manières :

- il peut être un des chemins candidats utilisé par MMRIS si celui-ci est libre de bout en bout,
- à défaut, il peut être calculé de sorte à éviter des RIS par une méthode basée sur les APF ;
- la planification de vitesse, *i.e.* le calcul du profil de vitesse sur le chemin \mathcal{P}_{ego} ou \mathcal{P}_{ref} , est effectuée par la méthode SGSPA décrite dans la section 3.3.

Les chemins et vitesses calculés pour l’ego-véhicule à chaque instant tiennent compte des obstacles fixes et mobiles pour fournir une trajectoire sans collision.

3.4.1.3 Obstacles virtuels et obstacles conditionnels

Cette représentation permet d’ajouter des obstacles virtuels pour représenter divers risques. Ici, nous utiliserons des obstacles virtuels pour les feux tricolores. Il est aussi possible d’utiliser des obstacles conditionnels pour représenter les règles de la route comme dans [Ajanovic *et al.*, 2018].

3.4.2 Architecture

3.4.2.1 Planification alternative

Pour permettre à la fois une adaptation de la vitesse sur un chemin connu et des changements de trajectoire, utilisons la décomposition chemin-vitesse en contraignant alternativement le chemin puis la vitesse ou inversement.

Par exemple, si nous avons un chemin, nous utiliserons une méthode de planification de vitesse le long de ce chemin (cf. figure 3.27). Dans le cas contraire, nous utiliserons une méthode de calcul de chemin pour trouver une alternative.

Décrivons maintenant les deux étapes principales structurant la planification alternative.

1. Au départ, nous planifions un chemin pour l’ego-véhicule via la méthode MMRIS-APF qui étend les RIS introduits dans [de Beaucorps *et al.*, 2018] et construit un chemin à l’aide des APF (MMRIS est détaillée dans la section 3.2.3.3). Cette méthode est robuste vis-à-vis de l’absence de voies de circulation définies car elle ne nécessite pas de pré-requis.

Connaissant les zones d’interaction potentielle calculées par MMRIS, un chemin candidat libre de bout en bout est sélectionné (section 3.2.3.2). Dans le cas contraire, une méthode de planification par APF basée sur la construction d’histogramme d’un champ de vecteurs [Koren et Borenstein, 1991] calcule une trajectoire pour l’ego-véhicule. La méthode APF est présentée dans la section 3.4.3. Enfin, la solution du mode le plus rapide est sélectionnée.

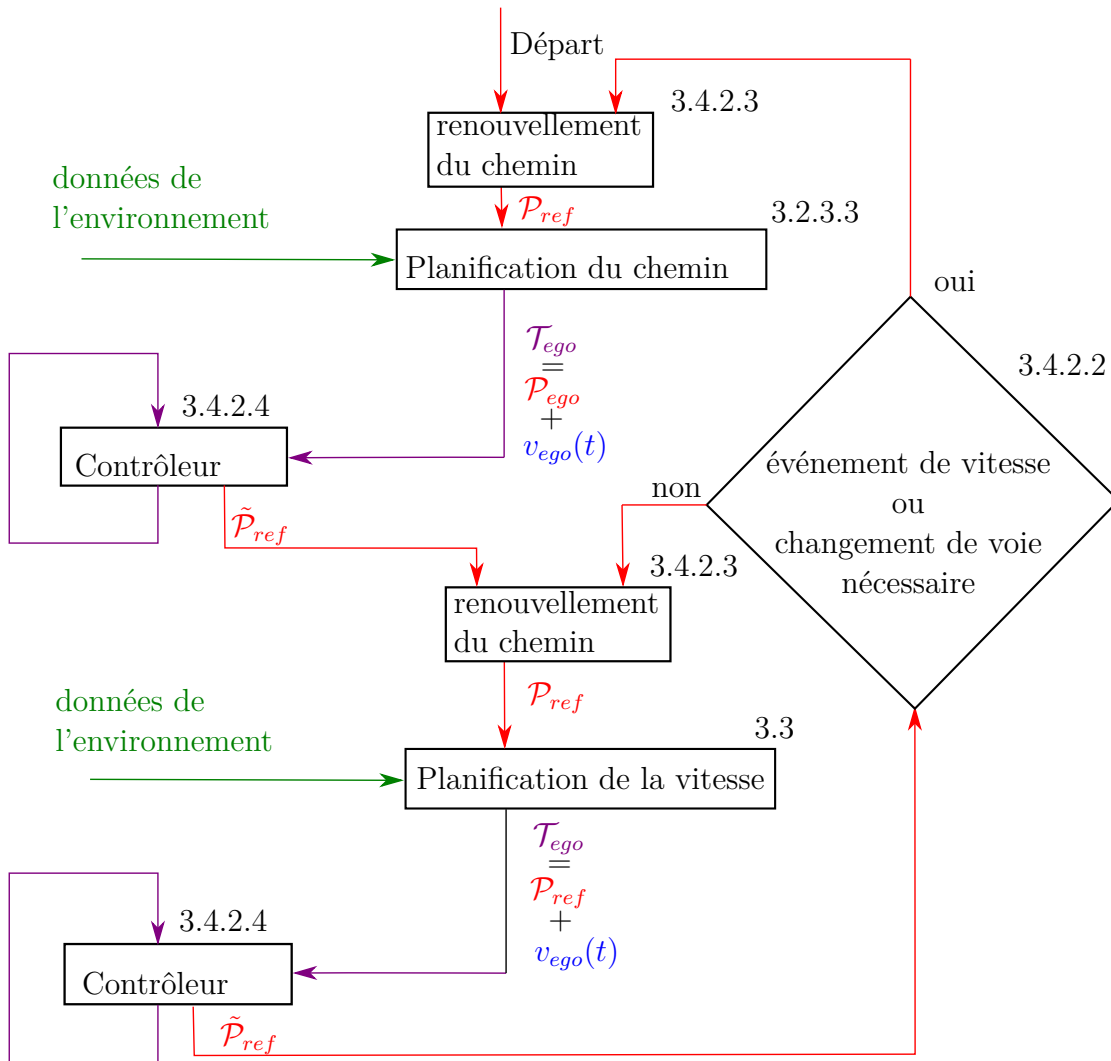


FIGURE 3.27 – Présentation schématique de l’architecture de la planification alternative.

Nous obtenons ainsi une trajectoire sûre \mathcal{T}_{ego} sous la forme d'un profil de vitesse $v_{ego}(t)$ et d'un chemin sûr suivant ce profil \mathcal{P}_{ego} .

Nous calculons donc un premier contrôle à partir de cette trajectoire et de la position de l'ego-véhicule (voir la section 3.4.2.4). Cette trajectoire peut être valide pour effectuer une mise à jour du contrôle durant un certain laps de temps en utilisant la position de l'ego-véhicule. Au bout d'un moment, l'utilisation de cette trajectoire peut devenir risquée. En effet, l'environnement a évolué et nos prévisions deviennent de moins en moins bonnes. Il faut alors calculer une nouvelle planification. Comme pour la méthode SGSPA vue dans la section 3.3, la planification de la méthode GPAD sera effectuée de manière régulière (la fréquence de planification étant inférieure à la fréquence du contrôleur (cf. figure 3.22)).

2. Avant d'effectuer la planification suivante, le chemin de référence \mathcal{P}_{ref} est renouvelé (section 3.4.2.3). Ce nouveau chemin de référence est utilisé pour planifier un profil de vitesse grâce à la méthode SGSPA (Safe Geometric Speed Planning Approach) [Poncelet *et al.*, 2020] en fonction des données de l'environnement de l'ego-véhicule.

Nous obtenons de la même manière une trajectoire sûre \mathcal{T}_{ego} sous forme d'un profil de vitesse sûr $v_{ego}(t)$ le long du chemin de référence \mathcal{P}_{ref} .

Une nouvelle série de contrôles est calculée à partir de cette trajectoire et de la position de l'ego-véhicule.

De même, nous renouvelons le chemin à partir de la trajectoire obtenue avant d'effectuer la planification suivante.

3. Après une planification de la vitesse, nous utilisons d'abord un test de décision "événement vitesse" ou "changement de voie" détaillé dans la section 3.4.2.2 pour décider s'il faut :
 - recalculer un nouveau chemin avec les MMRIS-APF (réponse "oui" et retour en 1)) ou
 - recalculer un profil de vitesse avec la méthode SGSPA en conservant le chemin (réponse "non" et retour en 2)).

3.4.2.2 Décision du calcul d'une manœuvre

Nous verrons dans la section 3.2.3.5 concernant les règles d'usage de la route que certaines situations exigent le calcul d'une manœuvre. Dans ce cas, la méthode MMRIS-APF est utilisée. Dans d'autres situations, des critères de vitesse permettent de décider s'il est utile d'envisager une manœuvre : si la vitesse planifiée est inférieure à la vitesse maximale autorisée durant une longue période ou si une forte perte de vitesse est prévue, une manœuvre pourrait permettre à l'ego-véhicule d'atteindre plus rapidement l'objectif désiré.

Ces critères peuvent s'écrire sous la forme :

$$c0 = v(0) - \min(v(t)) > \frac{v_{max}}{f0} \quad (3.73)$$

$$c1 = v_{max} - \max(v(t)) > \frac{v_{max}}{f1} \quad (3.74)$$

où $v(t)$ est la planification de la vitesse et v_{max} est la vitesse maximale autorisée, $f0$ et $f1$ sont des paramètres de réglage.

Si l'un des deux critères n'est pas respecté, un nouveau chemin est alors calculé.

3.4.2.3 Chemin de référence

Le chemin de référence \mathcal{P}_{ref} peut être la voie sur laquelle est situé l'ego-véhicule. Il est alors construit à partir d'un échantillonnage de points de la voie courante de l'ego-véhicule récupérés grâce à la carte de l'environnement. La longueur du chemin de référence est limitée par l'horizon temporel t_h et la vitesse maximale autorisée v_{max} qui donnent la distance maximale atteignable durant l'horizon temporel $s_{max} = v_{max}t_h$ ainsi que l'horizon visuel s_h . Pour éviter les calculs inutiles, nous voulons disposer d'un chemin de référence \mathcal{P}_{ref} de longueur fixe $s_{ref,max} = \min(s_{max}, s_h)$. Il existe des cas où l'ego-véhicule n'a pas de voie courante, par exemple quand la carte n'est pas complète ou quand l'ego-véhicule est trop éloigné d'une voie définie sur la carte. Dans ce cas, une méthode basée sur les APF est utilisée pour calculer le chemin de référence à partir des obstacles statiques (section 3.4.3).

Une partie du chemin de référence est en général réutilisée d'une planification à l'autre. Après une planification de la vitesse, nous utilisons l'ancien chemin de référence \mathcal{P}_{ref} . Après une planification de chemin, nous utilisons le chemin calculé $\mathcal{P}_{ref} = \mathcal{P}_{ego}$. Toutefois, après plusieurs calculs de contrôle et le passage d'un laps de temps, la position de l'ego-véhicule sur le chemin de référence a changé. Pour éviter les calculs inutiles, nous remplaçons \mathcal{P}_{ref} par $\tilde{\mathcal{P}}_{ref}$ où la partie de \mathcal{P}_{ref} située à l'arrière de l'ego-véhicule a été supprimée. Ce chemin est plus court, il faut donc le renouveler pour que sa longueur soit de nouveau égale à $s_{ref,max}$. Ceci est effectué en utilisant la voie sur laquelle est situé $\tilde{\mathcal{P}}_{ref}$ si elle existe, ou en calculant un nouveau chemin par la méthode décrite en section 3.4.3 dans le cas contraire.

3.4.2.4 Calcul du contrôle

Pour valider les approches et garantir la sécurité lors de la manœuvre, il faut s'assurer que l'ego-véhicule restera sur la trajectoire prévue.

Nous utilisons pour le suivi de chemin un contrôle de Stanley [Snider, 2009] avec un correcteur PID.

Il est donc possible de garantir que l'ego-véhicule ne s'éloignera pas trop de sa trajectoire.

3.4.3 Planification de chemin par APF

Pour certains calculs de chemins, nous utilisons une méthode basée sur une approche par APF [Koren et Borenstein, 1991].

Nous considérons quatre types d'éléments pour construire le champ de potentiels :

- le but local situé à l'horizon visuel ;
- les obstacles ;
- les voies ;
- le chemin précédemment calculé.

3.4.3.1 Grille de calcul

Dans le but de rendre le calcul de chemin par APF robuste aux changements de situations, nous utilisons les RS comme grille de calcul. Nous calculons une force associée à chaque point des RS en fonction de sa distance aux éléments cités plus haut. Par analogie avec l'électrostatique et pour faciliter le calcul des APF, nous allons calculer des charges associées aux différents éléments.

3.4.3.2 But local

Le but local participe à une force attractive dans le calcul de l'APF. Nous allons donc lui associer une charge positive par convention. La part de la charge du but en chaque point de coordonnées X est régie par l'équation suivante :

$$q_{but}(X) = |X_{but} - X| \quad (3.75)$$

où X_{but} sont les coordonnées du but.

3.4.3.3 Obstacles

Les obstacles participent à une force répulsive dans le calcul de l'APF. Nous allons donc lui associer une charge négative. Pour chaque point de coordonnées X , nous considérons la distance à l'obstacle le plus proche $d_{obstacle}(X)$ pour calculer la charge.

$$q_{obstacles}(X) = -d_{obstacle}(X) \quad (3.76)$$

3.4.3.4 Les voies et le chemin précédent

Les voies et le chemin précédent participent à une force attractive. Pour chaque point de coordonnées X , nous considérons la distance à la voie la plus proche $d_{voie}(X)$ ainsi que la distance au chemin précédent $d_{path}(X)$ pour calculer la charge.

$$q_{voie}(X) = d_{voie}(X) \quad (3.77)$$

$$q_{path}(X) = d_{path}(X) \quad (3.78)$$

3.4.3.5 Calcul de la charge totale

La somme des charges est pondérée comme suit :

$$q_{totale}(X) = k_1 * q_{but}(X) + k_2 * q_{obstacles}(X) + k_3 * q_{voie}(X) + k_4 * q_{path}(X) \quad (3.79)$$

où k_1 , k_2 et k_3 sont des paramètres de réglages de l'APF.

3.4.3.6 Calcul du chemin

Le chemin est obtenu par itération en partant d'une position p_0 . La somme des forces en un point de coordonnées p quelconque est donnée par l'équation suivante :

$$F(p) = \sum_{X \in RS_{p_0}(p)} \frac{X - p}{|X - p|^3 * q_{totale}^*(X)} \quad (3.80)$$

où $q_{totale}^*(X)$ est une version normalisée de $q_{totale}(X)$, $q_{totale}^*(X) = q_{totale}(X) - \min_{X \in RS}(q_{totale}(X)) + 1$ et où $RS(p)$ sont les RS accessibles depuis la position p . Pour ne pas recalculer les RS à chaque étape, nous remplaçons les $RS(p)$ par les $RS_{p_0}(p)$. Ils sont déduits des $RS(p_0)$ en supprimant les RS antérieures à la position p . Pour construire les $RS_{p_0}(p)$, les configurations de $RS(p_0)$ sont projetées sur le plan d'espace-temps $E(t = 0)$. On trouve la configuration la plus proche de p , que l'on note X_p . Dans les RS, chaque configuration est associée à un instant t . Les configurations de $RS(p_0)$ des instants antérieurs à X_p sont supprimées pour obtenir $RS_{p_0}(p)$.

La somme des forces nous donne la direction cible de l'ego-véhicule. En tenant compte de ses contraintes cinématiques de celui-ci, nous calculons la direction réellement adoptée pour cette itération. Nous obtenons alors la position p_1 et nous réitérons jusqu'à atteindre le but ou jusqu'à obtenir un chemin suffisamment long.

3.4.4 Conclusion et attentes

Avec un même type de représentation de l'occupation des obstacles et une méthode commune de calcul d'intersections, l'approche présentée peut calculer différents types de planification de mouvement en fonction du besoin. En effet, elle peut alternativement :

- permettre une planification de suivi de chemin avec un choix d'accélération restreint grâce à la méthode MMRIS ;
- et permettre une planification de la vitesse le long d'un chemin

La planification de chemin peut être restreinte à un choix de chemins candidats et, en dernier recours, calculer une manœuvre libre grâce aux RIS.

Nous pouvons aussi représenter les différentes règles d'usage et du code de la route.

Chapitre 4

Résultats et simulation

Table des matières

4.1	Scénarios et résultats de l’approche RIS	98
4.1.1	Implémentation de la méthode	98
4.1.1.1	Calcul des RS	98
4.1.1.2	Interpolation de l’occupation des obstacles	100
4.1.1.3	Calcul d’intersection dans le cas d’une interpolation <i>a priori</i>	100
4.1.2	Validation expérimentale	101
4.1.2.1	Présentation des environnements	101
4.1.2.2	Prise en compte des contraintes de mouvement pour les véhicules non-holonomes	103
4.1.2.3	Méthodes développées pour les tests d’évaluation	104
a)	Description des approches	105
b)	Optimisation des différentes approches	105
4.1.2.4	Influence de la prise en compte du comportement stochastique des obstacles dans l’approche RIS-path	107
4.1.2.5	Stratégie d’urgence pour l’ego-véhicule : comparai- son de RIS-path projection et RIS-path horizon	108
4.1.2.6	Résultat du RIS-path stochastique	109
a)	Mouvement holonome de l’ego-véhicule	109
b)	Mouvement non-holonome de l’ego-véhicule	112
4.2	Scénarios et résultats de l’approche SGSPA	114
4.2.1	Traversée d’un carrefour en X	114
4.2.1.1	Zone occultée vide	115
4.2.1.2	Un véhicule est sur la partie occultée de la route	116
4.2.2	Traversée d’un carrefour en T avec visibilité limitée	117
4.2.3	Traversée d’un carrefour en X avec dix voitures	118
4.2.4	Conclusion	119
4.3	Scénarios et résultats de l’approche GPAD	120
4.3.1	Présentation de l’environnement de tests	120
4.3.2	Présentation des scénarios et résultats	121

4.3.2.1	Premier scénario : l'ego-véhicule circule tout seul .	121
4.3.2.2	Deuxième scénario : un véhicule est arrêté sur la voie	123
4.3.2.3	Troisième scénario : deux véhicules sont arrêtés sur la voie	124
4.3.2.4	Quatrième scénario : un véhicule est arrêté sur la voie et un autre véhicule arrive en face	125
4.3.2.5	Cinquième scénario : un véhicule est arrêté sur la voie et un autre véhicule arrive trop rapidement en face	126

Dans le but de tester le bon fonctionnement des approches présentées dans le Chapitre 3, différents environnements simulés ont été utilisés. Les détails de l'implémentation de l'approche géométrique, des calculs d'intersections dans l'espace temps et de l'ensemble des approches testées sont présentés dans l'annexe 6.1.

Dans un premier temps, dans la section 4.1, nous allons présenter les scénarios et les résultats de l'approche RIS dans un contexte plutôt robotique. Ensuite, nous présenterons dans la section 4.2 les résultats de l'approche SGSPA dans un environnement urbain simulé avec CARLA. Enfin, nous discuterons dans la section 4.3 des résultats de l'approche GPAD dans un environnement urbain simulé ici aussi avec CARLA.

4.1 Scénarios et résultats de l'approche RIS

Pour évaluer les performances des RIS pour la planification de trajectoire, différentes méthodes ont été développées puis testées dans des environnements simulés. Avant de les présenter, nous détaillons dans la section 4.1.1 l'implémentation des RIS introduits dans la section 3.2.

Nous présenterons ensuite les évaluations expérimentales dans la section 4.1.2. Dans un premier temps, les scénarios sont composés d'environnements ouverts et non contraints, *i.e.* sans voie ni règle de circulation et les trajectoires calculées sont plutôt adaptées au mouvement d'un robot. Nous verrons que certains environnements peuvent conduire à l'élaboration de scénarios proches du milieu urbain, mais de manière sommaire. Ces scénarios et environnements ont pour objectif de valider expérimentalement les améliorations faites sur les RIS introduits dans [de Beaucorps *et al.*, 2018].

Une partie du travail présenté dans cette section a été effectué en collaboration avec Pierre de Beaucorps [de Beaucorps *et al.*, 2018, de Beaucorps, 2019].

4.1.1 Implémentation de la méthode

4.1.1.1 Calcul des RS

Les RS sont calculées pour un ensemble de valeurs constantes u_ζ respectant l'équation (3.43). Un exemple de surface est présenté en figure 4.3, ③.

En pratique, nous simplifions ce modèle pour pouvoir calculer les RS préalablement à leur utilisation. Nous calculons différents RS pour une configuration de référence de l'ego-véhicule $X_{ego,0}$ avec $(x_{ego}, y_{ego}, \theta_{ego}) = (0, 0, 0)$. Nous discrétisons les valeurs possibles de vitesse initiale de l'ego-véhicule $v_{ego,init}$ entre 0 et $v_{ego,max}$. Pour chaque valeur de $v_{ego,init}$, nous calculons les positions $x_{ego}(t_o)$ et $y_{ego}(t_o)$ pour toutes les valeurs de u_ζ et pour un temps discret. Nous transformons donc le système d'équations (3.42) :

$$\begin{cases} \theta_{ego}(t_{o+1}) &= \theta_{ego}(t_o) + u_\zeta dt \\ x_{ego}(t_{o+1}) &= x_{ego}(t_o) + v_{ego}(t) \sin(\theta_{ego}(t_{o+1})) dt \\ y_{ego}(t_{o+1}) &= y_{ego}(t_o) + v_{ego}(t) \cos(\theta_{ego}(t_{o+1})) dt \end{cases} \quad (4.1)$$

Pour n'importe quelle valeur de u_ζ non nulle, il existe un temps au bout duquel l'ego-véhicule retourne à son état initial (position et orientation). Ce comportement n'est pas désiré pour l'ego-véhicule car il empêche d'avoir une bijection entre les RS et l'espace (x, y) (cf. équation (3.29)). Nous décidons donc de supprimer l'ensemble des points tels que :

$$x_{ego}(t_{o+1}) * x_{ego}(t_o) < 0 \quad (4.2)$$

et ne calculons pas les positions futures des points ainsi supprimés. Chaque couche des RS est représentée par une ligne polygonale. Nous souhaitons interpoler la surface des RS entre ses couches temporelles. La surface est construite en reliant les lignes polygonales successives correspondant aux instants t_o, t_{o+1} .

Pour obtenir un maillage suffisamment précis, u est discrétisé en 65 valeurs. Ceci est raisonnable pour les lignes polygonales correspondant aux instants proches de l'horizon temporel, mais induit une discrétisation trop fine des lignes polygonales correspondant aux instants plus proches de l'instant initial. En effet, les lignes polygonales ont tendance à s'allonger et elles comportent de moins en moins de points au cours du temps. Pour conserver un écart raisonnable entre deux points successifs d'une ligne polygonale, nous supprimons des points des lignes de faible longueur. Pour conserver la forme des RS, ces points sont supprimés de manière régulière le long des lignes polygonales en fonction du ratio du nombre de points souhaité par unité de longueur que nous déterminons.

Pour construire la surface correspondant au véhicule de Dubins, il nous faut relier les lignes polygonales successives par des faces régulières. Or, le fait de supprimer des points des lignes polygonales nous empêche d'effectuer une mise en correspondance systématique en considérant le paramétrage u_ζ de chaque ligne polygonale. Nous savons cependant que les points sont organisés par ζ croissant.

Il existe différents cas de figure suivant les deux lignes que nous souhaitons relier :

- Les lignes polygonales peuvent être ouvertes ou fermées (si nous sommes avant ou après le premier demi-tour complet).
- Les deux lignes polygonales ont le même nombre de points ou non.

Dans le cas où les lignes considérées ont le même nombre de points, nous décidons de relier les points comme dans la figure 4.1. Si en revanche les couches n'ont pas le même nombre de points, nous allons procéder de la même manière en rajoutant régulièrement un triangle supplémentaire comme le montre la figure 4.2.

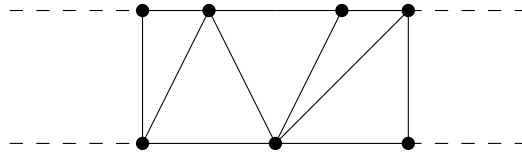


FIGURE 4.2 – Maillage réalisé pour construire la surface des RS entre deux couches n’ayant pas le même nombre de points.

Si la surface est fermée, nous relierons les points extrêmes entre eux. Si la surface est ouverte, nous ne les relierons pas.

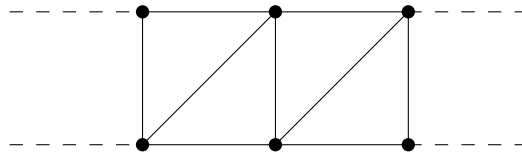


FIGURE 4.1 – Maillage réalisé pour construire la surface des RS entre deux couches ayant le même nombre de points.

Pour introduire un triangle supplémentaire quand les deux couches successives n’ont pas le même nombre de points, nous procédons de la même manière que pour supprimer des points : nous ajoutons régulièrement des triangles en fonction du ratio entre le nombre de points des deux lignes polygonales (cf. figure 4.2).

4.1.1.2 Interpolation de l’occupation des obstacles

Comme pour les RS, l’occupation des obstacles est connue à des instants donnés dans l’espace-temps. Nous devons donc interpoler l’occupation des obstacles à partir des occupations à chaque instant discret. Dans le cas des obstacles, l’utilisation de polygones permet de calculer l’enveloppe convexe de l’occupation de deux couches successives. Nous obtenons le volume d’occupation total en calculant l’union des volumes trouvés pour chaque couche.

4.1.1.3 Calcul d’intersection dans le cas d’une interpolation *a priori*

Une fois la surface construite, il nous faut calculer son intersection avec les volumes correspondant aux obstacles comme dans la figure 4.3.

L’approche RIS utilisant les RS basés sur le modèle de Dubins a été validée dans [de Beaucorps *et al.*, 2018] sur un ensemble de tests similaires à ceux présentés dans [Chiang *et al.*, 2017a].

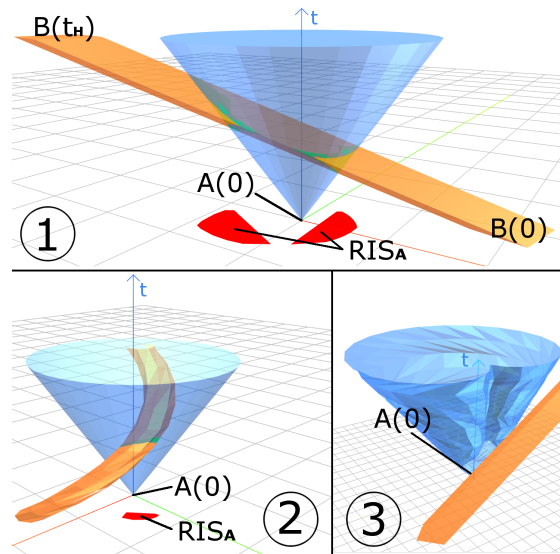


FIGURE 4.3 – RS dans l'espace-temps et RIS projetés à l'instant $t = 0$.

4.1.2 Validation expérimentale

Dans tous les scénarios présentés, l'ego-véhicule part d'un point donné et doit rejoindre un point cible sans entrer en collision avec un obstacle fixe ou mobile. Si l'ego-véhicule rejoint le point cible sans collision, le scénario se termine avec succès. Sinon, il s'agit d'un échec. Nous appelons "taux de succès" ou "résultat" le ratio du nombre de succès divisé par le nombre de scénarios. Il est exprimé en pourcentage.

4.1.2.1 Présentation des environnements

Trois types d'environnements ont été développés pour tester les performances des différentes méthodes. Les environnements contiennent des obstacles statiques et/ou dynamiques. Plus précisément, les différents environnements considérés sont les suivants :

- "Open", qui n'a pas d'obstacle statique et où les obstacles mobiles n'ont pas de direction particulière.
- "Lanes", où les obstacles mobiles suivent une voie et où il n'y a pas d'obstacle statique.
- "Door", où l'ego-véhicule doit franchir une porte et où les obstacles dynamiques n'ont pas de direction particulière.

Nous différencions l'environnement du scénario. Les trois environnements sont déclinés en différents scénarios qui dépendent du comportement des obstacles dynamiques, à savoir :

- "random" pour aléatoire : un scénario où la vitesse des obstacles est choisie

au hasard à chaque pas de temps.

- "spread" pour séparé : les obstacles sont aussi stochastiques mais leurs vitesses possibles peuvent être très lentes ou très rapides.

Pour chaque type de scénario, le nombre d'obstacles peut varier. Un scénario est donc défini par son type et par son nombre d'obstacles. Nous noterons donc "spread-75" un scénario de type "spread" dans lequel évoluent 75 obstacles dynamiques. Les scénarios de type "random" sont des scénarios inspirés de [Chiang *et al.*, 2017a]. Cependant, les valeurs introduites dans [Chiang *et al.*, 2017a] ont été modifiées en respectant les échelles pour uniformiser les scénarios. Les approches présentées dans [Chiang *et al.*, 2017a] obtiennent des résultats similaires dans les scénarios que nous avons adaptés. Les autres scénarios introduisent des variantes pour tester ces approches de façon plus approfondie, que ce soit en termes de densité d'obstacles, de valeurs de vitesse, ou de présence d'obstacles statiques.

Nous allons présenter ci-dessous les paramètres relatifs aux différents environnements et aux différents scénarios.

- Largeur : 1000 *pixels*
- Longueur : 1000 *pixels*
- Position initiale : [250, 500]
- Destination finale : [750, 500]
- Vitesse maximale de l'agent : 1.34 *pixels/step*
- Type d'ego-véhicule : ponctuel

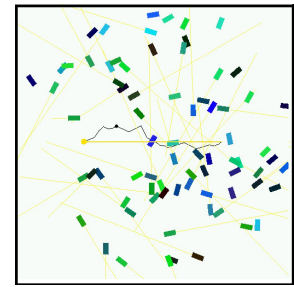


TABLEAU 4.1 – Environnement "Open".

- Largeur : 1000 *pixels*
- Longueur : 1000 *pixels*
- Position initiale : [500, 350]
- Destination finale : [500, 650]
- Vitesse maximale de l'agent : 1.34 *pixels/step*
- Type d'ego-véhicule : ponctuel

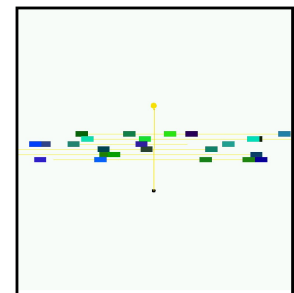


TABLEAU 4.2 – Environnement "Lanes".

- Largeur : 1000 *pixels*
- Longueur : 1000 *pixels*
- Position initiale : [200, 900]
- Destination finale : [800, 900]
- Vitesse maximale de l'agent : 1.34 *pixels/step*
- Type d'ego-véhicule : ponctuel
- Obstacles statiques : représentés en noir sur la figure de droite. Les zones marron représentent l'obstacle statique avec une marge.

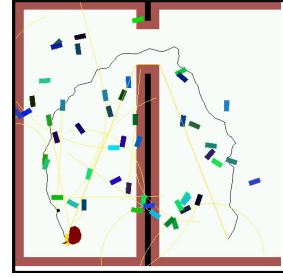


TABLEAU 4.3 – Environnement "Door".

- Forme des obstacles : rectangle 42×18 *pixels*
- Vitesses possibles des obstacles : [1.13, 1.69, 2.24, 2.81, 3.375] *pixels/step*
- Angle de braquage : trajectoire rectiligne
- Position initiale : choisie au hasard à l'initialisation

TABLEAU 4.4 – Scénarios "random".

- Forme des obstacles : rectangle 42×18 *pixels*
- Vitesses possibles des obstacles : [0.8, 0.8, 2.24, 4.0, 5.36, 5.36] *pixels/step*
- Angle de braquage : trajectoire rectiligne
- Position initiale : choisie au hasard à l'initialisation

TABLEAU 4.5 – Scénarios "spread".

4.1.2.2 Prise en compte des contraintes de mouvement pour les véhicules non-holonomes

Pour connaître l'apport de la prise en compte des contraintes de mouvement dans les RS des véhicules non-holonomes, nous comparons les résultats de la méthode RIS-path pour un véhicule non-holonyme dont les RS calculées au préalable tiennent compte des contraintes de mouvement de l'ego-véhicule dans un cas et n'en tiennent pas compte dans l'autre. En pratique, la première méthode utilise la surface de Dubins comme RS alors que la deuxième utilise le cône généré pour les véhicules holonomes.

L'utilisation des RS non-holonomes avec des obstacles non stochastiques améliore légèrement le taux de succès comme le montre la figure 4.4 relative à un scénario réalisé dans un environnement "door" et présenté dans [de Beaucorps, 2019].

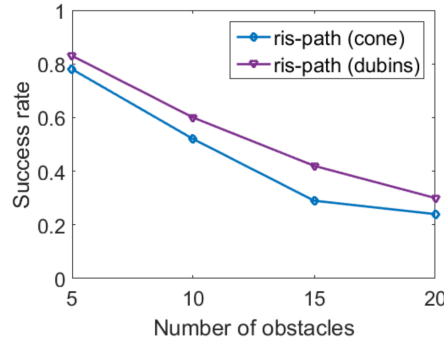


FIGURE 4.4 – Comparaison dans un environnement "door" de la méthode RIS-path avec un cône puis une surface de Dubins comme RS.

4.1.2.3 Méthodes développées pour les tests d'évaluation

Pour de comparer les RIS avec les méthodes existantes comme dans [Chiang *et al.*, 2017a], nous avons développé les approches suivantes :

- APF-statique [Latombe, 1991], qui construit un champ potentiel artificiel à partir des zones occupées par des obstacles à chaque instant.
- APF-dynamique [Ko et Lee, 1996], qui construit elle aussi un champ potentiel artificiel à partir des zones occupées par des obstacles mais en tenant compte de leur vitesse.
- APF-SR [Chiang *et al.*, 2015a], qui construit un champ potentiel artificiel à partir d'un champ scalaire. Ce champ scalaire est construit en additionnant les probabilités de présence des obstacles de l'instant initial à l'instant t_h .
- SES [Chiang *et al.*, 2015b], qui construit un RRT en testant pour chaque ajout d'un point dans l'arbre sa probabilité d'être dans un obstacle en utilisant un seuil.
- DRT [Chiang *et al.*, 2017b], qui procède de la même manière que la méthode SES en modifiant la loi de sélection des points du RRT pour autoriser un risque plus grand quand les instants considérés sont éloignés de l'instant présent.

Pour utiliser la répartition de probabilités décrite dans la section 3.2.2.2, les trois méthodes SES, DRT et APF-SR calculent à chaque instant la position (ou future position) relative de l'agent par rapport aux obstacles. Nous avons obtenu des performances similaires à celles présentées dans [Chiang *et al.*, 2017a], comme nous le verrons par la suite.

a) Description des approches Les méthodes testées dans les évaluations expérimentales sont rapidement présentées ci-dessous :

- APF-SR : la méthode APF-SR utilise la répartition de probabilités de présence des obstacles présentée en section 3.2.2.2. En additionnant les probabilités obtenues de l’instant initial à t_h , elle obtient un champ scalaire. À partir de celui-ci, la méthode calcule un champ de potentiel artificiel.
- SES : la méthode SES utilise la répartition de probabilités de présence des obstacles présentée en section 3.2.2.2. La méthode construit un RRT. Pour ce faire, elle sélectionne un point de l’environnement au hasard qu’elle relie au nœud le plus proche dans l’arbre (la position de l’agent étant ajoutée dans l’arbre au préalable). Sur ce segment, elle calcule le point qui peut être atteint en une itération depuis la position du nœud. Enfin, elle teste que la probabilité de présence d’un obstacle en ce point à l’instant $t + 1$ soit inférieure à un seuil. Si tel est le cas, le point est ajouté à l’arbre. Sinon, il n’est pas ajouté. La méthode recommence cette procédure jusqu’à avoir ajouté le nombre de points maximal autorisé. Une fois cet arbre construit, à chaque point est associé un coût calculé en fonction de la distance du point au but et de la probabilité de présence d’un obstacle en ce point. La méthode sélectionne le point présentant le coût le plus bas. Celui-ci et tous ses parents constituent un chemin.
- DRT : la méthode DRT est une extension de SES. Elle remplace le seuil de probabilité statique par un seuil dynamique. L’idée est d’augmenter ce seuil avec le temps.
- RIS-APF : la méthode RIS-APF utilise les RIS comme obstacles pour calculer un champ de potentiel artificiel.
- RIS-path : la méthode RIS-path utilise le graphe de visibilité [Lozano-Pérez et Wesley, 1979] qui permet de calculer le chemin optimal parmi des obstacles statiques polygonaux. Elle remplace les obstacles par les RIS comme dans [de Beaucorps *et al.*, 2018, de Beaucorps, 2019]. Les mêmes travaux ont étendu les graphes de visibilité aux véhicules non-holonomes.
- static-APF : la méthode statique-APF utilise la position des obstacles à chaque instant pour calculer un champ de potentiel artificiel.
- dynamic-APF : la méthode dynamic-APF est une extension de static-APF qui tient compte de la vitesse des obstacles pour calculer un champ de potentiel artificiel.

b) Optimisation des différentes approches Pour comparer les approches testées sur les RIS avec d’autres méthodes de l’état de l’art, les différentes méthodes ont été optimisées pour les scénarios utilisés pour la comparaison. Pour cela, les dif-

férents paramètres des approches ont été échantillonnés et évalués séparément pour essayer de se rapprocher de leurs optimums. Pour l'approche DRT, les paramètres optimisés sont :

- "switchTime" : le seuil de probabilité de présence au-dessus duquel un point du RRT est supprimé devient dynamique. Le paramètre "switchTime" est l'instant à partir duquel le seuil passe d'une valeur constante à une loi qui est fonction du temps.
- "nSafety" : il s'agit de la longueur minimale de validité d'un chemin en nombre de points.
- "nSafetyEmergency" : il s'agit de la valeur que prend nSafety en cas d'urgence.

Voici un exemple des résultats obtenus pour l'approche DRT [Chiang *et al.*, 2017b] dans l'environnement "open" sur des scénarios de type "spread-150" :

switchTime		nSafety		nSafetyEmergency	
valeur	résultat	valeur	résultat	valeur	résultat
5	0.55	3	0.47	10	0.6
10	0.64	5	0.56	25	0.56
20	0.65	10	0.65	40	0.65
30	0.65	15	0.68	50	0.66
40	0.59	20	0.65	75	0.58
75	0.61	35	0.78	100	0.53
		80	0.09		

TABLEAU 4.6 – Tableau comparatif des résultats moyens de la méthode DRT. Résultats en fonction de différentes valeurs de ses paramètres (afin d'optimiser chaque valeur indépendamment, les autres paramètres sont laissés à une valeur de référence ($switchTime = 20$, $nSafety = 10$, $nSafetyEmergency = 40$)). Les valeurs en rouge sont choisies pour la suite. (Les taux de réussite sont calculés sur 110 scénarios).

Ces résultats ont été obtenus en affinant les paramètres en fonction de la tendance des résultats précédents (par exemple, pour le paramètre "switchTime" le calcul a d'abord été réalisé pour les valeurs 5, 30, 75 avant d'ajouter des valeurs pour faire ressortir le maximum). Certains résultats ne dégagent pas de tendance claire : dans ce cas, la valeur du paramètre ayant obtenu le meilleur score a été choisie malgré le manque de finesse du choix des valeurs.

Pour l'approche SES testée dans l'environnement "open" pour des scénarios "random-75", les résultats sont :

nSafety	
valeur	résultat
10	0.83
25	0.96
35	0.81
50	0.46

TABLEAU 4.7 – Tableau comparatif des résultats moyens de la méthode SES. Résultats en fonction de différentes valeurs de son paramètre. La valeur en rouge est choisie pour la suite. (Les taux de réussite sont calculés sur 110 scénarios)

4.1.2.4 Influence de la prise en compte du comportement stochastique des obstacles dans l'approche RIS-path

Pour montrer l'influence de la prise en compte du comportement stochastique des obstacles sur le taux de réussite de la méthode RIS-path, nous comparons RIS-path avec une version dégénérée de la méthode qui tient compte de la vitesse moyenne des obstacles pour créer leur occupation et plutôt que d'effectuer un tirage de Monte-Carlo pour obtenir une répartition de probabilités de présence (section 3.2.2.2). La comparaison a été effectuée sur l'environnement "open" pour des scénarios de type "spread". Les résultats sont résumés par la figure 4.5 :

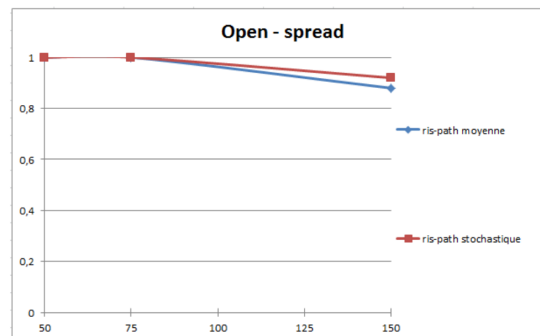


FIGURE 4.5 – Résultats des approches RIS-path en tenant compte ou non du caractère stochastique des obstacles. Réalisé sur l'environnement "Open" pour des scénarios de type "spread" (Les taux de réussite sont calculés sur 110 scénarios).

Nous pouvons voir que la prise en compte du caractère stochastique des obstacles n'a pas une grande influence sur le taux de succès de l'approche RIS-path. Cela est probablement dû au fait que la vitesse des obstacles est mise à jour trop régulièrement dans tous les scénarios, la variance de la probabilité de présence de

l'obstacle diminue. L'amélioration devient notable quand le nombre d'obstacles augmente fortement.

4.1.2.5 Stratégie d'urgence pour l'ego-véhicule : comparaison de RIS-path projection et RIS-path horizon

Les deux versions de l'approche RIS-path présentées dans le paragraphe (b) de la section 3.2.2.3 ont été comparées dans le but de sélectionner la meilleure stratégie d'urgence pour l'approche finale. Les deux versions ont été comparées dans l'environnement "open" car la présence d'obstacles statiques a peu d'influence sur les résultats sachant que la différence entre les versions ne porte que sur les obstacles dynamiques. En effet, pour la version "projection", l'ego-véhicule à l'arrêt ne peut être percuté par un obstacle statique. Quant à la version "horizon", les obstacles statiques ne sont pas impactés par la réduction d'horizon car leur projection sur le plan $t = 0$ reste la même.

Les deux versions ont été testées pour différents scénarios avec un véhicule holonome puis avec un ego-véhicule non-holonome. Les scénarios choisis pour le véhicule holonome sont de type "spread-150". En effet, le taux de réussite étant élevé pour l'ensemble des scénarios, choisir un type de scénario difficile (avec un nombre supérieur d'obstacles dynamiques) permet d'avoir plus de chance de remarquer une différence notable entre les deux versions.

Nous obtenons dans l'environnement "open" avec des scénarios de type "spread-150" les résultats figurant sur la figure 4.6 :

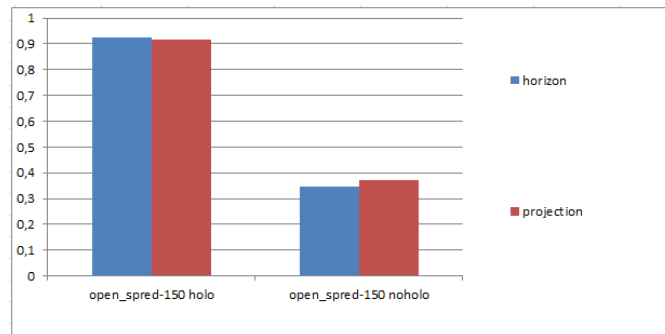


FIGURE 4.6 – Comparaison du taux de succès des deux versions de RIS-path sur trois types de scénarios dans l'environnement "open" (les taux de réussite sont calculés sur 110).

Les deux approches ont des résultats très proches. Nous avons choisi d'utiliser l'approche RIS-path horizon dans la suite des tests.

4.1.2.6 Résultat du RIS-path stochastique

Dans cette dernière section, nous présentons un benchmark identifiant les points forts et les points faibles de l'approche RIS-path par rapport aux approches de l'état de l'art présentées dans le paragraphe (a) de la section 4.1.2.3.

a) Mouvement holonome de l'ego-véhicule

Examinons ici les résultats obtenus par les différentes approches pour les environnements présentés précédemment quand l'ego-véhicule a un mouvement holonome.

Open L'environnement "open" est un environnement classique pour la planification de trajectoire en présence d'obstacles dynamiques. En effet, l'absence d'obstacle statique et la présence de très nombreux obstacles dynamiques permet de montrer l'aptitude des approches à éviter les collisions en présence de tels obstacles. Les résultats sont résumés par les figures 4.7 et 4.8.

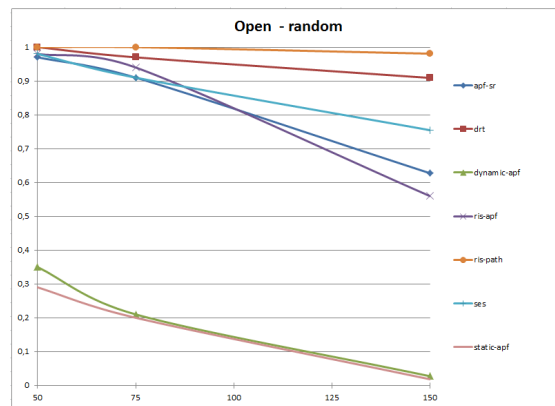


FIGURE 4.7 – Résultats des différentes approches de l'état de l'art sur l'environnement "Open" pour des scénarios de type "random" (les taux de réussite sont calculés sur 110 scénarios).

Pour les scénarios "random" nous identifions trois catégories d'approches :

- les approches qui ne tiennent pas compte du comportement stochastique des obstacles :
 - statique-APF ;
 - dynamique-APF ;
- les approches qui tiennent compte du caractère stochastique des obstacles, mais qui ne parviennent pas à gérer un trop grand nombre d'obstacles :
 - RIS-APF ;
 - APF-SR ;
 - SES ;

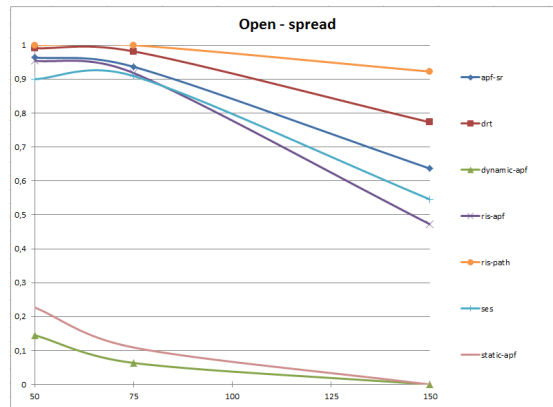


FIGURE 4.8 – Résultats des différentes approches de l'état de l'art sur l'environnement "Open" pour des scénarios de type "spread" (les taux de réussite sont calculés sur 110 scénarios).

- et les approches qui tiennent compte du caractère stochastique des obstacles et qui arrivent à gérer un grand nombre d'obstacles :
 - DRT ;
 - RIS-path.

Nous constatons que l'approche RIS-path a un taux de succès sensiblement plus élevé que DRT. Cet écart est encore plus important dans les scénarios de type "spread" et lorsque le nombre d'obstacles augmente.

Lanes L'environnement "lanes" met à l'épreuve la prévision que peuvent faire les approches. En effet, les approches trop réactives auront tendance à s'engager sur la route dès que la première voie est libre sans tenir compte de ce qui se passe sur les voies suivantes. De tels comportements mènent généralement à des situations de non-retour. C'est également un scénario classique pour la planification de trajectoire dans un environnement comportant des obstacles dynamiques : il correspond au cas d'un véhicule devant traverser une route sur laquelle circulent d'autres véhicules. Les résultats sont présentés dans les figures 4.9 et 4.10.

Les trois catégories définies plus haut persistent. Nous remarquons quand même que le nombre d'obstacles n'étant pas aussi important que dans l'environnement "open", cet environnement favorise les approches qui permettent d'avoir la meilleure prévision sur le moyen terme et ainsi évitent ainsi les situations de non-retour. Nous remarquons que la méthode RIS-APF (champ de potentiel artificiel construit grâce au RIS) est un peu moins performante que DRT et RIS-path mais qu'elle obtient un très bon résultat pour une approche réactive (APF-SR ayant un taux de succès bien inférieur). Ses résultats sont supérieurs à ceux obtenus dans l'environnement

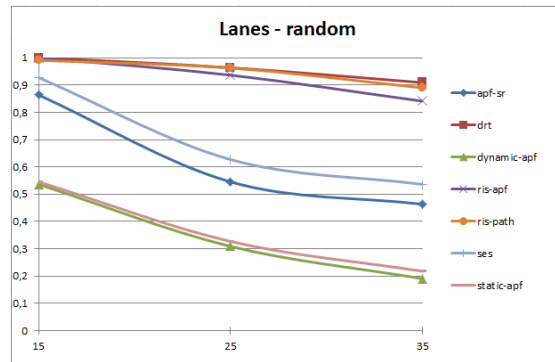


FIGURE 4.9 – Résultats des différentes approches de l'état de l'art sur l'environnement "Lanes" pour des scénarios de type "random" (les taux de réussite sont calculés sur 110 scénarios).

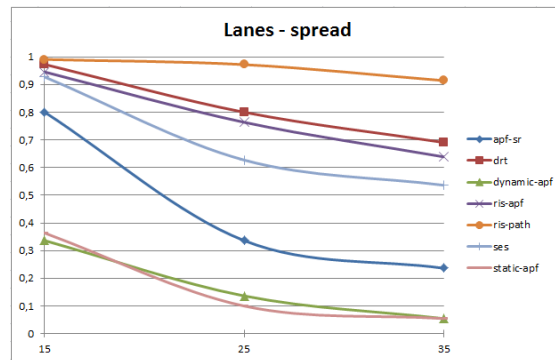


FIGURE 4.10 – Résultats des différentes approches de l'état de l'art sur l'environnement "Lanes" pour des scénarios de type "spread" (les taux de réussite sont calculés sur 110 scénarios).

"open".

La figure 4.10 nous confirme que l'approche RIS-path supporte bien les obstacles ayant des vitesses maximales et minimales très disparates, alors que les résultats des autres approches baissent en présence de ce type d'ego-véhicules.

Door

Le dernier environnement "door" combine des obstacles statiques et des obstacles dynamiques. Ceci nous permet de tester les approches sur un environnement plus contraint s'approchant donc davantage des situations de milieu urbain. Même si l'environnement est encore sommaire, nous pouvons repérer les approches les plus adaptées à ce type de contraintes. Les résultats sont présentés dans les figures 4.11 et 4.12.

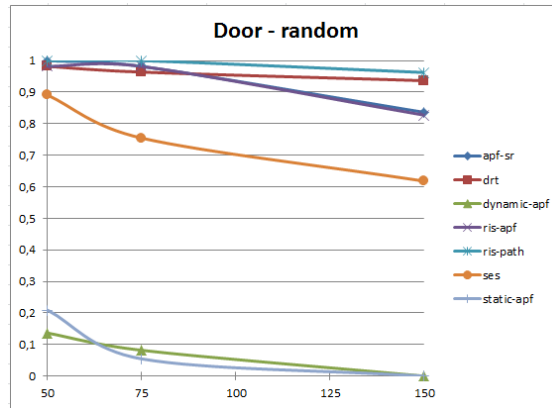


FIGURE 4.11 – Résultats des différentes approches de l'état de l'art sur l'environnement "Door" pour des scénarios de type "random" (les taux de réussite sont calculés sur 110 scénarios).

Nous remarquons que dans le scénario de type "spread", la méthode RIS-path voit ses résultats chuter légèrement en dessous de la méthode DRT. Ce n'est pas très étonnant : construire un chemin complet jusqu'à l'arrivée avec un graphe de visibilité est plus complexe qu'avec une RRT quand l'environnement devient plus contraint. En revanche, les chemins obtenus par la méthode RIS-path sont plus adaptés au milieu urbain car il y a moins de changement dans l'angle de braquage.

b) Mouvement non-holonyme de l'ego-véhicule

L'intérêt de la prise en compte des contraintes non-holonomes dans la construction des RIS a déjà été montré par [de Beaucorps *et al.*, 2018] (cf. figure 4.4). Nous voulons montrer que même avec les RS construit comme si l'ego-véhicule était holonome les RIS restent pertinents. Les approches SES et DRT tiennent compte des contraintes de mouvement de l'ego-véhicule. Nous comparons les approches RIS-path et DRT pour un véhicule non-holonyme. La comparaison n'est faite qu'avec la méthode DRT car l'objectif est de déterminer si RIS-path est toujours plus fiable que DRT pour des véhicule non-holonyme malgré la non-prise en compte des contraintes de mouvement (cf. section 4.1.2.2). Les expériences suivantes sont réalisées avec un véhicule de Dubins ayant les caractéristiques suivantes :

- $u_{max} = 0.09 \text{ rad/step}$
- $a_{max} = 0.7 \text{ pixels/seconde}^2$

Même si les RIS sont réalisés avec un cône comme RS pour l'ego-véhicule, le chemin tient compte des contraintes d'un véhicule non-holonyme. Nous utiliserons la méthode de recherche de chemin développée par [de Beaucorps *et al.*, 2018].

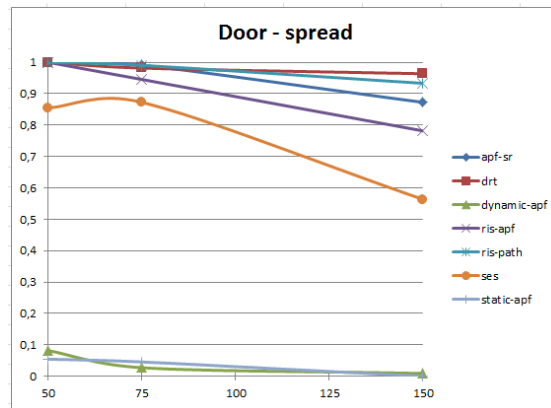


FIGURE 4.12 – Résultats des différentes approches de l'état de l'art sur l'environnement "Door" pour des scénarios de type "spread" (les taux de réussite sont calculés sur 110 scénarios).

Open Cette comparaison est faite sur l'environnement "open" pour des scénarios de type "spread". Les résultats sont résumés par la figure 4.13 :

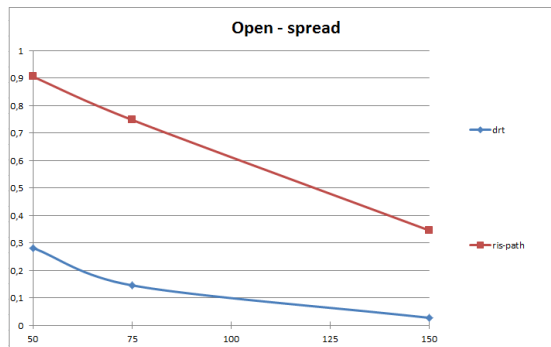


FIGURE 4.13 – Résultats des approches RIS-path et DRT sur l'environnement "Open" pour des scénarios de type "spread" (les taux de réussite sont calculés sur 110 scénarios).

Nous constatons que les deux méthodes sont très impactées par les contraintes non-holonomes. Ce résultat est attendu car tous les scénarios sont beaucoup plus difficiles pour des véhicules non-holonomes. Cependant, nous voyons que le taux de succès de l'approche DRT a chuté davantage que l'approche RIS-path, même pour les scénarios impliquant 50 obstacles où RIS-path garde un taux de succès honorable de 90,4%.

4.2 Scénarios et résultats de l’approche SGSPA

Des évaluations expérimentales ont été réalisées à l’aide du simulateur basé sur CARLA présenté dans l’Annexe 6.3 dans un environnement composé de plusieurs carrefours espacés de 40 m avec un immeuble cachant certaines parties de la route (exemple vidéo : <https://team.inria.fr/rits/membres/renaud-poncelet/>). Tous les autres véhicules sont contrôlés par un agent CARLA. Ils suivent un chemin aléatoire guidé par les voies. Plus précisément, ils peuvent tourner à gauche, à droite ou continuer tout droit au niveau des carrefours de façon aléatoire. Ils roulent à la vitesse maximale v_{max} , sauf lorsqu’un autre véhicule se trouve dans une zone de contrôle de dix mètres. Dans ce cas, ils freinent. Plusieurs scénarios ont été évalués dans lesquels l’ego-véhicule doit traverser un carrefour avec trois comportements différents :

- Omniscient : l’ego-véhicule connaît la position et la vitesse de tous les véhicules présents dans la scène, y compris ceux qui sont masqués.
- Naïf : l’ego-véhicule réagit par rapport aux véhicules visibles et ne prend pas en compte les parties occultées de la scène.
- SGSPA (Safe Geometric Speed Planning Approach) : l’ego-véhicule agit conformément à l’approche complète présentée dans la section 3.3.

Pour chaque comportement, le mouvement de l’ego-véhicule est simulé à l’aide d’un contrôle longitudinal décrit dans la section 3.3.1.5. Les paramètres suivants ont été utilisés lors des essais :

$$v_{max} = 8,3 \text{ m/s}, \quad (4.3)$$

$$a_{max} = 2,5 \text{ m/s}^2, \quad (4.4)$$

$$a_{min} = -4 \text{ m/s}^2, \quad (4.5)$$

$$\Delta T = 5 \text{ s}, \quad (4.6)$$

$$dt = 1/20 \text{ s}, \quad (4.7)$$

$$t_{gap} = 2 \text{ s}, \quad (4.8)$$

$$marge = 1 \text{ m} \quad (4.9)$$

La valeur de s_h est donnée par l’équation :

$$s_h = 2v_{max}t_h \quad (4.10)$$

4.2.1 Traversée d’un carrefour en X

La première situation est un carrefour en X où la route de droite est partiellement obstruée par un bâtiment. L’ego-véhicule doit traverser l’intersection et aller tout droit. Deux scénarios sont présentés :

4.2.1.1 Zone occultée vide

Dans ce premier cas, la route de droite ne contient aucun véhicule caché par le bâtiment. Le véhicule virtuel créé par la méthode SGSPA ne correspond ici à aucun véhicule réel (cf. figure 3.26).

Commentons le mouvement de l'ego-véhicule calculé par SGSPA et illustré dans la figure 3.26 :

- lorsque le véhicule virtuel est loin de l'intersection ($t = 0, 0 s$), la limite de la zone occultée est proche de l'intersection et le TP obstacle associé au véhicule virtuel est très long. L'algorithme prévoit donc d'aller à la vitesse maximale et de freiner lorsque la distance de freinage est atteinte ;
- plus tard ($t = 3, 6 s$), l'ego-véhicule atteint l'intersection et dispose d'une visibilité suffisante pour la traverser en toute sécurité car la limite de la zone d'occultation est plus éloignée ;
- enfin ($t = 5.8 s$), l'ego-véhicule traverse le carrefour et sa visibilité est complète.

Comme prévu, dans la figure 4.14, nous pouvons remarquer que les approches Omnisciente et Naïve planifient toutes deux leur vitesse de la même manière puisqu'il n'y a pas de véhicule caché.

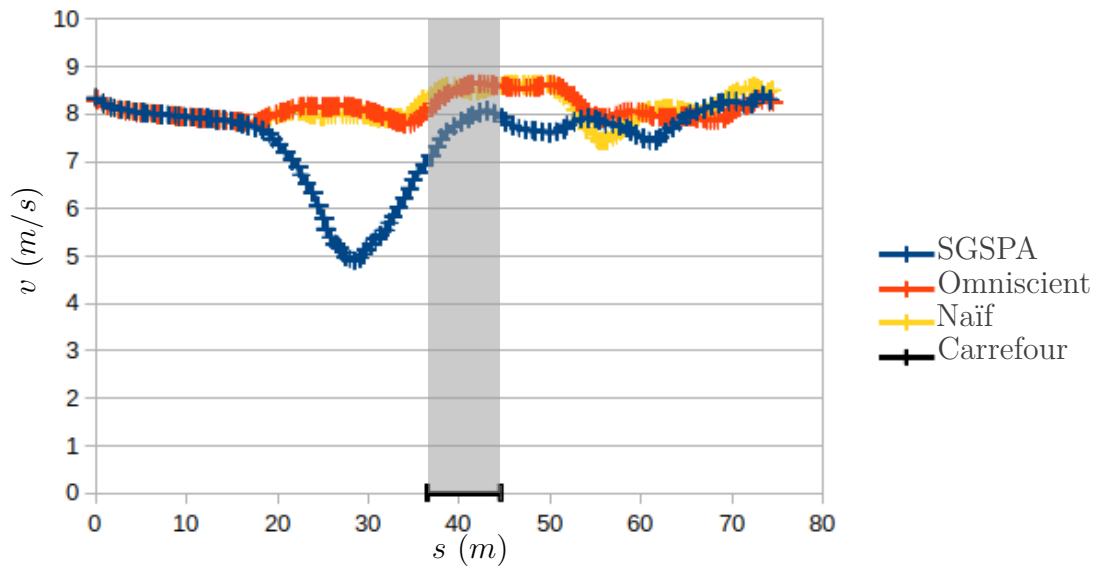


FIGURE 4.14 – Zone occultée vide : vitesse mesurée sur l'abscisse curviligne pour les trois approches.

4.2.1.2 Un véhicule est sur la partie occultée de la route

Dans ce scénario, l'autre véhicule roule à v_{max} , atteint le croisement approximativement au même pas de temps que l'ego-véhicule et s'y engage directement. L'ego-véhicule doit céder le passage à l'autre voiture. Examinons chaque approche, dans la figure 4.15 :

- dans la portion a, l'ego-véhicule est loin du carrefour et ne peut pas voir le véhicule caché. Les approches SGSPA et Naïve roulent à la vitesse maximale et l'approche Omnisciente ralentit. En fait, SGSPA réagit comme Naïve pour augmenter sa visibilité et Omniscient prend en compte le véhicule caché ;
- dans la portion b, SGSPA commence à freiner pour pouvoir s'arrêter avant le carrefour alors que l'approche Naïve va encore très vite ;
- dans la partie c, l'approche Naïve freine fortement pour éviter une collision avec l'autre véhicule, qui est maintenant visible. L'approche Omnisciente commence à accélérer car sa vitesse est suffisamment basse pour laisser passer l'autre véhicule.

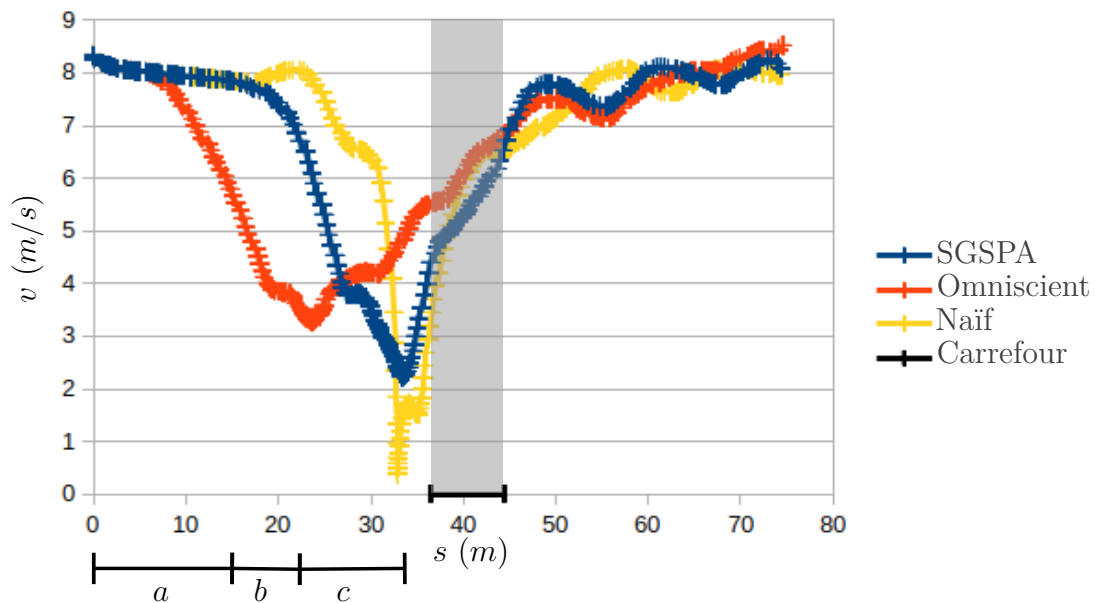


FIGURE 4.15 – Zone occultée avec un véhicule : vitesse mesurée sur l'abscisse curviligne pour les trois approches.

Toutes les approches ont passé l'intersection sans collision, mais seules les approches Omnisciente et SGSPA ont calculé des mouvements confortables.

4.2.2 Traversée d'un carrefour en T avec visibilité limitée

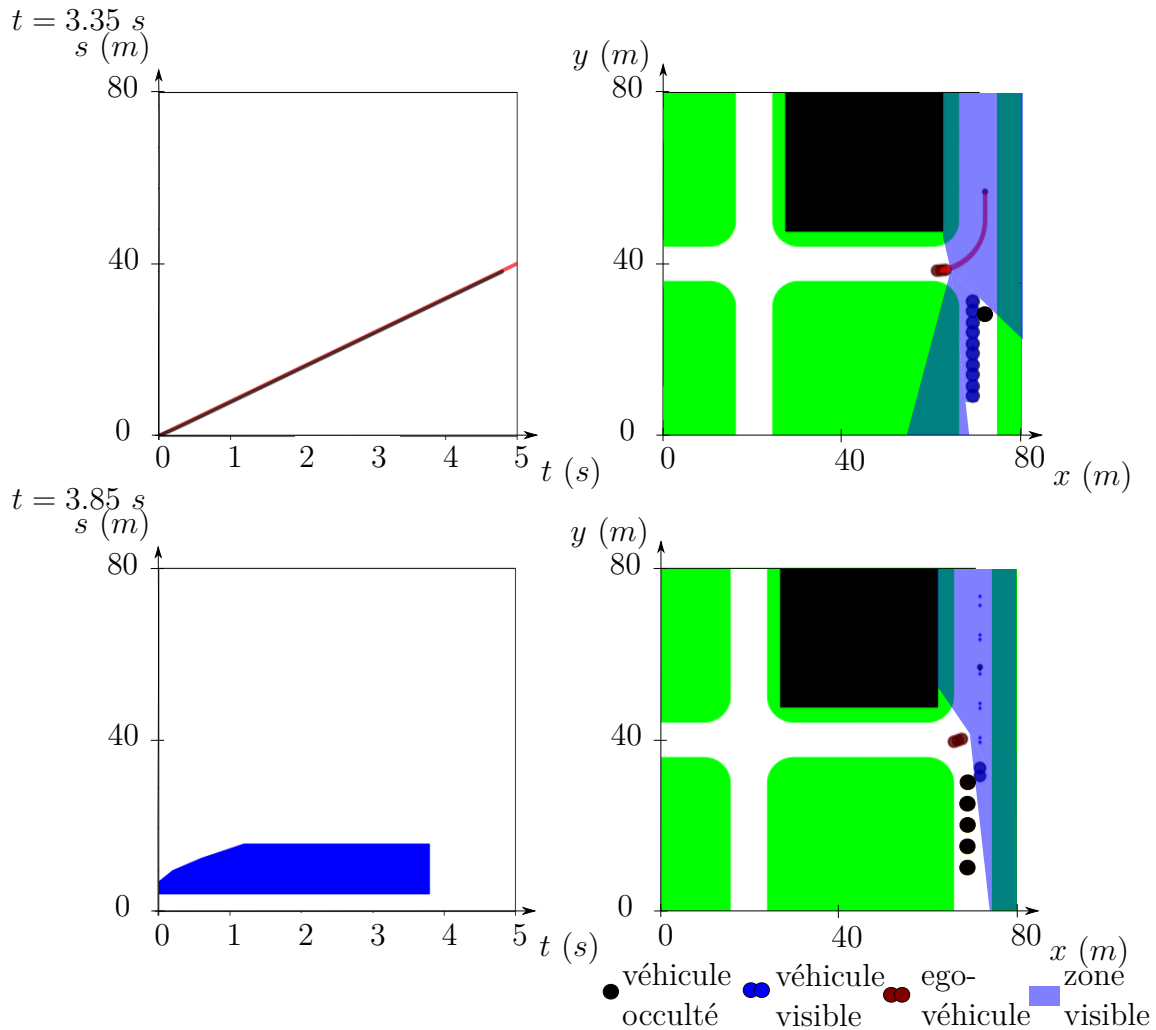


FIGURE 4.16 – Carrefour en T avec une visibilité limitée.

Sans tenir compte des parties occultées de la scène, l'approche Naïve n'anticipe pas l'arrivée du véhicule caché sur la droite et la collision ne peut être évitée.

Comparons maintenant les trois méthodes sur un scénario inspiré de [Tas et Stiller, 2018] : l'ego-véhicule doit tourner à gauche à un carrefour en T où la visibilité est limitée. Comme l'illustre la figure 4.16, les deux côtés sont occultés : à gauche par un bâtiment et à droite par une ligne de voitures arrêtées sur la route. La visibilité est davantage limitée sur le côté droit où l'arrivée d'un véhicule caché oblige l'ego-véhicule à lui céder le passage. Dans cet exemple, l'approche Naïve ne peut pas freiner assez fort pour éviter la collision, tandis que

les autres approches réduisent leur vitesse pour laisser passer l'autre véhicule, comme le montre la figure 4.17.

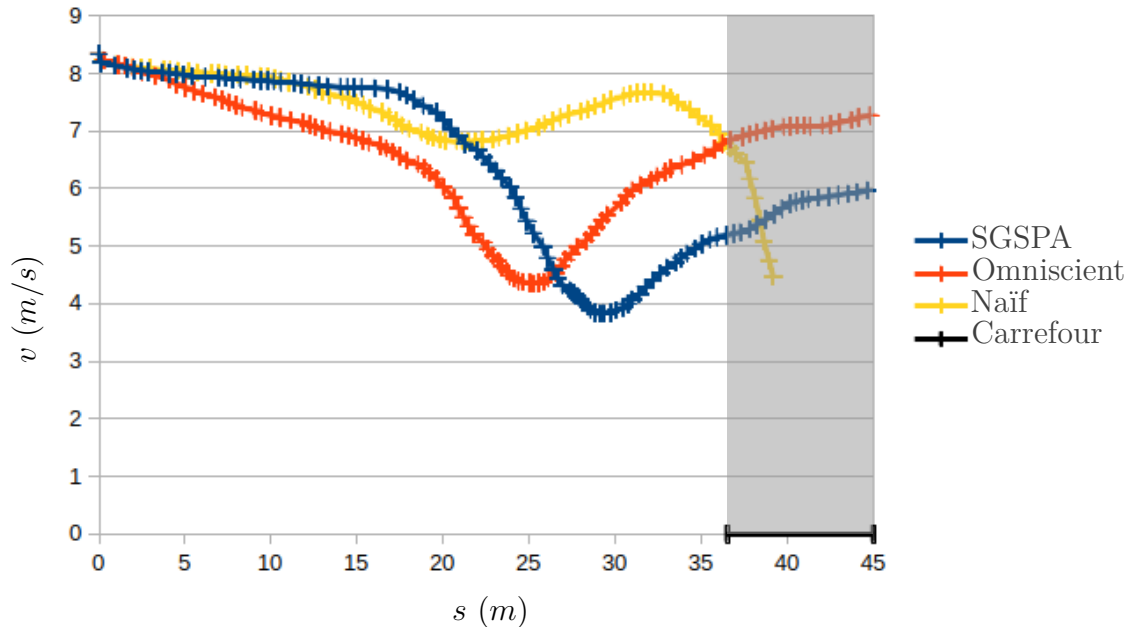


FIGURE 4.17 – Carrefour en T : vitesse mesurée sur l'abscisse curviligne pour les trois méthodes.

4.2.3 Traversée d'un carrefour en X avec dix voitures

Enfin, pour évaluer la sécurité de l'approche dans des cas plus complexes, nous avons créé 50 scénarios pour l'environnement du carrefour X de la section 4.2.1 avec dix véhicules positionnés aléatoirement sur les routes et suivant un chemin aléatoire. Les dix voitures ne peuvent pas toutes interagir avec l'ego-véhicule en même temps : par exemple, dans la figure 4.18, seules deux voitures sont impliquées. Notez que tous les véhicules ne sont pas cachés par le bâtiment. Dans tous les scénarios, l'ego-véhicule a dû traverser le carrefour en X. Nous avons comparé les deux comportements suivants :

- GSPA (Geometric Speed Planning Approach - approche de planification de la vitesse géométrique) elle ne garantit pas d'état de sécurité avant les intersections.
- SGSPA : l'ego-véhicule doit s'arrêter avant le carrefour s'il ne peut pas le traverser.

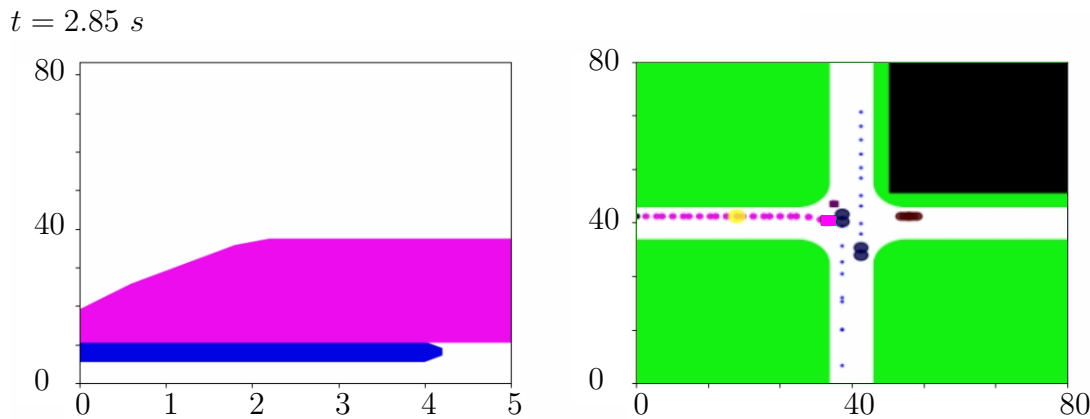


FIGURE 4.18 – Carrefour en X avec deux autres voitures arrivant à l'intersection en même temps.

L'ego-véhicule est à l'arrêt avant l'intersection. Le TP obstacle en bleu correspond à la voiture arrivant sur la gauche de l'ego-véhicule. Le TP obstacle correspondant à la voiture réelle arrivant par la droite est masqué par le TP obstacle virtuel qui représente la possibilité de la présence d'une voiture arrivant par la droite.

Comparons le taux de réussite et le temps passé dans le carrefour dans les deux cas et examinons les cas où un accident s'est produit :

- pour GSPA, sans état sûr avant les intersections, neuf collisions se sont produites (taux de réussite : 82 %), dont huit pour lesquelles l'ego-véhicule était à l'arrêt au milieu du carrefour. Il ne s'agit pas d'une sécurité passive comme définie par [Shalev-Shwartz *et al.*, 2017], car le véhicule se trouvait à l'intérieur du carrefour. Dans le dernier cas, l'ego-véhicule se déplaçait, mais l'autre véhicule en collision se trouvait derrière l'ego-véhicule. Le temps moyen passé dans le carrefour, en ne gardant que les cas de succès, est de 4,3 secondes ;
- pour SGSPA, nous avons obtenu un taux de réussite de 100 % et le temps moyen passé par l'ego-véhicule dans le carrefour était de 3,8 secondes

4.2.4 Conclusion

Nous avons proposé une approche géométrique calculant une planification de la vitesse sûre et confortable pour la conduite autonome en zone urbaine y compris dans le cas où certaines parties des voies sont occultées. Les évaluations montrent que la méthode SGSPA réalise à la fois les objectifs de sécurité et de confort dans tous les scénarios. L'approche est prudente mais pas trop conservatrice, grâce au profil de vitesse que nous choisissons à l'approche d'une région avec occultations :

nous décidons de nous déplacer rapidement vers l'intersection afin d'augmenter la visibilité de l'ego-véhicule tout en gardant suffisamment d'espace pour freiner. Cette approche peut être généralisée à d'autres obstacles dynamiques (tels que les piétons) en considérant l'existence possible et l'intention d'agents cachés comme [Nager *et al.*, 2019]. Cependant, dans certains cas de visibilité vraiment limitée, nous ne pouvons pas garantir que l'ego-véhicule ne soit bloqué dans un "état de gel" comme discuté dans [Lin *et al.*, 2019]. Dans ce cas, un conducteur humain conduira très lentement afin d'augmenter sa visibilité.

4.3 Scénarios et résultats de l'approche GPAD

4.3.1 Présentation de l'environnement de tests

Des évaluations expérimentales ont été réalisées à l'aide du simulateur basé sur CARLA présenté dans l'annexe 6.3. La carte intégrée à Carla "Town02" (cf. figure 4.19) a été utilisée pour réaliser les évaluations.

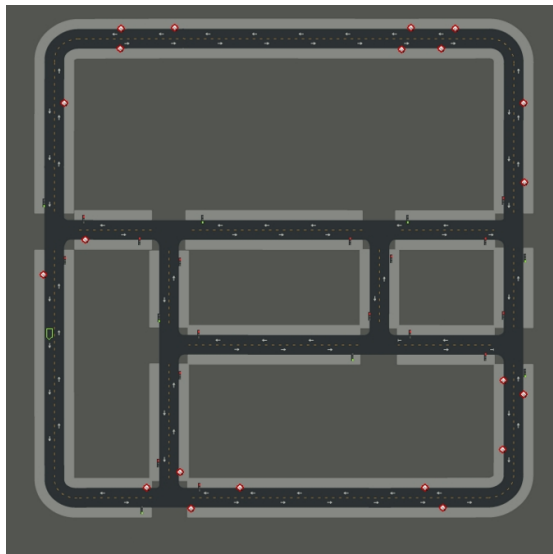


FIGURE 4.19 – Image vue du dessus de la carte "Town02" de CARLA. Les voies sont dessinées en noir et sont séparées par une bande discontinue jaune. Le sens de circulation est indiqué par une flèche blanche. Les feux tricolores sont indiqués par un pictogramme.

Pour vérifier que l'alternance s'effectue bien entre l'adaptation en vitesse et le calcul de chemin, nous avons conçu différents scénarios dont certains sont basés sur une situation de dépassement en milieu urbain. L'ego-véhicule doit aller jusqu'à un

premier feu tricolore puis tourner à droite au carrefour (cf. section 4.3.2.1). Sur cette première portion, nous utilisons plaçons un véhicule à l'arrêt sur la voie de l'ego-véhicule pour le scénario présenté en section 4.3.2.2. Pour ne pas rester bloqué, il devient donc nécessaire de réaliser un dépassement du véhicule arrêté. Ce qui nous intéresse est de montrer des cas où la manœuvre de dépassement nécessite l'utilisation des RIS. Il faut alors que les chemins candidats ne soient pas libres de bout en bout. Pour cela, nous introduisons un scénario de dépassement où un autre véhicule se trouve sur la voie opposée, d'abord à l'arrêt comme dans le scénario de la section 4.3.2.3, puis en mouvement.

Les paramètres utilisés pour les scénarios sont les suivants :

- $time - gap = 2 \text{ s}$,
- $margin = 0.15 \text{ m}$,
- $N_{conf} = 6$,
- paramètres des APF : $k_1 = 5, k_2 = k_3 = k_4 = 25$,
- $N_{dym} = 20$,
- $t_h = 6 \text{ s}$,
- $N_{disques} = 3$,
- période de *cooldown* : 10 s ,
- $a_{max} = 2.5 \text{ ms}^{-2}$
- $a_{max,urgence} = 4 \text{ ms}^{-2}$,
- ratio entre la fréquence de planification et de contrôle : 10 ,
- $L = 3 \text{ m}$,
- $f_0 = 4, f_1 = 2$

4.3.2 Présentation des scénarios et résultats

4.3.2.1 Premier scénario : l'ego-véhicule circule tout seul

Ce premier scénario permet d'observer le comportement de l'approche GPAD lorsqu'il n'y a pas d'obstacle sur la route. Nous devrions observer uniquement la méthode de calcul d'un profil de vitesse le long d'un chemin.

Dans la figure 4.20, les profils de vitesse calculés par la méthode SGSPA dans l'espace TP sont affichés en haut à gauche. La trajectoire candidate est en rouge et le résultat final lissé est en noir. À droite, quatre cartes présentent les résultats de l'approche MMRIS-APF. La carte récapitulative est positionnée en haut à gauche. Sur cette carte, l'ego-véhicule est dessiné en rouge foncé, son chemin de référence en vert, et le chemin final calculé en rouge. L'ego-véhicule est positionné sur la route la plus à droite de la carte et se déplace de haut en bas sur cette voie. Les obstacles statiques sont représentés en gris. Les agents de l'environnement sont en bleu foncé. Leurs mouvements prévisionnels sont représentés en violet si les véhicules sont prioritaires et en bleu clair sinon. Au-dessus de cette carte, il est écrit "no mode

4. Résultats et simulation

selected". Aucun mode de la méthode MMRIS n'a été sélectionné dans ce scénario, ce qui est normal puisqu'en effet cette méthode n'y est pas utilisée. Les trois autres cartes donnent les résultats de MMRIS-APF en fonction des trois modes possibles (décélération, vitesse constante ou accélération). La méthode MMRIS-APF n'étant pas utilisée dans ce scénario, les trois cartes sont identiques. Une illustration de la simulation figurant plus bas montre que la route est dégagée devant l'ego-véhicule.

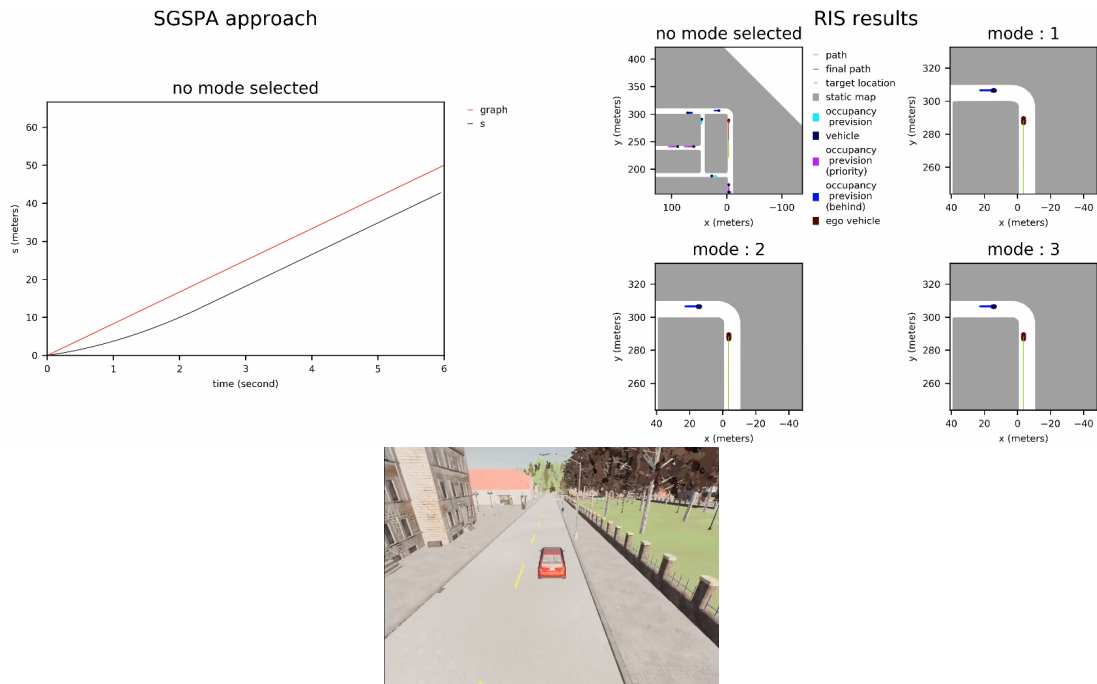


FIGURE 4.20 – Scénario où l'ego-véhicule ne rencontre pas d'obstacle sur sa voie. Début de la simulation.

Un peu plus tard, l'ego-véhicule s'approche d'un feu tricolore au rouge et doit s'arrêter, ce qui est illustré dans la figure 4.21. Le feu est représenté par un TP obstacle rouge dans l'espace TP de la méthode SGSPA. Le profil de vitesse calculé est tel que l'ego-véhicule s'arrête bien au feu rouge.

Enfin, quelques instants plus tard, le feu est repassé au vert (cf. figure 4.22). L'ego-véhicule a redémarré et a dépassé le feu tricolore. Nous voyons qu'un véhicule positionné plus loin sur la route contraint la méthode SGSPA à adapter la vitesse de l'ego-véhicule (TP obstacle en bleu) mais le suivi de chemin seul est suffisant pour finir le scénario.

Dans ce scénario, il n'a pas été nécessaire de recalculer un nouveau chemin et notre méthode de planification a fait uniquement appel au calcul d'un profil de vitesse via la méthode SGSPA.

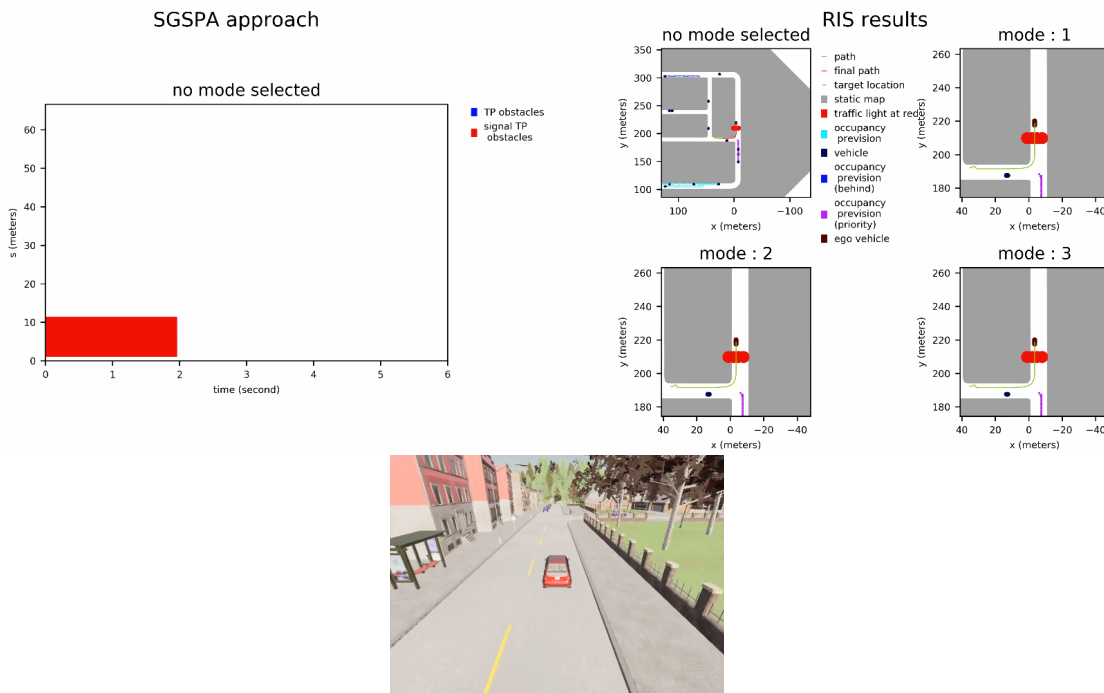


FIGURE 4.21 – Scénario où l’ego-véhicule ne rencontre pas d’obstacle sur sa voie. Arrêt à un feu rouge.

4.3.2.2 Deuxième scénario : un véhicule est arrêté sur la voie

Dans le second scénario, nous ajoutons un véhicule arrêté sur la chaussée. Nous nous attendons à ce que l’ego-véhicule doive dépasser ce véhicule et à ce que la méthode MMRIS-APF soit utilisée pour le calcul d’un nouveau chemin.

Au début de la simulation, dans la figure 4.23, un TP obstacle correspondant à un véhicule arrêté est inséré dans l’espace TP. La vitesse est suffisante pour respecter les critères des équations (3.73) et (3.74) et il n’est fait appel qu’au calcul d’un profil de vitesse.

Plus l’ego-véhicule avance et plus la vitesse planifiée est faible. À un moment, la vitesse maximale planifiée n’est plus suffisante pour respecter le critère de l’équation (3.74). Il faut alors calculer un nouveau chemin. Nous voyons dans la figure 4.24 qu’il a été fait appel à la méthode MMRIS-APF. Ici, seul le mode 3 d’accélération a été calculé. Sur la carte des RIS correspondant au mode 3, les chemins candidats qui sont aussi les RS sont dessinés en violet. Nous voyons dans la carte récapitulative des MMRIS-APF que le mode noté ici "*speed_up*" a été sélectionné. Le chemin final dessiné en rouge correspond au chemin candidat passant sur la voie opposée. Il a été sélectionné parmi les chemins candidats libres de bout en bout car il est au milieu de la voie.

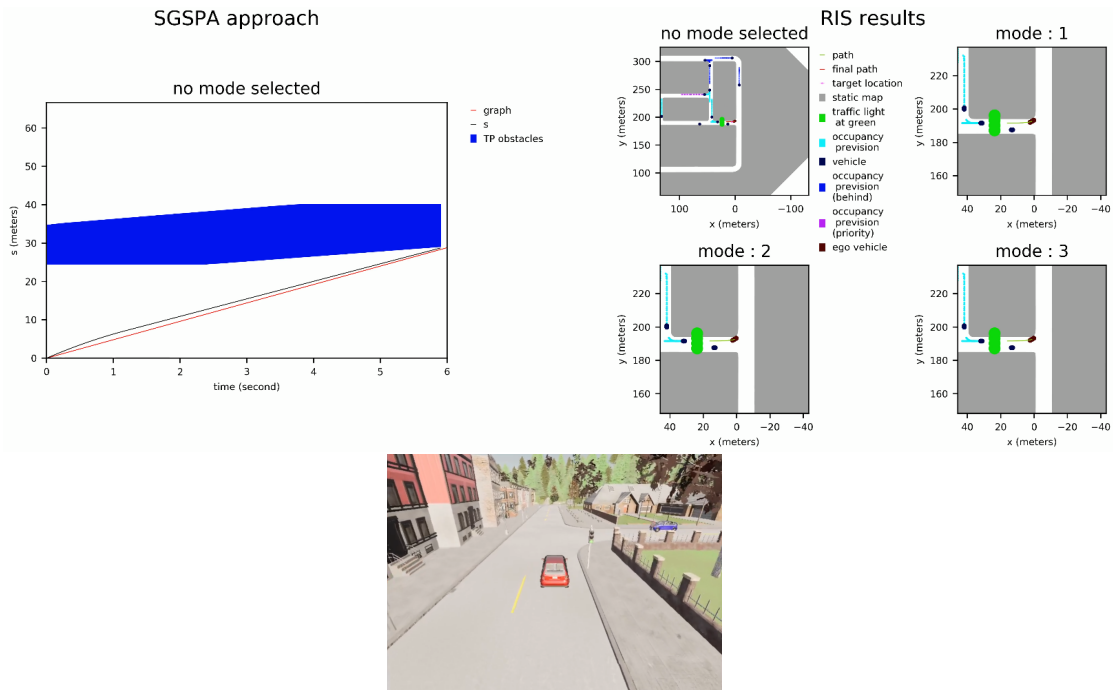


FIGURE 4.22 – Scénario où l’ego-véhicule ne rencontre pas d’obstacle sur sa voie. Le feu est passé au vert et un véhicule est positionné au loin.

Un peu plus tard, nous voyons qu’une fois le dépassement effectué, l’ego-véhicule se rabat bien sur sa voie (cf. figure 4.25). La méthode MMRIS-APF continue d’être utilisée car l’ego-véhicule ne se trouve pas sur sa voie de circulation normale : il est à contre-sens. Le chemin candidat arrivant au milieu de la voie de droite est favorisé par rapport aux autres chemins candidats libres.

4.3.2.3 Troisième scénario : deux véhicules sont arrêtés sur la voie

Ce troisième scénario a pour but de montrer le déroulement de l’approche dans un cas où les chemins candidats ne suffisent plus : aucun n’est libre de bout en bout. Les bribes de chemins candidats libres font alors office de RS permettant de calculer les RIS et ainsi de trouver un chemin par APF n’intersectant pas ces RIS. Pour ce faire, un autre véhicule à l’arrêt a été placé sur la voie opposée de manière à occuper le bout des chemins candidats lors du dépassement. Dans ce scénario, nous devrions obtenir un chemin qui slalome entre ces véhicules à l’arrêt.

Dans la figure 4.26, nous observons un départ similaire au scénario précédent. Nous pouvons remarquer le deuxième véhicule à l’arrêt sur la carte.

Dans la figure 4.27, nous constatons que le dépassement est bien planifié. En effet, le mode 3 est encore une fois sélectionné et le chemin final est visible sur la

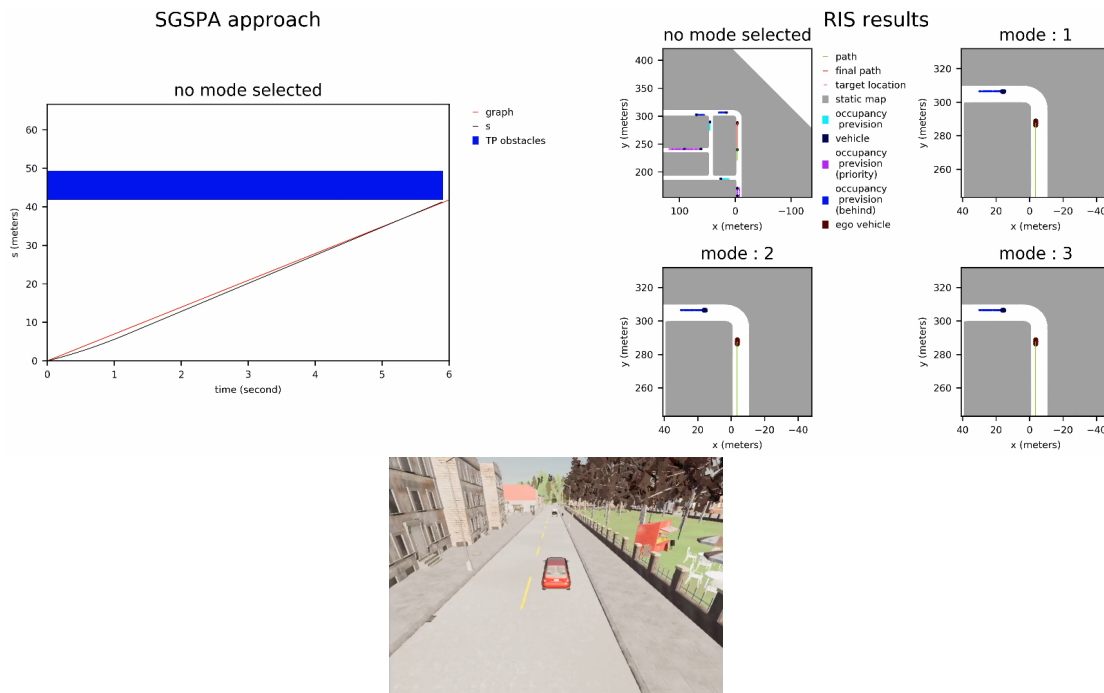


FIGURE 4.23 – Scénario avec un véhicule arrêté sur la voie. Début de la simulation.

vue globale. Ici, le résultat de la méthode est un peu différent que dans le scénario précédent. Nous voyons apparaître les RIS dessinés en marron dont certaines zones sont situées sur les voies. Chaque point des RS est aussi un point de la grille de calcul de l'APF. Allant du bleu au noir, les points de la grille de calcul les plus attractifs sont les plus clairs.

Dans la troisième figure 4.28, le chemin calculé grâce aux APF et évitant les RIS est bien réutilisé comme chemin de référence pour le suivi de chemin.

4.3.2.4 Quatrième scénario : un véhicule est arrêté sur la voie et un autre véhicule arrive en face

Ce quatrième scénario a pour objectif de montrer que la méthode fonctionne de manière similaire avec des véhicules en mouvement, créant des RIS équivalents aux scénarios précédents. Nous pouvons imaginer le dépassement d'un véhicule lent plutôt que d'un véhicule arrêté ou d'un véhicule arrivant en face plutôt que d'un véhicule arrêté sur la voie opposée.

Dans la figure 4.29, au début de la simulation, un véhicule est arrêté sur la voie de l'ego-véhicule. Cela le force à faire une manœuvre de dépassement similaire au dépassement observé dans le scénario de la section 4.3.2.2.

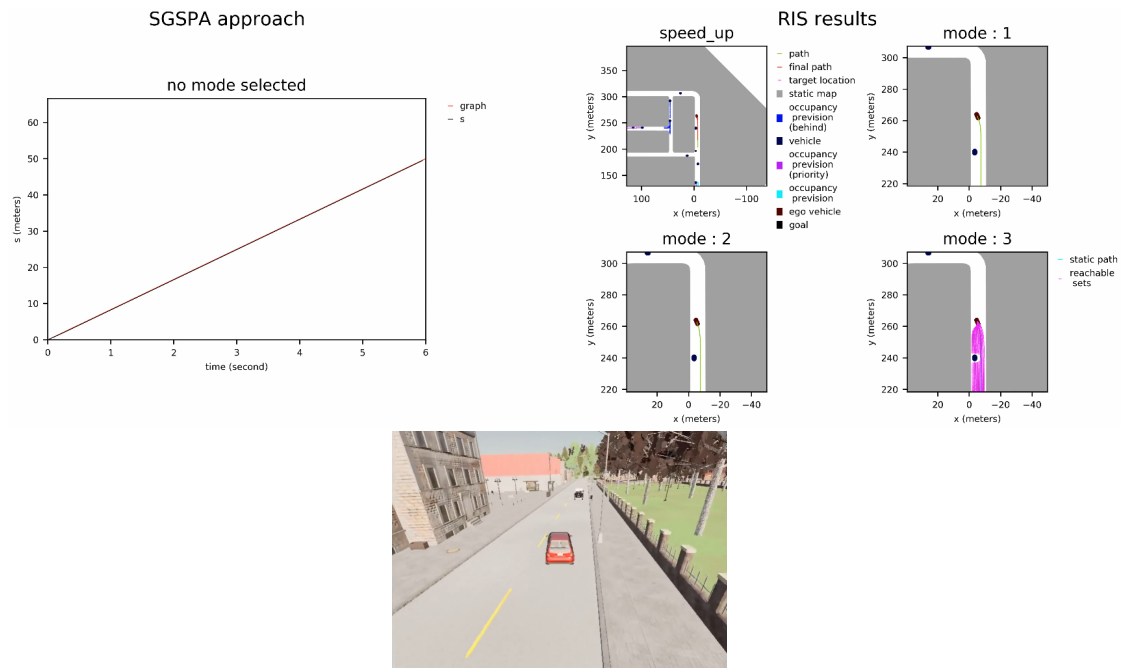


FIGURE 4.24 – Scénario avec un véhicule arrêté sur la voie. Début du dépassement.

Dans la seconde figure 4.30, nous voyons que le dépassement prévu est similaire à celui du scénario présenté dans la section 4.3.2.2.

Toutefois, l'apparition d'un véhicule arrivant en face force la méthode GPAD à utiliser les RIS pour calculer un nouveau chemin par APF afin de permettre à l'ego-véhicule de se rabattre car aucun des chemins candidats n'est utilisable. Nous pouvons voir ces résultats dans la troisième figure 4.31.

4.3.2.5 Cinquième scénario : un véhicule est arrêté sur la voie et un autre véhicule arrive trop rapidement en face

Ce dernier scénario est similaire au précédent sauf que le véhicule qui circule en face ne respecte pas les limitations de vitesse et arrive donc plus rapidement que prévu. L'objectif de ce scénario est de montrer que dans un cas d'urgence, l'approche GPAD peut quand même permettre de trouver une solution satisfaisante.

Dans la figure 4.32, un véhicule est encore arrêté sur la voie de l'ego-véhicule. Il est donc forcé à faire une manœuvre de dépassement similaire au scénario de la section 4.3.2.2.

Dans la figure 4.33, le dépassement prévu est similaire à celui observé dans le scénario présenté dans la section 4.3.2.2.

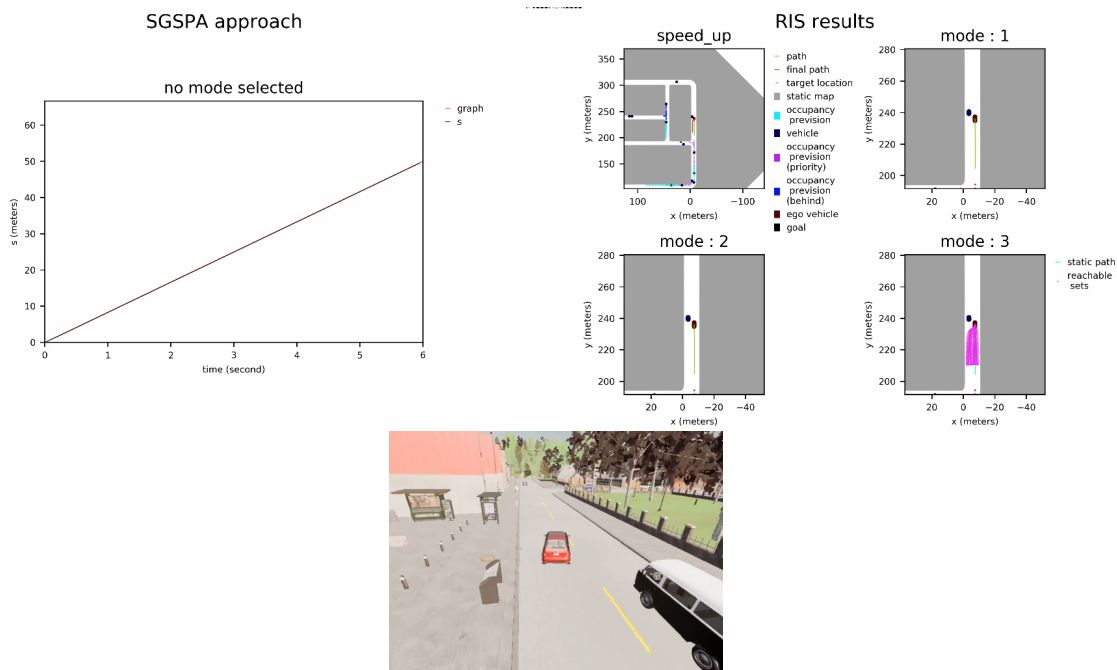


FIGURE 4.25 – Scénario avec un véhicule arrêté sur la voie. Fin du dépassement.

Par contre, la situation change dans la figure 4.34 : la méthode SGPA ne trouve pas de résultat pour le profil de vitesse. Le véhicule arrivant en face plus rapidement que prévu crée un TP obstacle sur tout le chemin. En effet, dans ce cas les prévisions ne permettent pas de trouver une issue sans collision. L’ego-véhicule doit donc effectuer une manœuvre. Cependant, la méthode MMRIS-APF ne donne pas de solution non plus. En effet, pour l’ensemble des modes, la vitesse reste trop élevée pour envisager de se rabattre à droite avec la décélération normale a_{max} . La méthode GPAD procède donc à un freinage d’urgence avec la décélération $a_{max,urgence}$.

Finalement, après le ralentissement de l’ego-véhicule à $a_{max,urgence}$, nous voyons dans la figure 4.35 que la solution de se rabattre est finalement envisageable avec le mode 2 qui est le mode à vitesse constante.

La décélération d’urgence permet de résoudre des situations jugées sans issue à partir des prédictions faites et des restrictions sur la décélération maximale.

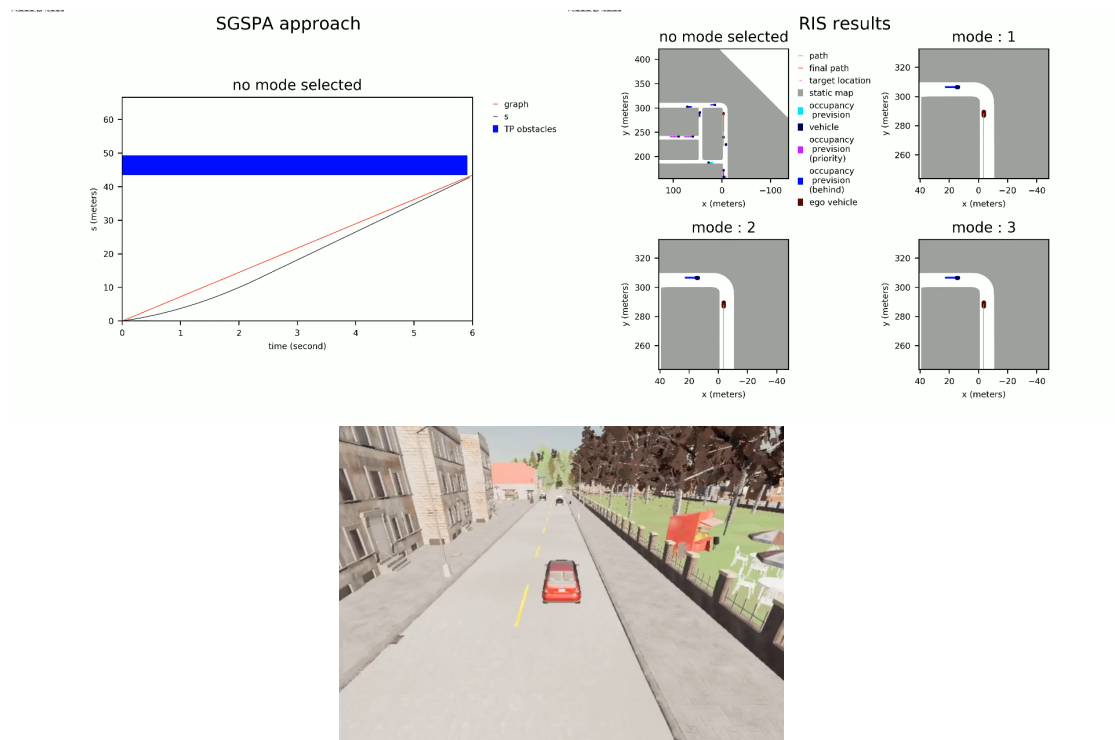


FIGURE 4.26 – Scénario avec deux véhicules arrêtés sur la voie. Début de la simulation.

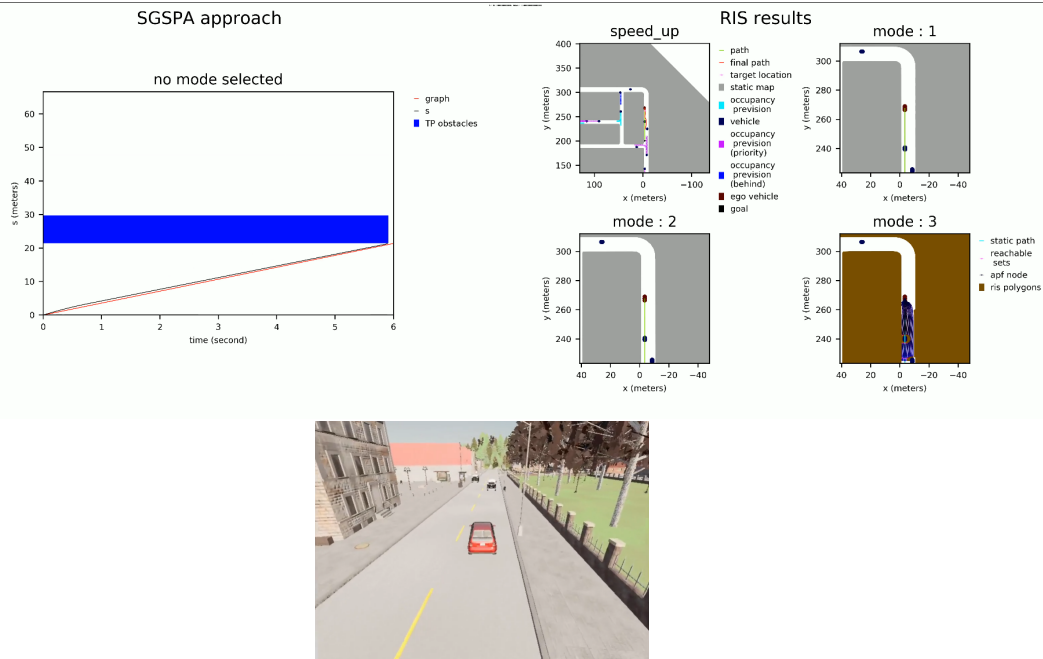


FIGURE 4.27 – Scénario avec deux véhicules arrêtés sur la voie. Début du dépassement.

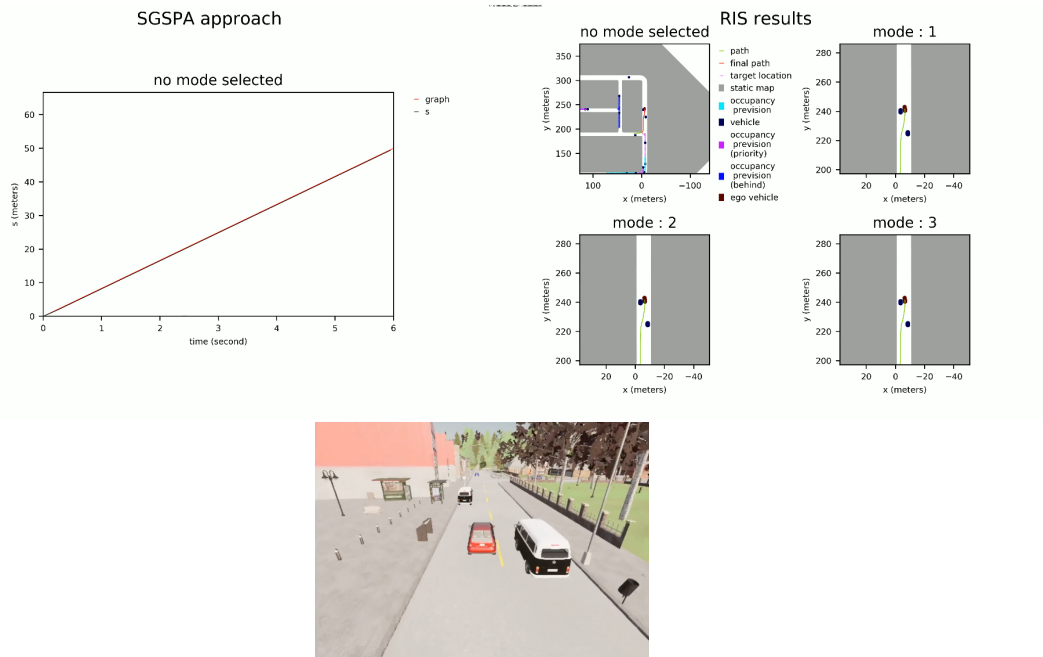


FIGURE 4.28 – Scénario avec deux véhicules arrêtés sur la voie. Fin du dépassement.

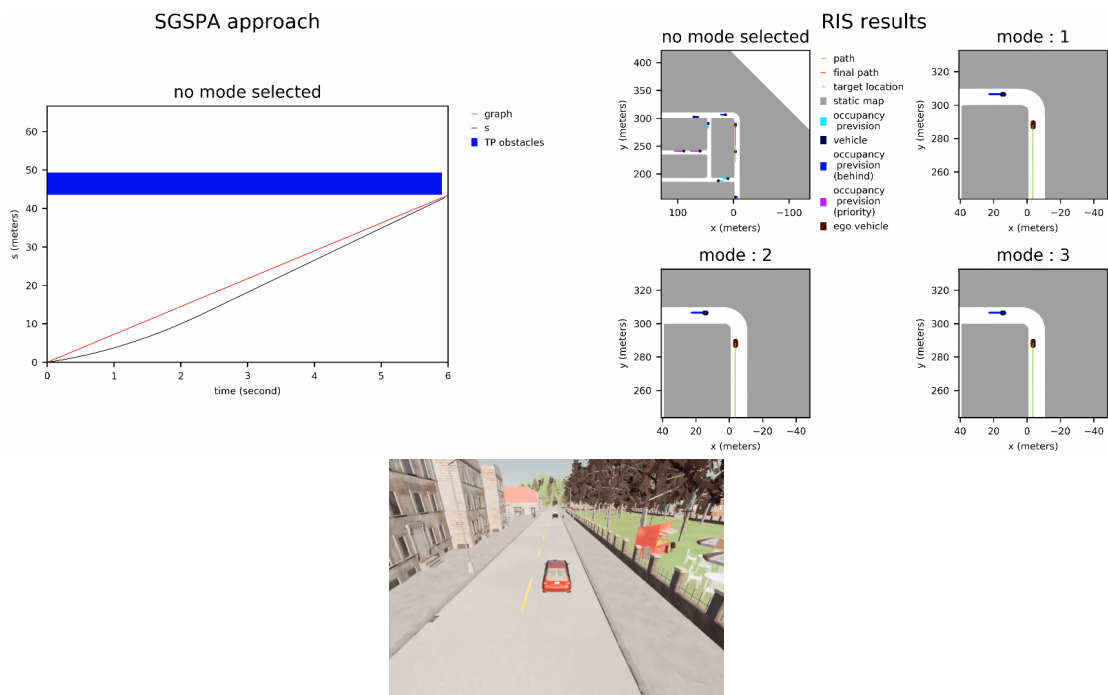


FIGURE 4.29 – Scénario avec un véhicule arrêté sur la voie et un véhicule arrivant en face.
Début de la simulation.

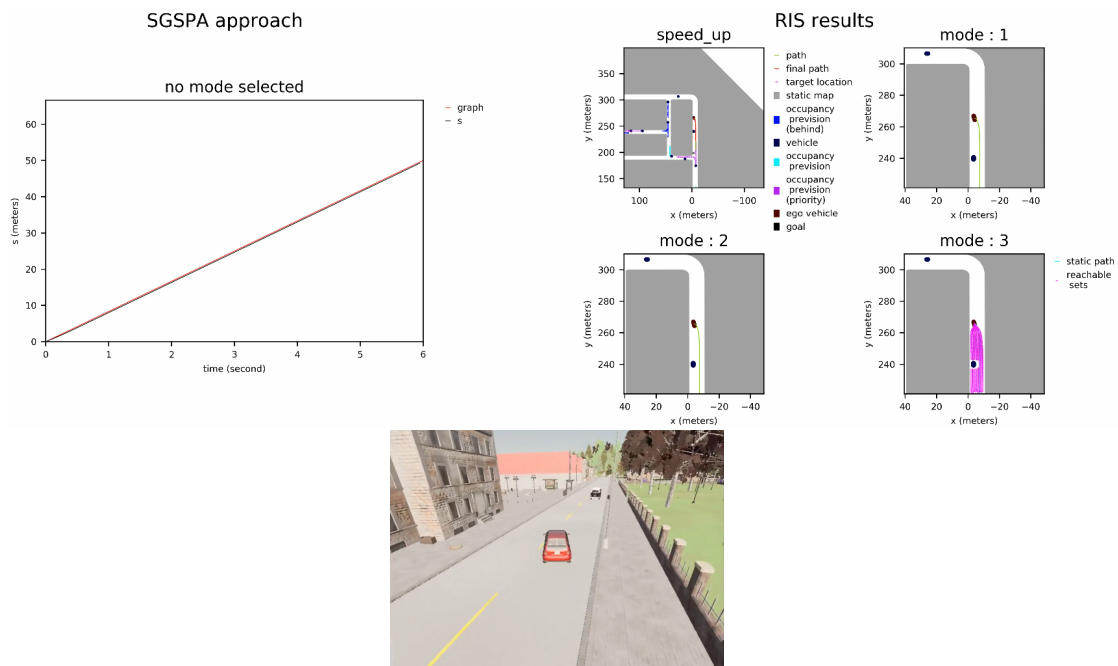


FIGURE 4.30 – Scénario avec un véhicule arrêté sur la voie et un véhicule arrivant en face.
Début du dépassement.

4. Résultats et simulation

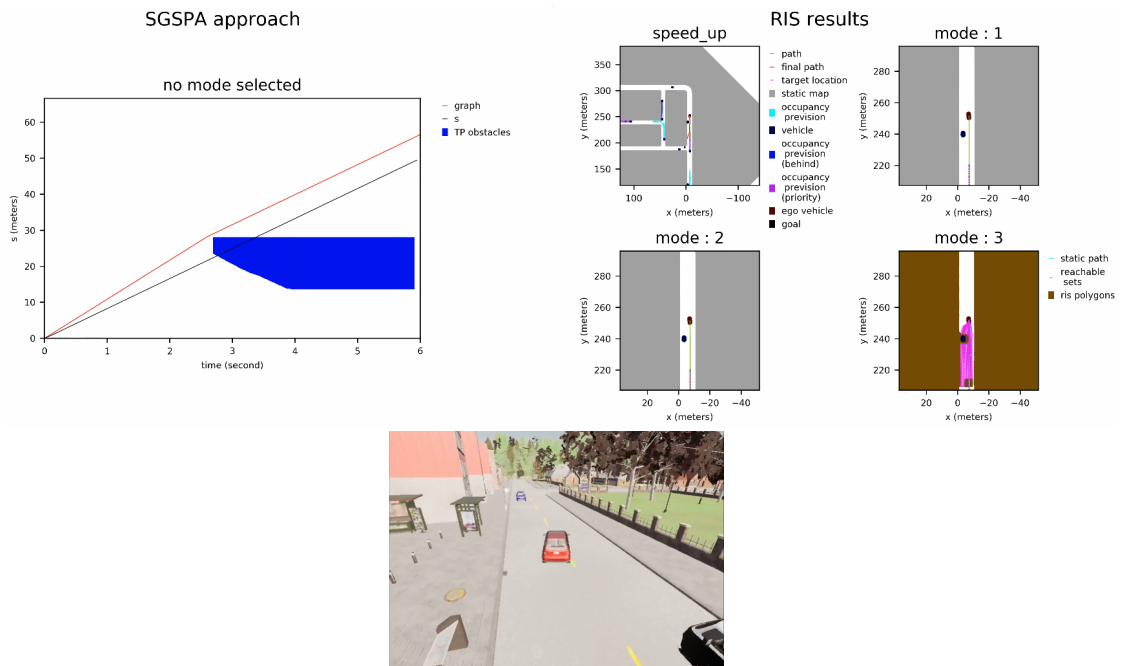


FIGURE 4.31 – Scénario avec un véhicule arrêté sur la voie et un véhicule arrivant en face.
Fin du dépassement.

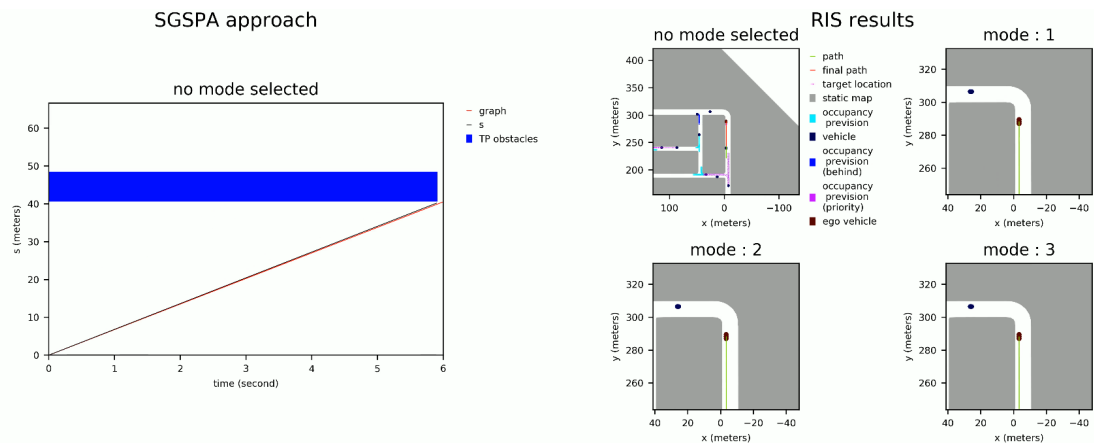


FIGURE 4.32 – Scénario avec un véhicule arrivant trop rapidement en face.
Début de la simulation.

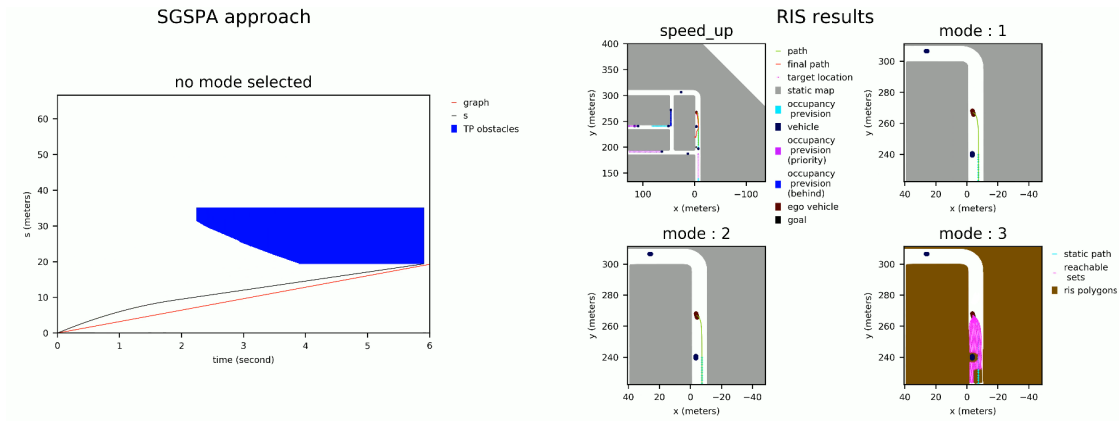


FIGURE 4.33 – Scénario avec un véhicule arrivant trop rapidement en face. Début du dépassement.

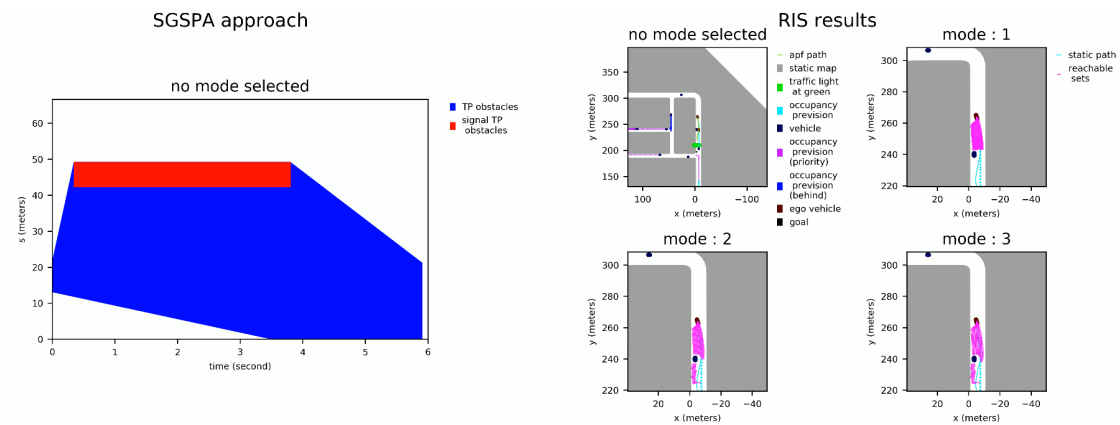


FIGURE 4.34 – Scénario avec un véhicule arrivant trop rapidement en face. Échec du dépassement et freinage d'urgence.

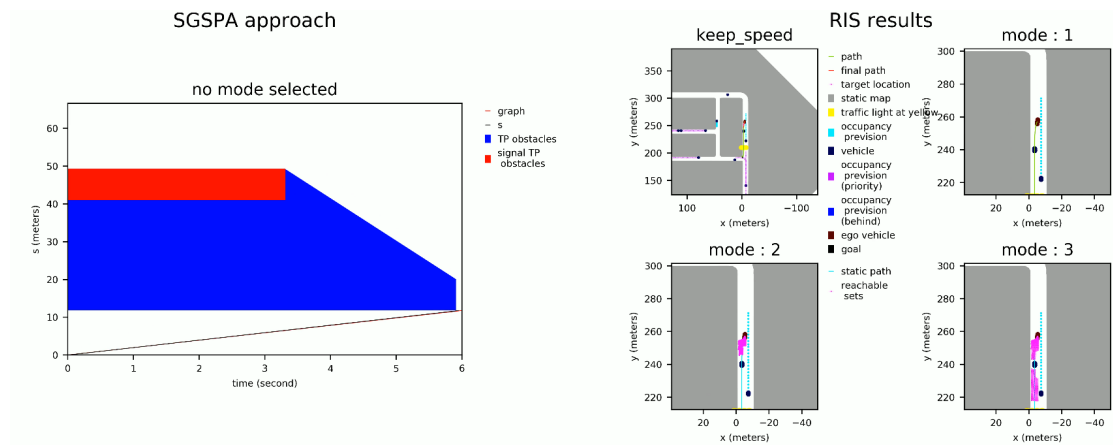


FIGURE 4.35 – Scénario avec un véhicule arrivant trop rapidement en face. Abandon du dépassement.

Chapitre 5

Conclusion et perspectives

Table des matières

5.1	Approche géométrique	137
5.2	Planification de chemin	137
5.3	Planification de vitesse	138
5.4	Planification en alternant chemin et vitesse	138
5.5	Extensions possibles des approches	138
5.5.1	Extensions possibles de la méthode SGSPA	139
5.5.2	Extensions possibles des méthodes basées sur les RIS	139
5.5.3	Extensions possibles de la méthode GPAD	140

5.1 Approche géométrique

Dans cette thèse, nous avons introduit, développé et évalué une méthode géométrique de planification en milieu urbain utilisant la décomposition chemin-vitesse. Cette méthode permet de passer facilement d'une planification d'un chemin à une planification de la vitesse. Les environnements urbains étant souvent encombrés, l'ego-véhicule devra souvent s'extraire de situations complexes et parfois imprévisibles du fait du non-respect du code de la route par les autres usagers. La possibilité de passer d'une planification de chemin à une planification de vitesse est donc indispensable pour la planification autonome en milieu urbain. Avant de combiner des approches de planification de vitesse et de chemin, il a été nécessaire de développer pour chacune d'entre elles des méthodes géométriques adaptées aux différentes contraintes et enjeux de la planification en milieu urbain.

5.2 Planification de chemin

Reprenant l'approche RIS [de [Beaucorps et al., 2018](#)] dédiée à la planification d'un chemin dans un environnement dynamique connaissant la vitesse de l'ego-véhicule, nous avons adapté la méthode aux cas :

- d'un véhicule non-holonyme circulant dans un environnement urbain en introduisant des contraintes liées au milieu urbain et au mouvement non holonyme dans le calcul des RS utilisés dans les RIS.
- d'obstacles dont le mouvement est incertain, représentés par des obstacles stochastiques. Ceci nous a permis de moduler la prévision de mouvement des autres véhicules sous forme de probabilités en les seuillant de manière dynamique.

Ces améliorations ont fait l'objet d'évaluations comparatives avec des approches de l'art. Il en ressort que notre approche rend la planification de chemin plus sûre et obtient des taux de collisions inférieurs aux méthodes existantes.

Nous avons ensuite proposé des extensions des RIS pour que la planification de chemin soit adaptée au milieu urbain.

D'abord, en utilisant des chemins de référence créés à partir des voies ou d'une méthode de planification ne considérant que les obstacles statiques, nous avons développé une méthode de chemins candidats qui permet d'identifier un chemin valide lorsque au moins un chemin candidat est libre de bout en bout et, à défaut, de calculer les RIS à partir de ces chemins candidats. Ensuite, nous avons développé la méthode MMRIS, extension de RIS, construisant des chemins pour différents modes de vitesse de l'ego-véhicule. Elle permet d'adapter à la fois le chemin de l'ego-véhicule et sa vitesse.

5.3 Planification de vitesse

Nous avons introduit et développé la méthode SGSPA de suivi de chemin pour planifier la vitesse sur un chemin donné au préalable. C'est une approche simple et rapide qui fournit un profil de vitesse peu conservateur. La méthode se caractérise par la prise en compte du risque induit par les zones occultées d'où des obstacles peuvent surgir et croiser la route de l'ego-véhicule. L'introduction d'obstacles virtuels pour modéliser ce risque a montré de bons résultats dans les simulations d'évaluations que nous avons réalisées. Nous avons pu tester différents scénarios dans un environnement urbain avec des configurations où le voisinage de l'ego-véhicule était encombré, notamment par des véhicules arrêtés sur la chaussée.

Par ailleurs, la méthode présente aussi la spécificité d'adopter un comportement proche de celui d'un conducteur humain. En effet, l'ego-véhicule maintient sa vitesse au maximum à l'approche de l'intersection, ralentit dans le cas où un véhicule surgirait puis accélère quand il observe que la voie est dégagée.

5.4 Planification en alternant chemin et vitesse

Pour finir, nous avons combiné ces deux approches adaptées au milieu urbain et y avons ajouté des fonctionnalités. Pour ce faire :

- Nous avons proposé une architecture combinant la planification de vitesse et la planification de chemin.
- Nous avons proposé une méthode MMRIS-APF basée sur des APF adaptés au milieu urbain et utilisant les RIS pour trouver un chemin.
- La méthode MMRIS présentant des possibilités limitées en termes de modes de vitesse (accélération, vitesse constante, décélération), nous l'avons complétée par la méthode de recherche de vitesse le long d'un chemin. La méthode SGSPA présente aussi l'avantage d'être moins coûteuse en temps de calcul que la méthode MMRIS-APF qui requiert un grand nombre de calculs d'intersections. Cela est particulièrement important puisqu'il s'agit de la méthode la plus utilisée par la méthode GPAD (bien que les deux méthodes puissent être utilisées à part égale, les scénarios de tests montrent que l'approche majoritairement utilisée est bien SGSPA).

Ces méthodes ont été validées sur des scénarios types de milieu urbain par simulation.

5.5 Extensions possibles des approches

Les différentes méthodes développées ont donc rempli les objectifs que nous nous étions fixés. Nous nous sommes efforcés de prendre en compte les aspects

spécifiques propres à la planification en milieu urbain. Il reste cependant des points à améliorer.

5.5.1 Extensions possibles de la méthode SGSPA

Un problème possible de la méthode SGSPA est qu'une visibilité trop réduite mène à une situation de blocage. Bien que les scénarios dans lesquels la visibilité était très réduite n'aient pas mené à de tels blocages, nous n'avons testé que des carrefours en T ou en X. Il peut exister des situations où la visibilité serait tellement réduite que le blocage devient inévitable. Notre proposition serait d'avancer malgré le risque à très faible vitesse pour améliorer la visibilité tout en ayant une distance de freinage la plus faible possible en cas de présence d'un véhicule. Il pourrait peut-être être possible de faire mieux en estimant la probabilité qu'un véhicule ne surgisse. Ceci constituerait une approche moins conservatrice face à ce problème.

De plus, la méthode de planification de vitesse a l'avantage de fonctionner avec les TP obstacles et d'être rapide. Cela est dû notamment au choix de lisser la vitesse obtenue par le graphe avec une saturation. Cette méthode simpliste ne permet pas de trouver une solution dans tous les cas et peut parfois causer des planifications omettant certaines solutions plus rapides. Une approche plus fine du passage entre la vitesse du graphe à la vitesse réalisable par l'ego-véhicule permettrait peut-être d'éviter de perdre ses solutions plus rapides sans trop ralentir le calcul de la méthode.

Une autre voie d'amélioration serait d'étendre les améliorations apportées aux RIS aux SGSPA et inversement. Nos apports majeurs à la méthode des RIS résident dans la prise en compte des contraintes des véhicules non holonomes et des obstacles stochastiques. Ces améliorations sont envisageables pour la méthode SGSPA. Avec cette méthode, nous obtenons la vitesse en appliquant des contraintes à la vitesse prévue par le graphe. Si cette vitesse sous contraintes entraîne une collision avec les TP obstacles, le graphe est révisé en conséquence. Il est possible d'utiliser n'importe quel type de contraintes pour mieux prendre en compte du comportement de l'ego-véhicule. Quant aux obstacles stochastiques, nous avons vu comment construire leur occupation dans l'espace-temps. Cette amélioration est donc directement transposable à la méthode SGSPA.

5.5.2 Extensions possibles des méthodes basées sur les RIS

L'apport majeur de la méthode SGSPA est la prise en compte du risque dû aux zones occultées grâce à l'utilisation de véhicules virtuels. Bien que nous sachions représenter ces risques sous la forme d'une occupation dans l'espace-temps, il nous semble difficile d'adapter l'utilisation de véhicules virtuels aux méthodes basées sur les RIS. Étant donné que les véhicules virtuels interviennent surtout pour traverser

des carrefours et que la méthode RIS n'est pas adaptée à cette situation (puisque un changement de chemin y est dangereux), utiliser des véhicules virtuels comme obstacle dans le cadre de la méthode RIS n'est pas pertinent.

Une piste plus intéressante serait plutôt de tenir compte de l'augmentation de la visibilité de l'ego-véhicule lors du calcul du chemin. Une première amélioration serait de favoriser les chemins candidats ayant une meilleure visibilité. On pourrait aussi déduire de la carte et des caractéristiques de l'ego-véhicule les zones où la visibilité est telle que l'approche SGSPA mènerait à une situation de blocage. Ces zones pourraient alors être considérées comme des obstacles fixes par la planification de chemin. Nous voyons alors qu'il est possible, là aussi, qu'une situation avec une visibilité trop réduite conduise à une situation de blocage.

Nous voyons aussi que les paramètres de la partie APF des MMRIS-APF pourraient être affinés pour obtenir des trajectoires qui soient plus précises. Une autre piste d'amélioration de la méthode porte sur la complexité des RIS qui est augmentée avec la méthode MMRIS (puisque le nombre de modes augmente d'autant la complexité). La méthode RIS étant coûteuse en calculs, il serait intéressant d'optimiser les calculs, notamment en explorant la possibilité d'une implémentation sur GPU. Actuellement, il arrive que la simulation ne tourne pas en temps réel et "attende" les résultats pour se rafraîchir.

5.5.3 Extensions possibles de la méthode GPAD

L'optimisation mentionnée plus haut serait nécessaire pour envisager une application réelle de la méthode hors simulation. À défaut, des approches trop lentes risqueraient de calculer une solution basée sur des données déjà trop anciennes. Il faudrait aussi améliorer l'architecture pour fournir un contrôle suffisamment sûr, même lorsque la méthode ne parvient pas à fournir une solution complète à temps.

Nous rappelons aussi que les approches développées dans cette thèse reposent sur l'hypothèse d'une prévision relativement juste des autres agents de l'environnement. Or, la recherche sur la prévision comportementale des agents ne permet pas encore de garantir une prévision parfaite. Par ailleurs, les méthodes utilisées pour effectuer ces prévisions sont en général très coûteuses en ressources. Quoi qu'il en soit, il faudrait toujours prioriser une communication entre les agents de la route lorsque cela est possible. Cela limiterait à la fois les incertitudes et le coût de calcul de la prévision.

Chapitre 6

Annexes

Table des matières

Glossaire	III
Notations	V
6.1 Implémentation de la méthode géométrique	VII
6.1.1 Calcul des RS	VII
6.1.1.1 RS discrets	VII
6.1.1.2 Occupation de l’ego-véhicule	VII
6.1.1.3 Simplification de l’occupation	VIII
6.1.2 Occupation des obstacles	X
6.1.2.1 Filtrage des obstacles	X
6.1.2.2 Cas des obstacles déterministes	X
6.1.2.3 Cas des obstacles stochastiques	X
6.1.3 Implémentation du calcul des RIS et des TP obstacles	XI
6.1.3.1 Implémentation du calcul d’intersection	XI
6.1.3.2 Interpolation des intersections	XI
6.2 Remplissage d’un rectangle par des disques	XIV
6.3 Implémentation d’un simulateur basé sur CARLA	XVI

Glossaire

A* Algorithme heuristique de recherche de chemin privilégiant la ligne droite dans sa recherche.

APF artificial potential field : méthode utilisant un champs de potentiel créé en associant une force répulsive aux obstacles et attractive au but.

DRT Dynamic Risk Tolerance : amélioration de l'approche SES [[Chiang et al., 2017b](#)].

Ego-véhicule Terme utilisé pour désigner le véhicule géré par l'approche.

Espace-temps Espace muni d'une dimension temporelle.

Filtre à particules Technique d'estimation de modèles fondées sur la simulation aussi connu sous le nom de méthode de Monte-Carlo séquentielle.

Graphe de visibilité Graphe d'emplacements intervisibles.

NP-difficile En informatique théorique, un problème NP-difficile est un problème vers lequel on peut ramener tout problème de la classe NP par une réduction polynomiale.

Obstacle virtuel Obstacle qui n'est pas réellement présent dans l'environnement et auquel des propriétés particulières peuvent être associées.

Optimal belief action utility Estimation de la meilleure action.

POMDP Partially Observable Markov Decision Process : modèle de décision Markovien avec des variables cachées.

RIS Reachable Interaction Sets : ensembles des zones d'interactions possibles avec d'autres véhicules.

RRT Algorithme d'exploration basé sur un graphe à croissance aléatoire.

RS Reachable Sets : ensemble des états atteignables par un véhicule.

SES Stochastic Ensemble Simulation : approche basée sur les RRT [[Chiang et al., 2015b](#)].

SR-sets Stochastic Reachable sets : RS avec intégration d'une incertitude sur la présence.

TP Time-Path : sous-espace de l'espace-temps.

VO Velocity Obstacles : méthode utilisant une représentation spécifique des obstacles en utilisant leur vitesse.

Véhicule holonome Véhicule ayant des contraintes mécaniques holonomes.

Véhicule non-holonome Véhicule ayant des contraintes mécaniques non-holonomes.

Véhicule virtuel Véhicule qui n'est pas réellement présent dans l'environnement et auquel des propriétés particulières peuvent être associées.

États Vecteur de variables définissant totalement l'objet à un instant donné.

Notations

\cdot projection orthogonale.

E plan de l'espace-temps. On associe la contrainte entre parenthèses. Par exemple, $E(t = 0)$ est le plan d'espace-temps tel que le temps vaut 0.

L distance entre les roues avant et les roues arrière.

$N_{\mathcal{P}}$ nombre de points p dans une ligne polygonale \mathcal{P} .

$N_{\mathcal{T}}$ nombre de points q dans une ligne polygonale \mathcal{T} .

N_{dyn} nombre d'obstacles dynamiques.

N_{mc} nombre d'itérations de la méthode de Monte-Carlo.

$N_{statique}$ nombre d'obstacles statiques.

P polygone.

X configuration du véhicule, appelée aussi état.

ΔT période de temps entre deux calculs des approches.

ι indice des instants de calcul des approches.

κ indice d'une vitesse parmi un ensemble de vitesses possibles.

λ variable de position d'un chemin dans un faisceau de chemins.

\mathcal{C} contraintes.

\mathcal{E} espace libre dans l'espace de configuration.

\mathcal{O} occupation d'un obstacle ou d'un véhicule. Elle peut être une collection de polygones ou de volumes.

\mathcal{P} chemin sous forme de ligne polygonale.

\mathcal{S} segment.

\mathcal{T} trajectoire sous forme de ligne polygonale.

ν^\perp vecteur unitaire perpendiculaire à ν .

ν vecteur unitaire tangent à un chemin.

ρ probabilité de présence.

- θ orientation d'un véhicule dans le plan (x, y) .
- a accélération.
- dt période de temps entre deux calculs du contrôle.
- d distance.
- i indice d'un point p dans un chemin \mathcal{P} .
- j indice d'un point q dans une ligne polygonale \mathcal{T} .
- k indice d'un obstacle statique P dans l'occupation des obstacles statiques $\mathcal{O}_{statique}$.
- l indice d'un obstacle dynamique.
- m indice de l'iteration de la méthode de Monte-Carlo.
- o indice des instants de calcul du contrôle.
- p point de l'espace.
- q point de l'espace-temps.
- r_c rayon de courbure.
- s abscisse curviligne.
- t_h horizon temporel.
- t_{gap} time-gap : période de temps à respecter entre le passage de deux véhicules en un point pour garantir la sécurité.
- t temps.
- u angle de braquage des roues.
- v vitesse.
- x variable d'espace.
- y variable d'espace.

6.1 Implémentation de la méthode géométrique

Nos méthodes de planification ont été implémentées puis évaluées dans différents environnements simulés. Nous allons commencer par présenter l'implémentation de l'approche géométrique, des calculs d'intersection dans l'espace-temps et des méthodes génériques nécessaires à l'ensemble des approches présentées dans le Chapitre 3. Les approches ont été implémentées en langage Python et les calculs géométriques 3D ont utilisé la librairie Python PyMesh [Zhou, 2020].

6.1.1 Calcul des RS

6.1.1.1 RS discrets

Comme nous l'avons vu dans la section 3.1, les RS se présentent comme des surfaces quelle que soit l'approche. Pour rappel, l'espace de configuration qui contient les RS est discrétisé. Le plus petit intervalle de temps dt utilisé dans les approches est celui entre deux calculs de mouvement de l'ego-véhicule t_o et t_{o+1} (cf. 3.22). C'est donc ce pas de temps qui est utilisé pour la discrétisation de l'espace de configuration, car aucun événement ne pourra être pris en compte par l'ego-véhicule avant la fin du pas de temps. Nous avons donc $o \in [0, N_{t_h}]$ où $N_{t_h} = \frac{t_h}{dt}$, en supposant que $\frac{t_h}{dt}$ est un entier. Suivant la méthode de création des RS, nous avons aussi un paramètre à parcourir pour représenter tous les possibles :

- $u \in [-u_{max}, u_{max}]$ pour une construction de RS à braquage constant (détaillée dans le paragraphe (a) dans la section 3.2.2.1).
- $\lambda \in [-\lambda_{max}, \lambda_{max}]$ pour une construction des RS par suivi de chemins candidats (détaillée dans le paragraphe (b) dans la section 3.2.3.1).

Ces paramètres sont eux aussi discrétisés pour permettre le calcul des RS. Nous découpons les intervalles de valeurs en N_{conf} valeurs.

Les RS sont donc une liste de N_{conf} configurations possibles pour chaque pas de temps t_o . La $\zeta^{\text{ème}}$ configuration du temps t_o sera notée $X_{ego,o,\zeta}$. Cette configuration est construite en suivant l'angle de braquage constant u_ζ ou en suivant le chemin $\mathcal{P}_{\lambda_{\zeta_0}}$ suivant la méthode de construction des RS choisie. En revanche, quelle que soit la méthode, nous avons :

$$RS(t_o) = \{X_{ego,o,\zeta} | \zeta \in [0, N_{conf}]\} \quad (6.1)$$

6.1.1.2 Occupation de l'ego-véhicule

Pour tenir compte de l'occupation de l'ego-véhicule parcourant les RS, nous calculons le polygone d'occupation $P_{ego}(X_{ego,o,\zeta})$ correspondant à chaque configuration $X_{ego,o,\zeta}$. Nous utilisons un rectangle pour la boîte englobante de l'ego-véhicule.

Nous notons $P_{ego,0}$ la boite englobante pour une configuration de référence $X_{ego,0}$ telle que $(x_{ego} = 0, y_{ego} = 0, \theta_{ego} = 0)$ qui a donc pour coordonnées :

$$P_{ego,0} = \left\{ \left[\frac{l_{ego}}{2}, \frac{L_{ego}}{2} \right], \left[-\frac{l_{ego}}{2}, \frac{L_{ego}}{2} \right], \left[-\frac{l_{ego}}{2}, -\frac{L_{ego}}{2} \right], \left[\frac{l_{ego}}{2}, -\frac{L_{ego}}{2} \right] \right\} \quad (6.2)$$

Pour obtenir $P_{ego}(X_{ego,o,\zeta})$ nous appliquons une translation et une rotation à la boite englobante.

$$P_{ego}(X_{ego,o,\zeta}) = P_{ego,0}R(\theta_{ego,o,\zeta}) + T(p_{ego,o,\zeta}) \quad (6.3)$$

avec R la matrice de rotation :

$$R(\theta_{ego,o,\zeta}) = \begin{bmatrix} \cos(\theta_{ego,o,\zeta}) & \sin(\theta_{ego,o,\zeta}) \\ -\sin(\theta_{ego,o,\zeta}) & \cos(\theta_{ego,o,\zeta}) \end{bmatrix} \quad (6.4)$$

et la translation :

$$T(p_{ego,o,\zeta}) = \begin{bmatrix} x_{ego,o,\zeta} \\ y_{ego,o,\zeta} \end{bmatrix} \quad (6.5)$$

6.1.1.3 Simplification de l'occupation

Pour accélérer les calculs d'intersection qui vont suivre, nous simplifions l'occupation de l'ego-véhicule. Pour cela, nous approximons chaque rectangle par une collection de disques. Nous approximons le rectangle par une collection $\mathcal{C}_{ego,0}$ de $N_{disques}$ disques identiques, le $\eta^{\text{ème}}$ disque étant $c_{ego,0,\eta}$ de centre $p_{ego,0,\eta}$ et de rayon r_{ego} . Il existe différentes stratégies pour couvrir un rectangle avec des disques identiques. Pour placer ces disques, nous utilisons un découpage en cellules du rectangle. Ensuite, nous prenons le cercle exinscrit de chaque cellule. Nous souhaitons que la boite englobante de l'ensemble des cercles soit la plus proche de la boite englobante de l'ego-véhicule. Nous verrons par la suite que l'augmentation de $N_{disques}$ augmente le temps de calcul des intersections de manière linéaire : ce calcul est une partie coûteuse de la méthode. Nous nous concentrerons donc sur un nombre restreint de disques. Nous différencions deux cas de figure, le cas où $N_{disques}$ est impair et le cas où il est pair. Si $N_{disques}$ est impair, le découpage choisi sera forcément horizontal. En revanche, quand $N_{disques}$ est pair, le découpage peut être vertical puis horizontal. Nous pouvons déterminer la meilleure méthode en fonction de la taille de la boite englobante et du nombre de disques. Pour la plupart des véhicules de taille classique, *i.e.*, environ deux fois plus longs que larges, il est plus intéressant d'utiliser uniquement le découpage horizontal. Avec ce découpage, l'approximation est plus grande pour la longueur que pour la largeur. Cela est moins gênant dans notre cas, car surévaluer la largeur de l'ego-véhicule peut mener à des situations de

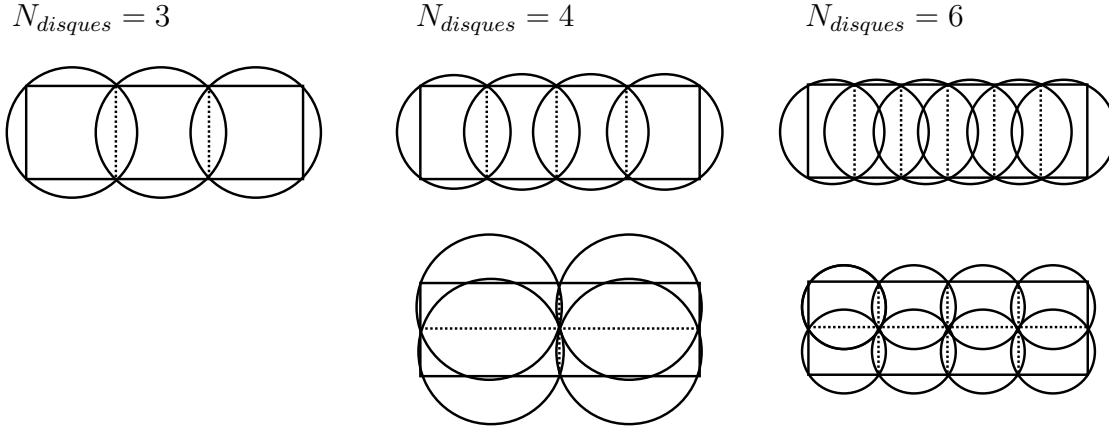


FIGURE 6.1 – Exemple de remplissage d'un rectangle par des disques.

blocage. Par la suite, nous ne considèrerons donc que le découpage horizontal (cf. figure 6.1 - voir l'annexe 6.2 pour le détail des calculs).

Pour calculer le centre des disques, nous calculons la longueur des cellules :

$$\delta = \frac{l_{ego}}{N_{disques}} \quad (6.6)$$

Nous en déduisons les centres du $\eta^{\text{ème}}$ disque (nous considérons $\eta \in [0, N_{disques} - 1]$) :

$$p_{ego,0,\eta} = \left[\delta \left(\eta - \frac{N_{disques} - 1}{2} \right), 0 \right] \quad (6.7)$$

et le rayon r_{ego} :

$$r_{ego} = \sqrt{\left(\frac{\delta}{2}\right)^2 + \left(\frac{L_{ego}}{2}\right)^2} \quad (6.8)$$

Nous avons aussi :

$$p_{ego,\eta}(X_{ego,o,\zeta}) = p_{ego,0,\eta}R(\theta_{ego,o,\zeta}) + T(p_{ego,o,\zeta}) \quad (6.9)$$

avec R la matrice de rotation :

$$R(\theta_{ego,o,\zeta}) = \begin{bmatrix} \cos(\theta_{ego,o,\zeta}) & \sin(\theta_{ego,o,\zeta}) \\ -\sin(\theta_{ego,o,\zeta}) & \cos(\theta_{ego,o,\zeta}) \end{bmatrix} \quad (6.10)$$

et la translation :

$$T(p_{ego,o,\zeta}) = \begin{bmatrix} x_{ego,o,\zeta} \\ y_{ego,o,\zeta} \end{bmatrix} \quad (6.11)$$

6.1.2 Occupation des obstacles

6.1.2.1 Filtrage des obstacles

Pour déterminer les N_{dyn} agents à considérer dans les calculs, il peut être nécessaire de filtrer les agents présents dans la scène. Nous avons déjà fait la distinction entre un véhicule complètement arrêté (alors considéré comme un obstacle statique) et un agent temporairement à l'arrêt qui pourrait repartir. Nous filtrons aussi les véhicules trop lointains pour être en interaction avec l'ego-véhicule durant l'horizon temporel. Si les agents respectent la vitesse maximale autorisée, les agents situés au-delà de la distance maximale de rencontre $d_{rencontre} = 2v_{max}t_h$ peuvent être ignorés.

6.1.2.2 Cas des obstacles déterministes

Dans le cas des obstacles déterministes, nous pouvons calculer leurs états futurs à partir de leur comportement prévu. Pour le $l^{\text{ème}}$ véhicule, parmi les N_{dyn} véhicules, nous connaissons donc son état $X_l(t_o)$. Nous en déduisons alors son occupation à l'instant t_o . Le polygone d'occupation de l'obstacle est un rectangle car l'occupation du véhicule déterministe est sa boîte englobante. Nous procédons de la même manière que pour l'ego-véhicule et nous trouvons $P_l(X_l(t_o))$ comme dans l'équation (6.3).

Dans ce cas, l'occupation peut aussi être approximée par des disques comme dans la section 6.1.1.3.

Nous notons ces disques $c_{l,\eta}(t_o)$ de centre $p_{l,\eta}(t_o)$ et de rayon r_l .

Pour ajouter une marge à ces occupations, nous pouvons augmenter le rayon des obstacles de la marge choisie. Les différentes marges seront détaillées en fonction des méthodes.

6.1.2.3 Cas des obstacles stochastiques

La simplification n'est pas possible dans le cas des obstacles stochastiques définis dans la section 3.2.2.2. Nous utiliserons dans ce cas des polygones pour l'occupation.

Pour appliquer une marge à ces occupations, nous pouvons appliquer aux polygones un grossissement (ou un *buffer*) de la valeur de la marge. Cela revient à faire la somme de Minkowski entre le polygone et un disque qui a pour diamètre la valeur de la marge.

Nous utiliserons la librairie Python Shapely [Gillies *et al.*, 2007] pour tous les calculs géométriques en deux dimensions.

6.1.3 Implémentation du calcul des RIS et des TP obstacles

6.1.3.1 Implémentation du calcul d'intersection

Le calcul d'intersection entre les RS de l'ego-véhicule et l'occupation des obstacles est nécessaire pour construire les TP obstacles ou les RIS. Pour ce faire, nous procédons pour chaque instant t_o à un calcul d'intersection entre les occupations des configurations possibles de l'ego-véhicule et l'occupation des obstacles. Si la simplification pour transformer les obstacles en collection de disques n'est pas faite, il faut calculer la distance du centre des disques $p_{ego,\eta}(X_{ego,o,\zeta})$ et les polygones des obstacles $P_l(X_l(t_o))$.

$$\begin{aligned} X_{ego,o,\zeta} \notin (RIS/TP_{obstacles}) &\iff \\ \forall \eta \in N_{disques}, \forall l \in N_{dyn}, & \\ dist(p_{ego,\eta}(X_{ego,o,\zeta}), P_l(X_l(t_o))) &> r_{ego} \end{aligned} \quad (6.12)$$

où $dist$ est la fonction qui renvoie la plus courte distance entre deux objets géométriques.

Si la simplification est aussi faite pour les obstacles, nous calculons alors la distance entre les centres des disques de l'ego-véhicule et le centre des disques de l'occupation des obstacles.

$$\begin{aligned} X_{ego,o,\zeta} \notin (RIS/TP_{obstacles}) &\iff \\ \forall (\eta_1, \eta_2) \in N_{disques}^2, \forall l \in N_{dyn}, & \\ dist(p_{ego,\eta_1}(X_{ego,o,\zeta}), p_{l,\eta_2}(t_o)) &> r_{ego} + r_l \end{aligned} \quad (6.13)$$

Cette méthode est beaucoup plus rapide car elle remplace le calcul de distance entre un polygone et un point par un calcul de distance entre points.

6.1.3.2 Interpolation des intersections

Le calcul d'intersection n'est cependant pas suffisant : la discrétisation du temps impose d'interpoler entre l'instant t_o et t_{o+1} pour reconstruire les RIS ou les TP obstacles.

Une méthode est d'interpoler les RS et l'occupation des obstacles en amont du calcul d'intersection. Toutefois, cela empêche les simplifications que nous avons vues précédemment et modifie la méthode pour le calcul des intersections. Cette implémentation est détaillée dans la section 4.1.1 et ne sera utilisée que dans la section 4.1.

Une autre façon de faire est d'interpoler a posteriori le résultat, mais en tenant compte de la position ζ des configurations concernées dans les RS et du temps t_o .

En effet, nous interpolerons de deux manières :

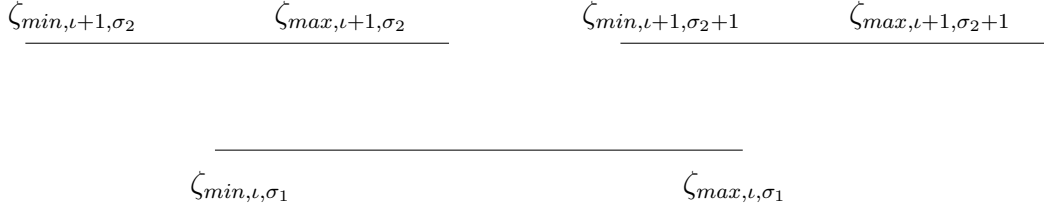


FIGURE 6.2 – Figure d’intersection des intervalles représentant deux instants consécutifs.

Les valeurs de l’intervalle du bas se recourent avec les deux intervalles du haut.

- entre les configurations successives d’un même paramètre de construction (λ_ζ ou u_ζ) dans le temps $X_{ego, o, \zeta}$ et $X_{ego, o+1, \zeta}$;
- et entre les configurations d’un même instant t_o ayant des paramètres de construction successifs $X_{ego, o, \zeta}$ et $X_{ego, o, \zeta+1}$.

Pour ce faire, nous procéderons de la façon suivante : à chaque instant t_o , nous regroupons par intervalle les configurations qui intersectent les obstacles. Nous obtenons $N_{int, o}$ intervalles de configuration. Le $\sigma^{\text{ème}}$ intervalle est caractérisé par ses valeurs extrêmes $\zeta_{min, o, \sigma}$ et $\zeta_{max, o, \sigma}$. Nous noterons les intervalles $I_{o, \sigma}$. Pour deux instants consécutifs t_o et t_{o+1} , si deux intervalles se recourent sur leurs valeurs extrêmes, nous construisons un polygone d’interpolation en prenant l’enveloppe convexe de l’ensemble de ses valeurs (cf. figures 6.2 et 6.3). L’interpolation faite ici permet de considérer que tout le polygone d’interpolation représente des configurations qui intersectent les obstacles sur la base des configurations discrètes qui intersectent les obstacles. Pour rappel, les TP obstacles sont contenus dans le sous-espace (s, t) et les RIS sont contenus dans le sous-espace (x, y) . Les configurations sont d’abord projetées dans le sous-espace correspondant pour supprimer une dimension. De plus, la dimension temporelle n’est plus une variable de la configuration : le temps étant discret, il suffit de connaître l’indice o de la couche temporelle à laquelle appartient la configuration. Nous pouvons écrire pour chaque couche o la relation donnant les polygones d’interpolation :

$$\begin{aligned}
 \exists(\sigma_1 \in N_{int, o}, \sigma_2 \in N_{int, o+1}) & | \zeta_{min, o, \sigma_1} \leq \zeta_{min, o+1, \sigma_2} \leq \zeta_{max, o, \sigma_1} \\
 & + \zeta_{min, o, \sigma_1} \leq \zeta_{max, o+1, \sigma_2} \leq \zeta_{max, o, \sigma_1} \quad (6.14) \\
 \Rightarrow \exists P_{o, \sigma_1, \sigma_2} & = conv(I_{o, \sigma_1} + I_{o+1, \sigma_2})
 \end{aligned}$$

où l’opérateur $+$ entre les conditions est un *ou* logique, l’opérateur $+$ entre les intervalles de configurations est la concaténation des éléments des intervalles en un ensemble de configurations et où *conv* est la fonction qui calcule l’enveloppe convexe d’un ensemble de valeurs.

Si un intervalle ne recoupe aucun autre intervalle, il est interpolé en polygone

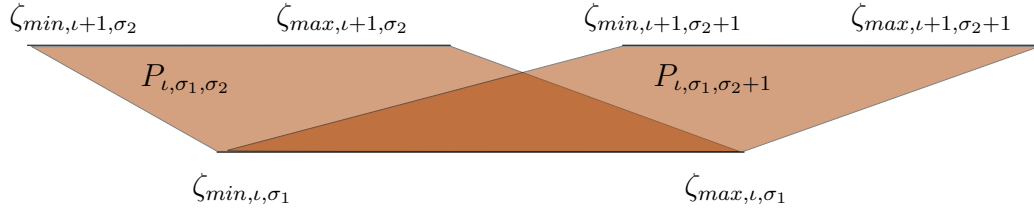


FIGURE 6.3 – Polygones d’interpolation représentant deux instants consécutifs. Les polygones sont construits en prenant l’enveloppe convexe des points dans les deux intervalles dont les valeurs se recourent.



FIGURE 6.4 – Figure sans intersection des intervalles.

en ne considérant que les valeurs de l’intervalle (cf. figure 6.4).

$$\begin{aligned}
 \forall (\sigma_1 \in N_{int, \sigma}, \sigma_2 \in N_{int, \sigma+1}) & | \zeta_{min, \sigma+1, \sigma_2} > \zeta_{max, \sigma, \sigma_1} \\
 & + \zeta_{max, \sigma+1, \sigma_2} < \zeta_{min, \sigma, \sigma_1} \\
 \Rightarrow \exists P_{\sigma, \sigma_1} & = conv(I_{\sigma, \sigma_1})
 \end{aligned} \tag{6.15}$$

En calculant les polygones d’interpolation pour chaque couche temporelle jusqu’à la couche $t_{h-1} = t_h - dt$, nous obtenons une collection de polygones d’interpolation.

Nous obtenons les TP obstacles ou les RIS en fusionnant tous les polygones d’interpolation.

$$(RIS/TP_{obstacles}) = \bigcup P_{\sigma, \sigma_1, \sigma_2} \cup \bigcup P_{\sigma, \sigma_1} \tag{6.16}$$

6.2 Comparaison des méthodes de remplissage d'un rectangle par des disques

Dans le cas où le nombre de disques est pair, nous considérons deux découpages du rectangle possibles : le découpage horizontal uniquement et un découpage vertical suivi d'un découpage horizontal. Dans le cas où le nombre de disques est impair, nous ne considérons que le découpage horizontal.

L'aire de la boîte englobante du découpage horizontal A_1 :

$$A_1 = 2r_{ego}((n-1)\delta + 2r_{ego}) \quad (6.17)$$

En remplaçant (6.6) et (6.8) dans l'équation (6.17), nous obtenons :

$$A_1 = \sqrt{L^2 + \frac{l^2}{n^2}} \left(l \frac{n-1}{n} + \sqrt{L^2 + \frac{l^2}{n^2}} \right) \quad (6.18)$$

Pour le découpage vertical puis horizontal, nous obtenons l'aire A_2

$$A_2 = \left(\frac{L}{2} + 2r'_{ego} \right) \left(\left(\frac{n}{2} - 1 \right) \delta' + 2r'_{ego} \right) \quad (6.19)$$

δ' vaut :

$$\delta' = \frac{2l}{n} \quad (6.20)$$

et le rayon r'_{ego} vaut :

$$r'_{ego} = \frac{1}{2} \sqrt{\left(\frac{L}{2} \right)^2 + \left(\frac{2l}{n} \right)^2} \quad (6.21)$$

En remplaçant cette fois (6.20) et (6.21) dans l'équation (6.19), nous obtenons :

$$A_2 = \left(\frac{L}{2} + \sqrt{\left(\frac{L}{2} \right)^2 + \left(\frac{2l}{n} \right)^2} \right) \left(l \frac{n-2}{n} + \sqrt{\left(\frac{L}{2} \right)^2 + \left(\frac{2l}{n} \right)^2} \right) \quad (6.22)$$

Nous pouvons alors calculer A_1 et A_2 en fonction des paramètres. L'aire la plus petite étant la meilleure, il est préférable de découper horizontalement quand $A_1 < A_2$. Dans le cas contraire, un découpage vertical est préférable.

Pour comparer les valeurs d'aires entre elles, nous pouvons aussi utiliser $A_0 = l_{ego}L_{ego}$, l'aire de la boîte englobante de l'ego-véhicule (comme dans le tableau 6.1).

Nous remarquons alors que le découpage vertical est intéressant uniquement quand le véhicule est très large ou que l'on utilise de nombreux cercles.

		A_1	$\frac{A_1}{A_0}$	A_2	$\frac{A_2}{A_0}$
$L = 1, l = 1.5$	n = 2	2.50	1.67	3.29	2.19
	n = 3	2.36	1.57		
	n = 4	2.34	1.56	2.31	1.54
	n = 5	2.34	1.56		
	n = 6	2.35	1.57	2.06	1.37
	n = 7	2.36	1.57		
	n = 8	2.37	1.58	1.97	1.31
$L = 1, l = 2.0$	n = 2	3.41	1.71	5.28	2.64
	n = 3	3.05	1.53		
	n = 4	2.93	1.47	3.42	1.71
	n = 5	2.88	1.44		
	n = 6	2.87	1.44	2.89	1.45
	n = 7	2.86	1.43		
	n = 8	2.87	1.44	2.66	1.33
$L = 1, l = 2.5$	n = 2	4.56	1.82	7.77	3.11
	n = 3	3.86	1.54		
	n = 4	3.60	1.44	4.79	1.92
	n = 5	3.49	1.40		
	n = 6	3.43	1.37	3.88	1.55
	n = 7	3.40	1.36		
	n = 8	3.39	1.36	3.48	1.39

TABLEAU 6.1 – Tableau des aires en fonction des paramètres.

6.3 Implémentation d'un simulateur basé sur CARLA

Pour tester les problématiques liées aux contraintes du milieu urbain dans les approches SGSPA (section 3.3) et MMRIS (section 3.4), nous avons développé un environnement de test plus complet.

Pour ce faire, nous avons développé nos approches sur le simulateur automobile CARLA [Dosovitskiy *et al.*, 2017] utilisant un moteur physique pour la simulation : l'Unreal Engine [Epic Games, 2020].

Le simulateur CARLA est open source et a été choisi pour ses possibilités d'adaptations. Il est initialement peu adapté à la simulation de méthodes de navigation mais il est possible de contrôler l'ego-véhicule parmi des agents contrôlés par un gestionnaire de trafic intégré à CARLA.

Les adaptations que nous avons réalisées sont disponibles au https://gitlab.inria.fr/rponcele/carla_pythonapi. Elles comportent une API qui complète celle de CARLA, une carte supplémentaire et les données nécessaires au déroulement des scénarios.

Bibliographie

- [Ajanovic *et al.*, 2018] AJANOVIC, Z., LACEVIC, B., SHYROKAU, B., STOLZ, M. et HORN, M. (2018). Search-based optimal motion planning for automated driving. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4523–4530. <http://arxiv.org/abs/1803.04868>. 35, 39, 88
- [Bender *et al.*, 2015] BENDER, P., TAŞ, Ömer Şahin., ZIEGLER, J. et STILLER, C. (2015). The combinatorial aspect of motion planning : Maneuver variants in structured environments. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 1386–1392. <https://doi.org/10.1109/IVS.2015.7225909>. 29, 38
- [Bouton *et al.*, 2018] BOUTON, M., NAKHAEI, A., FUJIMURA, K. et KOCHENDERFER, M. J. (2018). Scalable decision making with sensor occlusions for autonomous driving. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2076–2081. <https://doi.org/10.1109/ICRA.2018.8460914>. 35, 39
- [Braess, 1968] BRAESS, D. (1968). Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung*, 12(1):258–268. <https://doi.org/10.1007/BF01918335>. 4
- [Canny et Reif, 1987] CANNY, J. et REIF, J. (1987). New lower bound techniques for robot motion planning problems. *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, pages 49–60. <http://ieeexplore.ieee.org/document/4568255/>. 52
- [Chakravarthy et Ghose, 1998] CHAKRAVARTHY, A. et GHOSE, D. (1998). Obstacle avoidance in a dynamic environment : A collision cone approach. *IEEE Transactions on Systems, Man, and Cybernetics-Part A : Systems and Humans*, 28(5):562–574. <https://doi.org/10.1109/3468.709600>. 28, 38
- [Chiang *et al.*, 2015a] CHIANG, H.-T., MALONE, N., LESSER, K., OISHI, M. et TAPIA, L. (2015a). Path-guided artificial potential fields with stochastic reachable sets for motion planning in highly dynamic environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2347–2354. <https://doi.org/10.1109/ICRA.2015.7139511>. 31, 38, 104

- [Chiang *et al.*, 2015b] CHIANG, H.-T., RACKLEY, N. et TAPIA, L. (2015b). Stochastic ensemble simulation motion planning in stochastic dynamic environments. *In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3836–3843. <https://doi.org/10.1109/IROS.2015.7353916>. 32, 62, 65, 104, IV
- [Chiang *et al.*, 2017a] CHIANG, H.-T. L., HOMCHAUDHURI, B., SMITH, L. et TAPIA, L. (dec. 2017a). Safety, challenges, and performance of motion planners in dynamic environments. *In 2017 International Symposium on Robotics Research (ISRR), Puerto Varas, Chile*. https://doi.org/10.1007/978-3-030-28619-4_55. 100, 102, 104
- [Chiang *et al.*, 2017b] CHIANG, H.-T. L., HOMCHAUDHURI, B., VINOD, A. P., OISHI, M. et TAPIA, L. (2017b). Dynamic risk tolerance : Motion planning by balancing short-term and long-term stochastic dynamic predictions. *In IEEE International Conference on Robotics and Automation (ICRA)*, pages 3762–3769. <https://doi.org/10.1109/ICRA.2017.7989434>. 32, 38, 63, 104, 106, III
- [Chung *et al.*, 2009] CHUNG, W., KIM, S., CHOI, M., CHOI, J., KIM, H., MOON, C.-b. et SONG, J.-B. (2009). Safe navigation of a mobile robot considering visibility of environment. *IEEE Transactions on Industrial Electronics*, 56(10):3941–3950. <https://doi.org/10.1109/TIE.2009.2025293>. 33, 39
- [Claussmann *et al.*, 2020] CLAUSSMANN, L., REVILLOUD, M., GRUYER, D. et GLASER, S. (2020). A review of motion planning for highway autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 21(5):1826–1848. 86
- [COM 789 final, 2020] COM 789 final (2020). COMMUNICATION DE LA COMMISSION AU PARLEMENT EUROPÉEN, AU CONSEIL, AU COMITÉ ÉCONOMIQUE ET SOCIAL EUROPÉEN ET AU COMITÉ DES RÉGIONS - Stratégie de mobilité durable et intelligente – mettre les transports européens sur la voie de l’avenir. <https://eur-lex.europa.eu/legal-content/FR/TXT/?uri=CELEX:52020DC0789>. 4
- [Comité d’ORAD, 2021] Comité d’ORAD (2021). *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. On-Road Automated Driving (ORAD) committee. https://doi.org/10.4271/J3016_202104. 3
- [Damerow *et al.*, 2017] DAMEROW, F., PUPHAL, T., LI, Y. et EGGERT, J. (2017). Risk-based driver assistance for approaching intersections of limited visibility. *In IEEE International Conference on Vehicular Electronics and Safety, (ICVES)*, pages 178–184. IEEE. <https://doi.org/10.1109/ICVES.2017.7991922>. 34, 39, 82

-
- [de Beaucorps, 2019] de BEAUCORPS, P. (2019). *Planification de trajectoire dans un environnement peu contraint et fortement dynamique*. Thèse de doctorat, Sorbonne University, France. 98, 104, 105
- [de Beaucorps et al., 2018] de BEAUCORPS, P., VERROUST-BLONDET, A., PONCELET, R. et NASHASHIBI, F. (2018). RIS : A framework for motion planning among highly dynamic obstacles. *In 15th International Conference on Control, Automation, Robotics and Vision, (ICARCV)*, pages 1387–1393. <https://hal.inria.fr/hal-01903318>. 5, 32, 38, 55, 57, 88, 98, 100, 105, 112, 137
- [Dijkstra, 1959] DIJKSTRA, E. W. (1959). A note on two problems in connexion with graphs. *Numer. Math.*, 271:269–271. <https://doi.org/10.1007/BF01386390>. 21
- [Dosovitskiy et al., 2017] DOSOVITSKIY, A., ROS, G., CODEVILLA, F., LÓPEZ, A. et KOLTUN, V. (2017). CARLA : an open urban driving simulator. *In 1st Annual Conference on Robot Learning, CoRL 2017, Mountain View, California, USA, November 13-15, 2017, Proceedings*, volume 78 de *Proceedings of Machine Learning Research*, pages 1–16. PMLR. <http://proceedings.mlr.press/v78/dosovitskiy17a.html>. XVI
- [Dubins, 1957] DUBINS, L. E. (1957). On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents. *Source : American Journal of Mathematics*, 79(3):497–516. <https://doi.org/10.2307/2372560>. 32
- [Epic Games, 2020] EPIC GAMES (2020). Unreal engine. <https://www.unrealengine.com>. XVI
- [Erdmann et Lozano-Pérez, 1987] ERDMANN, M. et LOZANO-PÉREZ, T. (1987). *On multiple moving objects*, volume 2. Springer. <https://doi.org/10.1109/ROBOT.1986.1087401>. 28, 37, 41
- [Fedotov et al., 2011] FEDOTOV, A., PATSKO, V. et TUROVA, V. (2011). Reachable sets for simple models of car motion. *Recent Advances in Mobile Robotics*. <https://doi.org/10.5772/26278>. 32, 38, 59
- [Fiorini et Shiller, 1993] FIORINI, P. et SHILLER, Z. (1993). Motion planning in dynamic environments using the relative velocity paradigm. *Proceedings IEEE International Conference on Robotics and Automation*, pages 560–565 vol.1. <https://doi.org/10.1109/ROBOT.1993.292038>. 28, 38
- [Fiorini et Shiller, 1998] FIORINI, P. et SHILLER, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research*, 17(7):760–772. <https://doi.org/10.1177/027836499801700706>. 28, 30, 38

- [Fox *et al.*, 1997] FOX, D., BURGARD, W. et THRUN, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33. <https://doi.org/10.1109/100.580977>. 28, 29, 38
- [Fujimura et Samet, 1990] FUJIMURA, K. et SAMET, H. (1990). Motion planning in a dynamic domain. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 324–330. <https://doi.org/10.1109/ROBOT.1990.125996>. 28, 37
- [Ge et Cui, 2002] GE, S. S. et CUI, Y. J. (2002). Dynamic motion planning for mobile robots using potential field method. *Autonomous robots*, 13(3):207–222. <https://doi.org/10.1023/A:1020564024509>. 27, 37, 38
- [Gerdes et Rossetter, 2001] GERDES, J. C. et ROSSETTER, E. J. (2001). A Unified Approach to Driver Assistance Systems Based on Artificial Potential Fields. *Journal of Dynamic Systems, Measurement, and Control*, 123(3):431–438. <https://doi.org/10.1115/1.1386788>. 35, 39
- [Gillies *et al.*, 2007] GILLIES, S. *et al.* (2007). Shapely : manipulation and analysis of geometric objects. <https://github.com/Toblerity/Shapely>. X
- [Gomes *et al.*, 2012] GOMES, J., VELHO, L. et SOUSA, M. C. (2012). *Computer graphics : Theory and practice*. A K Peters/CRC Press. <https://doi.org/10.1201/b11857>. 48
- [González *et al.*, 2016] GONZÁLEZ, D., PÉREZ, J., MILANÉS, V. et NASHASHIBI, F. (2016). A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):1135–1145. <https://doi.org/10.1109/TITS.2015.2498841>. 19, 86
- [Hart *et al.*, 1972] HART, P. E., NILSSON, N. J. et RAPHAEL, B. (1972). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *ACM SIGART Bulletin*, 4(37):28–29. <http://ieeexplore.ieee.org/document/4082128/> <http://portal.acm.org/citation.cfm?doid=1056777.1056779>. 24, 36
- [Hilgert *et al.*, 2003] HILGERT, J., HIRSCH, K., BERTRAM, T. et HILLER, M. (2003). Emergency path planning for autonomous vehicles using elastic band theory. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, 2:1390–1395. <https://doi.org/10.1109/AIM.2003.1225546>. 36, 39
- [Hoermann *et al.*, 2017] HOERMANN, S., KUNZ, F., NUSS, D., RENTER, S. et DIETMAYER, K. (2017). Entering crossroads with blind corners. A safe strategy for autonomous vehicles. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 727–732. <https://doi.org/10.1109/IVS.2017.7995803>. 34, 39, 82
- [Hoy *et al.*, 2015] HOY, M., MATVEEV, A. S. et SAVKIN, A. V. (2015). Algorithms for collision-free navigation of mobile robots in complex cluttered environments : a survey. *Robotica*, 33(3):463–497. <https://doi.org/10.1017/S0263574714000289>. 19

- [Huang et Chung, 2004] HUANG, H.-P. et CHUNG, S.-Y. (2004). Dynamic visibility graph for path planning. In *IEEE Intelligent Robots and Systems (IROS)*, pages 2813–2818. <https://doi.org/10.1109/IROS.2004.1389835>. 31, 36, 38, 39
- [Huang, 2009] HUANG, L. (2009). Velocity planning for a mobile robot to track a moving target—a potential field approach. *Robotics and Autonomous Systems*, 57(1):55–63. <https://doi.org/10.1016/j.robot.2008.02.005>. 27, 28, 37, 38
- [Hubmann et al., 2016] HUBMANN, C., AEBERHARD, M. et STILLER, C. (2016). A generic driving strategy for urban environments. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 1010–1016. <https://doi.org/10.1109/ITSC.2016.7795679>. 33, 38
- [Hubmann et al., 2019] HUBMANN, C., QUETSCHLICH, N., SCHULZ, J., BERNHARD, J., ALTHOFF, D. et STILLER, C. (2019). A POMDP Maneuver Planner For Occlusions in Urban Scenarios. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 1909–16. <https://doi.org/10.1109/IVS.2019.8814179>. 35, 39
- [Johnson et Hauser, 2012] JOHNSON, J. et HAUSER, K. (2012). Optimal acceleration-bounded trajectory planning in dynamic environments along a specified path. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2035–2041. <https://doi.org/10.1109/ICRA.2012.6225233>. 31, 38, 56
- [Johnson et Hauser, 2013] JOHNSON, J. et HAUSER, K. (2013). Optimal longitudinal control planning with moving obstacles. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 605–611. <https://doi.org/10.1109/IVS.2013.6629533>. 31, 38
- [Kant et Zucker, 1986] KANT, K. et ZUCKER, S. W. (1986). Toward Efficient Trajectory Planning : The Path-Velocity Decomposition. *The International Journal of Robotics Research*, 5(3):72–89. <https://doi.org/10.1177/027836498600500304>. 27, 31, 38, 52, 77, 78, 81
- [Karaman et Frazzoli, 2011] KARAMAN, S. et FRAZZOLI, E. (2011). Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research*, 30(7):846–894. <https://doi.org/10.48550/arXiv.1105.1186>. 32, 38
- [Karanam et al., 2022] KARANAM, S. K., DUHAUTBOUT, T., TALJ, R., CHERFAOUI, V., AOUN, F. et GUILLEMARD, F. (2022). Virtual Obstacle for a Safe and Comfortable Approach to Limited Visibility Situations in Urban Autonomous Driving. *IEEE Intelligent Vehicles Symposium (IV 2022)*. <https://doi.org/10.1109/IV51971.2022.9827372>. 34, 39, 82
- [Keller et al., 2014] KELLER, M., HOFFMANN, F., BERTRAM, T., HASS, C. et SEEWALD, A. (2014). Planning of Optimal Collision Avoidance Trajectories

- with Timed Elastic Bands. *IFAC Proceedings Volumes*, 47(3):9822–9827. <https://doi.org/10.3182/20140824-6-ZA-1003.01143>. 36, 39
- [Khatib, 1985] KHATIB, O. (1985). Real-time obstacle avoidance for manipulators and mobile robots. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 500–505. <https://doi.org/10.1109/ROBOT.1985.1087247>. 25, 27, 36
- [Ko et Lee, 1996] KO, N. Y. et LEE, B. H. (1996). Avoidability measure in moving obstacle avoidance problem and its use for robot motion planning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 1296–1303. <https://doi.org/10.1109/IROS.1996.568984>. 27, 37, 38, 104
- [Koren et Borenstein, 1991] KOREN, Y. et BORENSTEIN, J. (1991). Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 1398–1404 vol.2. <https://doi.org/10.1109/ROBOT.1991.131810>. 88, 92
- [Koutsoupias et Papadimitriou, 2009] KOUTSOPIAS, E. et PAPADIMITRIOU, C. (2009). Worst-case equilibria. *Computer Science Review*, 3(2):65–69. <https://www.sciencedirect.com/science/article/pii/S1574013709000203>. 4
- [Krüger, 2019] KRÜGER, T.-J. (2019). Graph-Based Speed Planning for Autonomous Driving. Mémoire de D.E.A., Freie Universität Berlin. https://www.mi.fu-berlin.de/inf/groups/ag-ki/Theses/Completed-theses/Master_Diploma-theses/2019/Krueger/MA_Krueger.pdf. 31, 38
- [Latombe, 1991] LATOMBE, J. (1991). *Robot motion planning*, volume 124 de *The Kluwer international series in engineering and computer science*. Kluwer. <https://doi.org/10.1007/978-1-4615-4022-9>. 19, 23, 25, 26, 27, 36, 47, 104
- [LaValle, 1998] LAVALLE, S. M. (1998). Rapidly-exploring random trees : A new tool for path planning. *The annual research report*. <http://msl.cs.uiuc.edu/~lavalle/papers/Lav98c.pdf>. 24, 26, 31, 36, 38
- [LaValle, 2006] LAVALLE, S. M. (2006). *Planning Algorithms*. Cambridge University Press. <https://doi.org/10.1017/CB09780511546877>. 13, 19, 24, 32, 38
- [Lin et al., 2019] LIN, X., ZHANG, J., SHANG, J., WANG, Y., YU, H. et ZHANG, X. (2019). Decision making through occluded intersections for autonomous driving. In *IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 2449–2455. <https://doi.org/10.1109/ITSC.2019.8917348>. 35, 39, 120
- [Lozano-Pérez, 1983] LOZANO-PÉREZ, T. (1983). Spatial planning : A configuration space approach. *IEEE Transactions on Computers*, C-32(2):108–120. <https://doi.org/10.1109/TC.1983.1676196>. 48

- [Lozano-Pérez et Wesley, 1979] LOZANO-PÉREZ, T. et WESLEY, M. A. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Commun. ACM*, 22(10). <https://doi.org/10.1145/359156.359164>. 12, 14, 19, 20, 48
- [Lozano-Pérez et Wesley, 1979] LOZANO-PÉREZ, T. et WESLEY, M. A. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10):560–570. <https://doi.org/10.1145/359156.359164>. 105
- [Malone et al., 2017] MALONE, N., CHIANG, H.-T., LESSER, K., OISHI, M. et TAPIA, L. (2017). Hybrid dynamic moving obstacle avoidance using a stochastic reachable set-based potential field. *IEEE Transactions on Robotics*, 33(5):1124–1138. <https://doi.org/10.1109/TR0.2017.2705034>. 32, 38
- [Metropolis et Ulam, 1949] METROPOLIS, N. et ULAM, S. M. (1949). The monte carlo method. *Journal of the American Statistical Association*, 44 247:335–41. <https://doi.org/10.1080/01621459.1949.10483310>. 24
- [Minguez et al., 2008] MINGUEZ, J., LAMIRAUX, F. et LAUMOND, J.-P. (2008). Motion planning and obstacle avoidance. In *Springer handbook of robotics*, pages 827–852. Springer. https://doi.org/10.1007/978-3-319-32552-1_47. 19
- [Nager et al., 2019] NAGER, Y., CENSI, A. et FRAZZOLI, E. (2019). What lies in the shadows? Safe and computation-aware motion planning for autonomous vehicles using intent-aware dynamic shadow regions. In *International Conference on Robotics and Automation (ICRA)*, pages 5800–5806. <https://doi.org/10.1109/ICRA.2019.8793557>. 120
- [Narksri et al., 2019] NARKSRI, P., TAKEUCHI, E., NINOMIYA, Y. et TAKEDA, K. (2019). Crossing Blind Intersections from a Full Stop Using Estimated Visibility of Approaching Vehicles. In *IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 2427–2434. <https://doi.org/10.1109/ITSC.2019.8917323>. 33, 39
- [Naumann et al., 2019] NAUMANN, M., KÖNIGSHOF, H., LAUER, M. et STILLER, C. (2019). Safe but not Overcautious Motion Planning under Occlusions and Limited Sensor Range. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 140–145. <https://doi.org/10.1109/IVS.2019.8814251>. 34, 39, 82
- [Orzechowski et al., 2018] ORZECHOWSKI, P. F., MEYER, A. et LAUER, M. (2018). Tackling Occlusions Limited Sensor Range with Set-based Safety Verification. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1729–1736. <https://doi.org/10.1109/ITSC.2018.8569332>. 34, 39, 82
- [Paden et al., 2016] PADEN, B., ČÁP, M., YONG, S. Z., YERSHOV, D. et FRAZZOLI, E. (2016). A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*, 1(1):33–55. <https://doi.org/10.48550/arXiv.1604.07446>. 19, 32, 86

- [Poncelet *et al.*, 2020] PONCELET, R., VERROUST-BLONDET, A. et NASHASHIBI, F. (2020). Safe geometric speed planning approach for autonomous driving through occluded intersections. In *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 393–399. <https://doi.org/10.1109/ICARCV50220.2020.9305405>. 76, 90
- [Quinlan et Khatib, 1993] QUINLAN, S. et KHATIB, O. (1993). Elastic bands. Connecting path planning and control. *Proceedings - IEEE International Conference on Robotics and Automation*, 2:802–807. <https://doi.org/10.1109/ROBOT.1993.291936>. 36, 39
- [Reichardt et Schick, 1994] REICHARDT, D. et SCHICK, J. (1994). Collision avoidance in dynamic environments applied to autonomous vehicle guidance on the motorway. *Intelligent Vehicles Symposium, Proceedings*, pages 74–78. <https://doi.org/10.1109/IVS.1994.639475>. 35, 39
- [Renault Group, 2020] Renault Group (2020). Les enjeux de la voiture autonome. <https://www.renaultgroup.com/news-onair/actualites/les-enjeux-de-la-voiture-autonome/>. 3
- [Rösmann *et al.*, 2012] RÖSMANN, C., FEITEN, W., WÖSCH, T., HOFFMANN, F. et BERTRAM, T. (2012). Trajectory modification considering dynamic constraints of autonomous robots. In *7th German Conference on Robotics, ROBOTIK 2012*, pages 74–79. <https://ieeexplore.ieee.org/document/6309484>. 36, 39
- [Rungis marché international, 2016] Rungis marché international (2016). Transport, la logistique du dernier kilometre. <https://www.rungisinternational.com/a-1-heure-du-digital/transport-logistique-dernier-kilometre/>. 3
- [Sadou *et al.*, 2004] SADOU, M., POLOTSKI, V. et COHEN, P. (2004). Occlusions in obstacle detection for safe navigation. *IEEE Intelligent Vehicles Symposium, Proceedings*, pages 716–721. <https://doi.org/10.1109/IVS.2004.1336472>. 18
- [Sánchez *et al.*, 2022] SÁNCHEZ, M. G., NYBERG, T., PEK, C., TUMOVA, J. et MARTIN, T. (2022). Foresee the Unseen : Sequential Reasoning about Hidden Obstacles for Safe Driving. *Intelligent Vehicles Symposium, 2022 IEEE*. <https://doi.org/10.1109/IV51971.2022.9827171>. 34, 39, 82
- [Sattel et Brandt, 2005] SATTEL, T. et BRANDT, T. (2005). Ground vehicle guidance along collision-free trajectories using elastic bands. *Proceedings of the American Control Conference*, 7:4991–4996. <https://doi.org/10.1109/ACC.2005.1470798>. 36, 39
- [Schorner *et al.*, 2019] SCHORNER, P., TOTTEL, L., DOLL, J. et ZOLLNER, J. M. (2019). Predictive trajectory planning in situations with hidden road users using partially observable markov decision processes. In *IEEE Intelligent Vehicles Sym-*

- posium (IV)*, pages 2299–2306. <https://doi.org/10.1109/IVS.2019.8814022>. 35, 39
- [Seder et Petrović, 2007] SEDER, M. et PETROVIĆ, I. (2007). Dynamic window based approach to mobile robot motion control in the presence of moving obstacles. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 1986–1991. <https://doi.org/10.1109/ROBOT.2007.363613>. 28, 38
- [Shalev-Shwartz *et al.*, 2017] SHALEV-SHWARTZ, S., SHAMMAH, S. et SHASHUA, A. (2017). On a formal model of safe and scalable self-driving cars. *CoRR*, abs/1708.06374. <http://arxiv.org/abs/1708.06374>. 119
- [Snider, 2009] SNIDER, J. M. (2009). Automatic steering methods for autonomous automobile path tracking. Rapport technique CMU-RI-TR-09-08, Carnegie Mellon University, Pittsburgh, PA. https://www.ri.cmu.edu/pub_files/2009/2/Automatic_Steering_Methods_for_Autonomous_Automobile_Path_Tracking.pdf. 91
- [Tas et Stiller, 2018] TAS, O. S. et STILLER, C. (2018). Limited Visibility and Uncertainty Aware Motion Planning for Automated Driving. *In IEEE Intelligent Vehicles Symposium (IV)*, pages 1171–1178. <https://doi.org/10.1109/IVS.2018.8500369>. 34, 39, 82, 117
- [Tsourveloudis *et al.*, 2001] TSOURVELOUDIS, N. C., VALAVANIS, K. P. et HEBERT, T. (2001). Autonomous vehicle navigation utilizing electrostatic potential fields and fuzzy logic. *IEEE transactions on robotics and automation*, 17(4):490–497. <https://doi.org/10.1109/70.954761>. 28, 37
- [van den Berg *et al.*, 2008] van den BERG, J., LIN, M. et MANOCHA, D. (2008). Reciprocal velocity obstacles for real-time multi-agent navigation. *In IEEE International Conference on Robotics and Automation (ICRA)*, pages 1928–1935. <https://doi.org/10.1109/ROBOT.2008.4543489>. 28, 38
- [van den Berg et Overmars, 2008] van den BERG, J. et OVERMARS, M. (2008). Planning time-minimal safe paths amidst unpredictably moving obstacles. *The International Journal of Robotics Research*, 27(11-12):1274–1294. <https://doi.org/10.1177/0278364908097581>. 28, 37
- [Wilkie *et al.*, 2009] WILKIE, D., van den BERG, J. et MANOCHA, D. (2009). Generalized velocity obstacles. *In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5573–5578. <http://doi.org/10.1109/IROS.2009.5354175>. 28, 38
- [Wolf et Burdick, 2008] WOLF, M. T. et BURDICK, J. W. (2008). Artificial potential functions for highway driving with collision avoidance. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 3731–3736. <https://doi.org/10.1109/ROBOT.2008.4543783>. 35, 39

- [Yu *et al.*, 2019] YU, M. Y., VASUDEVAN, R. et JOHNSON-ROBERSON, M. (2019). Occlusion-aware risk assessment for autonomous driving in urban environments. *IEEE Robotics and Automation Letters*, 4(2):2235–2241. <https://doi.org/10.1109/LRA.2019.2900453>. 33, 39
- [Zhou, 2020] ZHOU, Q. (2020). Pymesh. <https://github.com/PyMesh/PyMesh>. VII

RÉSUMÉ

L'objectif de cette thèse est de planifier le mouvement d'un véhicule autonome pour qu'il puisse circuler sans collision dans un environnement urbain en présence d'autres obstacles mobiles. Ceci implique la prise en compte de contraintes propres au véhicule (dynamiques, énergétiques, liées aux systèmes de perception ou de prévision), relatives à la sécurité, au confort des usagers et à l'environnement proche du véhicule. Celles-ci dépendent de la topologie de la route, des règles de conduite et des autres acteurs en mouvement dans le voisinage de l'ego véhicule. Dans le cadre de cette thèse, nous nous sommes intéressés plus particulièrement : (i) à la perception limitée ou partielle de l'environnement proche du véhicule et en particulier à la présence d'obstacles fixes ou mobiles occultant une partie de son voisinage proche ; (ii) à l'incertitude sur la prédiction du mouvement des autres agents ; (iii) aux manœuvres à effectuer dans des scénarios urbains où les voies peuvent être encombrées par de nombreux véhicules arrêtés ou en mouvement ; (iv) et au respect de la signalisation. Toutes ces problématiques ont été traitées par des approches géométriques en adaptant de différentes manières le principe de décomposition chemin-vitesse introduit par Kant et Zucker (1986).

Le simulateur CARLA a été adapté pour valider les approches proposées sur différents scénarios reproduisant des situations typiques d'interaction avec d'autres véhicules en milieu urbain, comme, par exemple, le franchissement d'un carrefour avec une visibilité réduite, le dépassement d'un véhicule lent ou arrêté sur sa voie, la conduite sur une portion de route réglementée par des feux de circulation, etc.

MOTS CLÉS

Véhicule autonome, Planification de trajectoire, Décomposition chemin-vitesse, Robotique.

ABSTRACT

The objective of this thesis is to plan the motion of an autonomous vehicle so that it can navigate and interact with other moving obstacles without collision in an urban environment. This involves taking into account constraints that may be specific to the vehicle (dynamic, energetic, linked to perception or forecasting systems), relating to the safety and comfort of users or to the environment close to the vehicle. These last depend on the topology of the road, driving rules and other moving actors in the neighborhood of the ego-vehicle. In the context of this thesis, we focused on : (i) the limited or partial perception of the environment close to the vehicle and in particular the presence of static or mobile obstacles hiding part of its neighborhood ; (ii) uncertainty about the prediction of the motion of other agents ; (iii) maneuvers to be carried out in urban scenarios where the lanes can be congested with plenty of other static or moving obstacles ; (iv) and the respect of the road signs. All these issues have been addressed through geometric approaches by adapting in different ways the principle of path-velocity decomposition introduced by Kant and Zucker (1986).

The CARLA simulator has been adapted to validate the proposed approaches on different scenarios reproducing typical situations of interaction with other vehicles in an urban environment, such as, for example, crossing a crossroads with reduced visibility, overtaking a slow or stopped vehicle in its lane, driving in a section of road regulated by traffic lights, etc.

KEYWORDS

Self-driving car, Trajectory planning, Path-velocity decomposition, Robotics.