



HAL
open science

Planification de trajectoire dans un environnement peu contraint et fortement dynamique

Pierre de Beaucorps

► **To cite this version:**

Pierre de Beaucorps. Planification de trajectoire dans un environnement peu contraint et fortement dynamique. Intelligence artificielle [cs.AI]. Sorbonne Université, 2019. Français. NNT : 2019SORUS082 . tel-03369204

HAL Id: tel-03369204

<https://theses.hal.science/tel-03369204>

Submitted on 7 Oct 2021

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.

École doctorale Informatique, Télécommunications et Électronique (Paris)

Doctorat Sorbonne Université

THÈSE

pour obtenir le grade de docteur délivré par

L'Université Pierre et Marie Curie

présentée et soutenue publiquement par

Pierre DE BEAUCORPS

PLANIFICATION DE TRAJECTOIRE DANS UN ENVIRONNEMENT PEU CONTRAINT ET FORTEMENT DYNAMIQUE

Directrice de thèse : **Anne VERROUST-BLONDET**

Co-encadrant de thèse : **Fawzi NASHASHIBI**

Jury

| | | |
|------------------------------|---------------------------------------|------------|
| M. Michel Basset, | Professeur | Rapporteur |
| M. Thierry Fraichard, | Chargé de recherche (HDR) | Rapporteur |
| M. Faiz Ben Amar, | Professeur | Examineur |
| M. David Filliat, | Professeur | Examineur |
| M. Benazouz Bradai, | Valeo Project Manager & Senior Expert | Invité |

Table des matières

| | |
|--|------------|
| Table des matières | iii |
| Liste des figures | v |
| Liste des tableaux | ix |
| 1 Introduction | 1 |
| 1.1 Objectifs de la thèse | 4 |
| 1.2 Décomposition du problème de la planification de trajectoire | 5 |
| 1.3 Contributions principales | 7 |
| 1.4 Liste des publications | 7 |
| 2 Planification de vitesse dans un environnement urbain dynamique | 9 |
| 2.1 Introduction | 11 |
| 2.2 État de l'art | 11 |
| 2.3 Conception et développement d'un outil de simulation | 30 |
| 2.4 Planification de vitesse aux intersections | 36 |
| 2.5 Extension aux environnements semi-contraints | 47 |
| 2.6 Tests sur véhicules réels | 68 |
| 2.7 Conclusion | 76 |
| 3 Planification de chemin dans un environnement fortement dynamique | 77 |
| 3.1 Introduction | 79 |
| 3.2 État de l'art | 79 |
| 3.3 Reachable Interaction Sets : présentation de l'approche | 93 |
| 3.4 Application des RIS à la planification de mouvement | 98 |
| 3.5 Validation des approches de planification | 111 |
| 3.6 Conclusion | 128 |
| 4 Conclusion | 129 |
| 4.1 Planification de vitesse | 130 |
| 4.2 Planification de chemin | 131 |
| 4.3 Perspective future : vers une planification chemin et vitesse? | 132 |
| 5 Références | 133 |
| A Liste des acronymes | I |
| B Glossaire | III |

Liste des figures

| | |
|---|----|
| 1.1 Emissions de gaz à effet de serre en France | 3 |
| 1.2 L'intelligence de la voiture autonome | 4 |
| 1.3 Architecture générale | 6 |
| 2.1 Les différents types de modélisation du mouvement | 22 |
| 2.2 Topologies de route principalement utilisées avec le simulateur | 31 |
| 2.3 Panneaux de signalisation disponibles dans le simulateur | 31 |
| 2.4 Exemple d'utilisation du script de génération de carte | 33 |
| 2.5 Vue du poste de conduite en simulation | 34 |
| 2.6 Aperçu du rendu de simulation | 34 |
| 2.7 Menu graphique du mode moniteur | 35 |
| 2.8 Schémas des contrôleurs | 36 |
| 2.9 Les trois intersections urbaines utilisées dans les scénarios | 38 |
| 2.10 Les six scénarios pour la topologie de carrefour en X | 38 |
| 2.11 Schéma de l'intersection durant la synchronisation | 39 |
| 2.12 Consigne du profil de vitesse lors du suivi | 40 |
| 2.13 Questionnaire de fin | 41 |
| 2.14 Schéma explicatif du PET | 42 |
| 2.15 Profils de vitesse d'engagement enregistrés en simulation et leurs clusters | 43 |
| 2.16 Profils de vitesse cédant le passage enregistrés en simulation et leurs clusters | 43 |
| 2.17 Profils de référence après clustering et leur simulation | 44 |
| 2.21 Les trois scénarios considérés | 49 |
| 2.22 Différentes situations de croisement problématiques | 52 |
| 2.23 Comportement ACC du profil de croisière | 54 |
| 2.24 Schéma d'une situation avec deux zones de collision potentielle | 55 |
| 2.25 L'arbre des combinaisons possibles entre les différents profils de références | 56 |
| 2.26 Les combinaisons possibles de profils de vitesse pour deux zones | 57 |
| 2.27 Aperçu de la topologie de route semi-contrainte dans le simulateur | 58 |
| 2.28 Les différentes variantes de chemins du scénario « cut-in » | 59 |
| 2.29 Les chemins du VA et des véhicules obstacles sur le scénario « yield » | 60 |
| 2.30 Les chemins des véhicules obstacles sur le scénario « merge » | 60 |
| 2.31 Histogrammes des valeurs de TG pour le scénario « cut-in » | 62 |
| 2.32 Les positions d'arrêt du VA sur le scénario « yield » avec les chemins effectifs | 63 |
| 2.33 TG des différents cas de figure du scénario « yield » | 64 |
| 2.34 TG des deux comportements du VA dans le scénario « merge » | 64 |
| 2.35 Séquence de rendus pour un cas du « groupe A » | 66 |
| 2.36 Séquence de rendus pour un cas du « groupe B » | 66 |
| 2.37 Vue satellite de l'environnement de test | 69 |
| 2.38 Intérieur du véhicule automatisé | 70 |
| 2.39 Optimisation exacte du calcul du TG | 70 |
| 2.40 Approximation du calcul du TG | 70 |
| 2.41 Prise en compte de l'incertitude de l'angle de lacet | 71 |

| | |
|--|-----|
| 2.42 Profil de la vitesse mesurée du véhicule | 72 |
| 2.43 Décision de céder le passage en situation réelle | 73 |
| 2.44 Décision de s'engager en situation réelle | 74 |
| 2.45 Décision de céder le passage avec un seuil de TG agressif | 75 |
| 3.1 Construction d'un graphe de visibilité | 81 |
| 3.2 Constitution des champs de potentiel artificiel | 81 |
| 3.3 Développement d'un arbre à exploration rapide | 82 |
| 3.4 Exemples de chemins de Dubins | 84 |
| 3.5 Exemple de chemin de Reeds-Shepp | 84 |
| 3.6 Les nœuds dans l'espace temps-état | 85 |
| 3.7 Trajectoire évitant les cônes des obstacles | 86 |
| 3.8 Construction d'un obstacle vitesse | 87 |
| 3.9 méthode APF-SR : champ de potentiel associé à la densité de probabilité de présence | 89 |
| 3.10 méthode SES : utilisation d'un arbre RRT dans un environnement dynamique | 89 |
| 3.11 Exemple de Reachable Interaction Sets | 94 |
| 3.12 Reachable set d'un robot non-holonyme | 95 |
| 3.13 Deux lois stochastiques pour la vitesse longitudinale | 96 |
| 3.14 Cartes de probabilité d'un mouvement déterministe | 96 |
| 3.15 Cartes de probabilité d'un mouvement avec aléa | 96 |
| 3.16 Cartes de probabilité d'un mouvement avec beaucoup d'aléa | 97 |
| 3.17 Influence du seuil sur le volume de l'obstacle stochastique | 97 |
| 3.18 Définition de la destination intermédiaire | 99 |
| 3.19 Composantes des potentiels de <i>ris-apf</i> | 100 |
| 3.20 L'approche de planification locale <i>ris-bézier</i> avec deux courbes de Bézier | 101 |
| 3.21 Le chemin de visibilité autour d'un sommet d'obstacle | 103 |
| 3.22 Exemple d'un graphe des combinaisons possibles d'arcs de cercle | 104 |
| 3.23 Construction du chemin de Dubins complet | 105 |
| 3.24 Exemples de chemins de Dubins complets | 106 |
| 3.25 Cas de figure d'échec de l'approche | 107 |
| 3.26 Les différents espaces de configuration | 109 |
| 3.27 Volume 3D d'un obstacle dans l'espace-temps avec le nouvel espace de configuration | 110 |
| 3.28 Trajectoires d'urgence pour sortir le robot d'un RIS | 110 |
| 3.29 Les quatre environnements de validation | 113 |
| 3.30 Taux de succès avec des obstacles plus lents que le robot | 114 |
| 3.31 Taux de succès avec des obstacles plus rapides que le robot | 114 |
| 3.32 Taux de succès pour l'environnement <i>cross</i> | 114 |
| 3.33 Longueur du chemin avec des obstacles plus lents que le robot | 115 |
| 3.34 Longueur du chemin avec des obstacles plus rapides que le robot | 115 |
| 3.35 Longueur du chemin pour l'environnement <i>cross</i> | 115 |
| 3.36 Aperçu des polygones de RIS durant la planification | 116 |
| 3.37 Séquence comparant les approches <i>ris-apf</i> , <i>dynamic-apf</i> et <i>vo</i> | 116 |
| 3.38 Séquence comparant les trois approches qui utilisent les RIS | 117 |
| 3.39 Autre séquence comparant les trois approches qui utilisent les RIS | 117 |
| 3.40 Tests sur la période de re planification | 119 |
| 3.41 Les quatre environnements de validation | 121 |
| 3.43 Environnement open et vitesses var-faible | 124 |
| 3.44 Environnement open et vitesses var-élevée | 124 |
| 3.45 Environnement open et vitesses var-très-élevée | 124 |
| 3.47 Environnement gate et vitesses var-faible | 125 |
| 3.48 Environnement gate et vitesses var-élevée | 125 |
| 3.49 Environnement gate et vitesses var-très-élevée | 125 |
| 3.51 Environnement lanes et vitesses var-faible | 126 |

| | |
|---|-----|
| 3.52 Environnement open-rect et vitesses var-faible | 126 |
| 3.53 Environnement open-rect avec un long robot | 126 |
| 3.55 Tracés de 3 chemins dans l'environnement open | 127 |
| 3.56 Tracés de 3 autres chemins dans l'environnement open | 127 |
| 3.57 Tracés de 3 chemins dans l'environnement open-rect | 127 |

Liste des tableaux

| | | |
|-----|--|-----|
| 1.1 | Les 6 niveaux d'automatisation de la conduite | 5 |
| 2.1 | Les principaux modèles de Markov | 12 |
| 2.2 | Tableau récapitulatif des approches de planification de trajectoire pour un véhicule automatisé | 21 |
| 2.3 | Relations entre TTC, TG et le danger réel de la situation | 29 |
| 2.4 | Paramètres des configurations | 39 |
| 2.5 | Paramètres du modèle IDM | 51 |
| 2.6 | Les différentes situations finales du scénario « yield » | 64 |
| 3.1 | Tableau récapitulatif des principales approches robotiques de planification de trajectoire en présence d'obstacles mobiles | 92 |
| 3.2 | La loi de vitesse var-faible | 120 |
| 3.3 | La loi de vitesse var-élevée | 120 |
| 3.4 | La loi de vitesse var-très-élevée | 120 |

Chapitre 1

Introduction

« Un conducteur dangereux, c'est celui qui vous dépasse malgré tous vos efforts pour l'en empêcher. »

Woody Allen

Sommaire

| | |
|---|----------|
| 1.1 Objectifs de la thèse | 4 |
| 1.2 Décomposition du problème de la planification de trajectoire | 5 |
| 1.3 Contributions principales | 7 |
| 1.4 Liste des publications | 7 |

Le secteur des transports et la progression des technologies y étant associées ont souvent été le symbole du développement de nos sociétés modernes. L'efficacité des moteurs, l'aérodynamisme et le design de la carrosserie, le confort de l'habitacle n'ont eu de cesse d'être améliorés par les avancées technologiques au cours du siècle passé.

Aujourd'hui, l'avenir de la voiture se dessine entre électrification et automatisation, promettant respectivement une pollution moindre et un usage optimisé des véhicules.

D'aucuns prédisent les principaux bénéfices apportés par les avancées technologiques des voitures durant les prochaines années :

Augmentation de la sécurité routière

Grâce aux avancées en matière d'automatisation, les accidents de la route deviendraient moins fréquents. En effet, 3 400 victimes de ces accidents sont décédées en France en 2017 (source : INSEE). Ce chiffre a néanmoins pu être divisé par 3 au cours des 20 dernières années, il s'agit donc d'éviter encore plus d'accidents notamment grâce à des systèmes d'aide à la conduite (ADAS) plus performants capables de prévenir le conducteur plusieurs secondes avant une situation critique. On envisage également la mise en place de systèmes de détection du niveau de vigilance du conducteur (*driver monitoring*), ainsi que, à terme, la mise en circulation de voitures totalement autonomes.

Amélioration du confort de conduite

Ce bénéfice est souvent mis en avant car il concerne directement le quotidien du conducteur. On peut d'ores et déjà observer des véhicules de série avec une option de stationnement automatique, un système d'aide au maintien dans la voie, ou encore un pilotage semi-automatique dans les embouteillages.

Fluidification des axes routiers

Au niveau des infrastructures, la possibilité de communiquer automatiquement avec les véhicules permettrait d'optimiser les flux en adaptant en temps réel la signalisation, de désengorger les agglomérations, ainsi que d'éviter la formation d'embouteillages sur les voies rapides grâce à la possibilité de former spontanément des trains de véhicules (*platooning*).

Diminution de la pollution

En 2016, l'Agence nationale de santé publique a recensé 48 000 morts prématurées liées à la pollution par les particules fines en France. Ce chiffre indirect est établi à partir d'une corrélation statistique entre les niveaux de pollution aux particules fines et les risques pour la santé. Avec l'électrification des véhicules, on peut s'attendre à ce que les villes deviennent plus respirables. Dans une moindre mesure, l'automatisation des véhicules peut également permettre d'améliorer le profil d'accélération utilisé en ville, ce qui a pour effet de limiter la pollution.

Cependant, il serait partiel d'évoquer ces perspectives optimistes d'amélioration des systèmes de transports en omettant ce qui constitue probablement un des principaux défis majeurs à mener dans ce domaine : la décarbonation des transports. Car même si on a l'habitude d'attribuer notre confort de vie moderne et nos systèmes de transports efficaces aux améliorations technologiques fulgurantes du XX^e siècle, ils se sont également développés grâce à un apport énergétique massif qui a été permis par un recours quasi-systématique à l'extraction d'hydrocarbures fossiles.

Or, le changement climatique, dont l'origine anthropique ne laisse plus de place au doute (cf. [INTERGOVERNMENTAL PANEL ON CLIMATE CHANGE \[2018\]](#)), est une des limites à cette utilisation des énergies fossiles. La finitude de ces stocks en est une autre, puisque le pic global d'extraction de pétrole a récemment été estimé à devoir survenir durant la décennie 2020-2030 (cf. [IEA \[2018\]](#)).

Pour prendre l'exemple français, selon l'Agence de l'Environnement et de la Maîtrise de l'Énergie, le secteur des transports représente 33% de la consommation d'énergie finale en France, ce qui en fait le principal secteur d'émission de CO₂ avec 39% des émissions totales de GES, dont 25% pour le seul transport routier (voir Fig. 1.1).

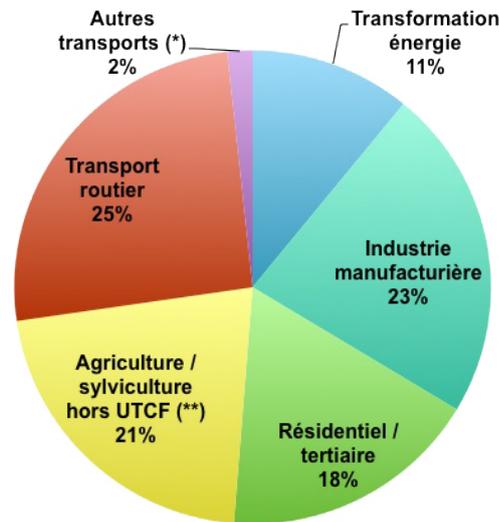


FIGURE 1.1 – Répartition par secteur des émissions de gaz à effet de serre en France. (*) le transport aérien ou maritime international n'est pas pris en compte sur ce graphe (source : CITEPA)

Il apparaît donc urgent de diminuer fermement la consommation globale de carburants d'origine fossile dans le domaine des transports. Il est ici crucial de considérer l'aspect « global » de la consommation, et ce afin d'éviter deux problèmes notables :

- Il est tout à fait possible de diminuer localement la pollution sans pour autant améliorer la balance planétaire des émissions de GES.
- De même, une amélioration de l'efficacité énergétique sur un véhicule n'est pas forcément corrélée avec une baisse de la consommation énergétique globale.

Le premier problème consiste à déporter la pollution d'une zone urbaine vers une autre région, ce qui peut être bénéfique pour la santé des habitants mais ne change rien aux problèmes des émissions de CO₂ et de déplétion des ressources fossiles. L'électrification massive des véhicules, dans les villes chinoises notamment, en est un exemple dans la mesure où l'énergie primaire nécessaire à la consommation électrique de ces véhicules trouve toujours majoritairement sa source dans des centrales à charbon. Il faut également tenir compte de l'importante énergie grise requise lors de la production du véhicule, ainsi que des futurs problèmes probables d'approvisionnement en métaux précieux notamment utilisés dans les batteries (cf. [PITRON \[2018\]](#)).

Le deuxième, plus subtil, est lié à la notion d'effet rebond. Historiquement décrit par l'économiste William Stanley Jevons dans « The coal question » (cf. [JEVONS \[1866\]](#)), l'amélioration de l'efficacité énergétique d'une machine a toujours eu pour corollaire une augmentation globale de la consommation énergétique pour la simple raison qu'une efficacité accrue d'une technologie engendre une croissance de ses usages. Ce paradigme demeure valable depuis que Jevons l'a étudié à propos de la consommation de charbon au début de l'ère industrielle.

Pour prendre un exemple plus moderne, entre la Citroën 2CV de 1968 et la Citroën C1 de 2018, la consommation de carburant est restée inchangée en 50 ans à environ 5 L aux 100 km, et ce en dépit des améliorations majeures en matière d'efficacité énergétique des moteurs à combustions. La raison principale tient au fait que le véhicule est 1,7 fois plus lourd, et 2 fois plus puissant de sorte à être capable d'atteindre 170 km/h.

Ainsi il faut être particulièrement vigilant lorsqu'on évalue les mérites écologiques d'une solution technique, et particulièrement dans le domaine de la mobilité, dès lors que des usages nouveaux se développent au point de venir compenser par la consommation ce qui a été gagné par la

technologie.



FIGURE 1.2 – Intelligence et voiture autonome selon ©Philippe Geluck.

En résumé, il nous semble que l'enjeu majeur des transports de demain semble plus se situer du côté de la sobriété énergétique et de la lutte contre les émissions de gaz à effet de serre que de celui d'un confort de conduite accru ou d'un gain marginal en matière de sécurité, ce qui n'est pas toujours mis en avant par ceux qui industrialisent ces solutions technologiques.

La priorité des avancées technologiques d'automatisation et de communication entre véhicules pourrait être de remplacer, en environnement urbain, la majeure partie des véhicules personnels par des voitures partagées afin de réduire globalement le parc de véhicules, et de faciliter la supervision d'une transition massive vers des voitures légères et électriques. Nous pensons que les futures avancées des voitures intelligentes de demain leur permettraient de s'intégrer pleinement aux réseaux de transports en commun, et ainsi d'effectuer la liaison du dernier mètre pour les usagers auparavant trop éloignés de ces réseaux.

Cependant, les technologies actuelles en matière d'automatisation des véhicules ne sont pas encore suffisantes pour permettre le déploiement de tels usages. L'environnement urbain en particulier, qui comporte un grand nombre d'utilisateurs de la route ayant des intentions très différentes et interagissant de façon complexe, présente de nombreux défis techniques. La planification de trajectoire d'un véhicule automatisé dans un environnement très dynamique constitue un de ces défis, et fait l'objet de cette thèse.

1.1 Objectifs de la thèse

La question de la planification de trajectoire d'un véhicule robotisé fait l'objet de nombreux travaux, dont une partie porte sur la circulation dans des environnements urbains. La majorité des approches existantes sont dédiées à des situations où le mouvement du véhicule est restreint par des voies de circulation marquées au sol et par une signalisation verticale claire. La trajectoire à suivre peut ainsi être réduite à un ensemble discret de manœuvres. Or, ces différentes restrictions, qui constituent ce que l'on appelle la « topologie de la route », ne permettent pas en pratique de couvrir tous les cas de figure d'un environnement urbain réaliste.

Les travaux de cette thèse porteront sur des cas où l'environnement n'est quasiment pas contraint. Nous considérerons des intersections non contrôlées par des feux de signalisation et de larges espaces navigables où les contraintes imposées par le marquage au sol sont peu nombreuses. L'objectif est d'effectuer la planification de trajectoire dans un tel environnement, et en présence d'autres véhicules en mouvement.

Ces travaux sont destinés à être utilisés pour contrôler le mouvement d'une voiture autonome

| Niveau d'autonomie | Nom | Description |
|--------------------|-------------------------------|---|
| 0 | Aucune assistance | Le conducteur contrôle complètement le véhicule. |
| 1 | Assistance à la conduite | Le système peut assister le conducteur en prenant en charge le contrôle longitudinal ou le contrôle latéral du véhicule. |
| 2 | Automatisation partielle | Dans certaines situations, le système peut prendre en charge le contrôle du véhicule, mais le conducteur reste responsable et doit reprendre le contrôle si nécessaire. |
| 3 | Automatisation conditionnelle | Dans certaines conditions (ex : autoroute), la conduite peut être totalement déléguée au système. Le conducteur doit rester capable de reprendre le contrôle immédiatement si nécessaire. |
| 4 | Forte automatisation | Si les conditions sont réunies, la conduite est complètement déléguée au système et le conducteur n'en a plus la responsabilité. Dans le cas contraire, le système doit être en mesure de s'arrêter seul. |
| 5 | Automatisation complète | Ces systèmes n'ont jamais besoin de l'action du conducteur. La voiture conduit seule dans toutes les conditions. |

TABLEAU 1.1 – Les 6 niveaux d'automatisation de la conduite, selon le standard J3016 de la SAE International.

dans un contexte urbain. Nous considérerons également le problème plus théorique de la planification de mouvement d'un robot qui doit rejoindre une destination en évitant des obstacles pouvant se déplacer plus rapidement que le robot.

En termes de niveau d'automatisation du véhicule, cette planification de trajectoire pourrait être intégrée dans un système de niveau 3 (automatisation conditionnelle, ADAS) à 5 (automatisation complète, véhicule autonome), selon le standard J3016 (cf. Tableau 1.1). Notre cadre de travail est de considérer un type d'environnement particulier que l'on qualifiera de « semi-contraint » où une topologie de route vraiment stricte ne peut être spécifiée, comme dans le cas de l'autoroute ou d'une intersection urbaine comportant des feux de signalisation.

On peut trouver des environnements « semi-contraints » où cette topologie est moins clairement définie sur certaines larges intersections de métropoles (à Paris : la place de la Bastille, la place de l'Étoile...), au niveau des péages d'autoroutes, sur certains parkings où le marquage au sol est absent et sur des voies comportant des zones de travaux. Le code de la route s'applique toujours dans ces cas, cependant il est bien plus difficile pour un système robotique d'identifier un ensemble discret de manœuvres, ou de tracer un chemin pour le véhicule sans s'appuyer sur la topologie de l'environnement.

Les travaux de cette thèse ont été menés au sein de l'équipe RITS (*Robotics and Intelligent Transportation Systems*) d'Inria. Ils ont bénéficié du soutien financier de Valeo Driving Assistance Research, dans le cadre du projet Valeo Drive4U®, dirigé par Benazouz Bradai.

1.2 Décomposition du problème de la planification de trajectoire

Nous considérerons par la suite que nous disposons d'une perception adéquate de l'environnement, ainsi que d'une prédiction des futurs mouvements des obstacles mobiles présents dans le voisinage du véhicule. Il est également admis qu'un module de contrôle ajuste les déplacements longitudinaux et latéraux du véhicule en fonction de la trajectoire optimale retenue par la planification.

Il est assez courant de décomposer le calcul de trajectoire en effectuant premièrement une recherche de chemin qui contourne les obstacles statiques et de mener ensuite une planification dynamique de la vitesse sur ce chemin afin de tenir compte des obstacles mobiles. L'architecture générale du système de transport robotisé correspondant à cette décomposition est schématisée Fig. 1.3. On notera ici la distinction entre les notions de chemin et de trajectoire : le premier

consiste en une succession de points de l'espace navigable que le système doit suivre pour se rendre à une destination, tandis que la trajectoire en est l'extension temporelle qui ajoute l'information du profil de vitesse à suivre le long du chemin.

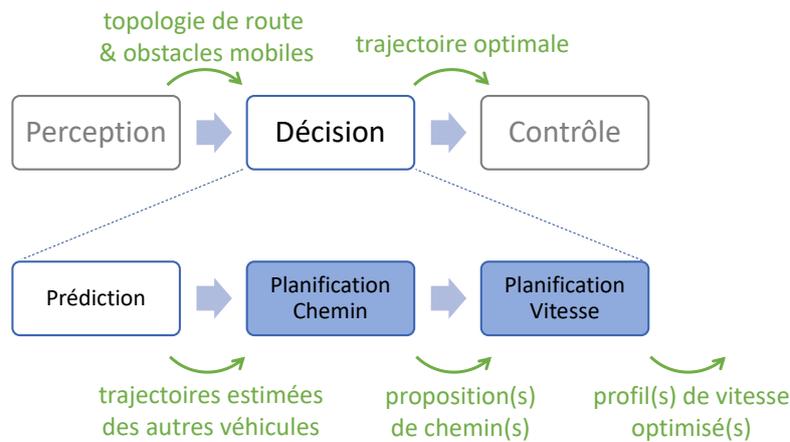


FIGURE 1.3 – Architecture générale de la décision et planification de trajectoire d'un système de transport robotisé.

Cette stratégie de décomposition est proposée pour la première fois dans **KANT et ZUCKER [1986]**, afin de réduire la complexité du problème. L'intuition derrière cette décomposition est présentée comme provenant d'un scénario de soirée cocktail durant laquelle une personne cherche à se déplacer à travers une foule dense. Les auteurs expliquent qu'une approche simple et souvent préférée dans ce cas est de suivre le chemin direct vers la destination, et de moduler sa vitesse de façon réactive en fonction des obstacles que l'on rencontre.

Pour autant on pourrait envisager l'approche réciproque où le profil de vitesse est déterminé en premier, puis un chemin permettant de le respecter est recherché. Par exemple, on peut observer cette deuxième situation lorsqu'un piéton pressé contourne une poussette croisant son chemin : en présence d'un obstacle plutôt long et relativement lent, le piéton préférera souvent conserver sa vitesse quitte à allonger son parcours.

Il est entendu que ces deux approches sont deux extrémités d'un continuum, qui offre une infinité de combinaisons vitesse-chemin possibles. Cependant on remarque que pour une même durée de trajet, la préférence pour un chemin plus court ou une vitesse plus élevée dépend en partie du contexte.

Deux stratégies résultant de cette décomposition chemin-vitesse sont suivies dans cette thèse, chacune faisant l'objet d'un chapitre.

Le chapitre 2 présente des approches de planification du profil de vitesse pour un véhicule automatisé circulant dans un environnement « semi-contraint ». Une approche d'adaptation dynamique de la vitesse est d'abord introduite pour des scénarios d'intersections simples (carrefour en X et carrefour giratoire). Elle utilise des profils de vitesse inspirés de la conduite humaine combinés à une prédiction de risque, ce qui permet de générer un profil de vitesse le plus confortable possible tout en évitant les autres véhicules. Cette approche est ensuite généralisée pour des environnements « semi-contraints ». Des validations quantitatives en simulation ainsi que qualitatives sur un véhicule automatisé sont proposées.

Dans le chapitre 3 un nouveau cadre de travail est introduit pour effectuer la planification de chemin d'un système robotisé en présence d'obstacles mobiles. La spécificité de cette approche est de permettre un évitement des obstacles y compris lorsque ceux-ci sont « très dynamiques », c'est à dire ayant une vitesse nettement plus grande que celle du robot. L'approche consiste à émettre une hypothèse a priori sur la vitesse du robot, et à s'en servir pour réduire d'une dimension le problème de planification. Une carte de l'espace navigable tenant compte de cette hypothèse de vitesse est dressée, qui représente un problème dual de planification. Une recherche de

chemin statique sur cette carte permet d'obtenir une trajectoire dynamique qui évite les obstacles mobiles. Des validations quantitatives comparent les performances de ces travaux aux approches de l'état de l'art.

1.3 Contributions principales

Les principales contributions de cette thèse sont les suivantes :

- Le développement d'un outil de simulation en 3D de scénarios de conduite avec une modélisation dynamique réaliste des véhicules. Il a été conçu sur l'architecture client-serveur afin de permettre à une multitude de conducteurs humains de participer à la simulation sans avoir à installer aucun logiciel. Les approches introduites ont pu être validées avec ce simulateur qui a également permis d'étudier le comportement des conducteurs humains en simulation.
- Une approche de planification de trajectoire pour un véhicule automatisé sur une topologie de route semi-contrainte. Un modèle MPC ajuste le profil de vitesse dans des limites acceptables pour les capacités du véhicule et le confort de ses occupants, de sorte à minimiser le risque de collision avec les autres véhicules présents dans l'environnement.
- L'introduction des RIS : une nouvelle représentation d'un problème de planification de trajectoire qui utilise une hypothèse sur le mouvement du véhicule ego afin d'agrèger les trois dimensions d'espace et de temps en une carte en deux dimensions sur laquelle une recherche de chemin classique peut s'opérer. Le robot qui suit un chemin évitant les RIS sur cette carte maximise ses chances d'éviter les obstacles au cours du temps.
- Une recherche de chemin de Dubins basée sur un graphe de visibilité, permettant d'obtenir une planification de chemin complète en tenant compte des contraintes cinématiques d'un robot non holonome.

1.4 Liste des publications

- P. de Beaucorps, T. Streubel, A. Verroust-Blondet, F. Nashashibi, B. Bradai, and P. Resende, « Decision-making for Automated Vehicles at Intersections Adapting Human-like Behavior » IEEE Intelligent Vehicles Symposium (IV), Los Angeles, 2017
- P. de Beaucorps, T. Streubel and F. Nashashibi, « Evaluation of automated vehicle behavior in intersection scenarios » RSS2017 - Road Safety & Simulation International Conference, The Hague, 2017
- P. de Beaucorps, A. Verroust-Blondet, R. Poncelet and F. Nashashibi, « RIS : A Framework for Motion Planning among Highly Dynamic Obstacles » 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), Singapour, 2018
- [soumis] P. de Beaucorps, T. Streubel, A. Verroust-Blondet and F. Nashashibi « Decision-making for Automated Vehicles in Semi-constrained Road Topologies by Speed Adaptation » IEEE Transactions on Intelligent Transportation Systems (T-ITS), 2019

Chapitre 2

Planification de vitesse dans un environnement urbain dynamique

« Il ne faut pas confondre vitesse et précipitation. »

Proverbe français

Sommaire

| | |
|---|-----------|
| 2.1 Introduction | 11 |
| 2.2 État de l'art | 11 |
| 2.2.1 Approches en planification de trajectoire pour un véhicule automatisé | 12 |
| 2.2.1.1 Processus de décision markovien partiellement observable | 12 |
| 2.2.1.2 Ontologies | 13 |
| 2.2.1.3 Décomposition chemin vitesse | 14 |
| 2.2.1.4 Planification avec commande prédictive | 15 |
| 2.2.1.5 Champs de potentiel artificiel | 16 |
| 2.2.1.6 Bandes élastiques | 17 |
| 2.2.1.7 Théorie des jeux | 17 |
| 2.2.1.8 Apprentissage profond | 17 |
| 2.2.1.9 Décision coordonnée | 18 |
| 2.2.1.10 Analyse du profil de vitesse des conducteurs humains | 19 |
| 2.2.1.11 Tableau récapitulatif | 19 |
| 2.2.2 Prédiction de mouvement | 22 |
| 2.2.2.1 Prédictions basées sur une extrapolation physique | 23 |
| 2.2.2.2 Prédictions incluant le contexte statique | 23 |
| 2.2.2.3 Prédictions incluant le contexte dynamique | 24 |
| 2.2.2.4 Discussion sur les environnements semi-contraints | 25 |
| 2.2.3 Métriques d'évaluation du risque | 26 |
| 2.2.3.1 Présentation de métriques courantes | 26 |
| 2.2.3.2 État de l'art des métriques de risque en situation de suivi | 27 |
| 2.2.3.3 État de l'art des métriques de risque en situation de croisement | 28 |
| 2.2.4 Conclusion | 29 |
| 2.3 Conception et développement d'un outil de simulation | 30 |
| 2.3.1 Critères retenus pour le simulateur | 30 |
| 2.3.2 Architecture en réseau du simulateur | 32 |
| 2.3.3 Modèle physique des véhicules | 32 |
| 2.3.4 Génération de topologie de route | 32 |
| 2.3.5 Visualisation de la scène et mode moniteur | 33 |

| | | |
|------------|---|-----------|
| 2.3.6 | Systèmes d’asservissement longitudinal et latéral | 35 |
| 2.3.7 | Conclusion | 36 |
| 2.4 | Planification de vitesse aux intersections | 36 |
| 2.4.1 | L’environnement des intersections | 37 |
| 2.4.1.1 | Scénarios | 37 |
| 2.4.1.2 | Configurations | 38 |
| 2.4.1.3 | Algorithme de synchronisation | 39 |
| 2.4.2 | Acquisition de données | 40 |
| 2.4.3 | Conception de l’algorithme | 41 |
| 2.4.3.1 | Profils de vitesse de référence | 42 |
| 2.4.3.2 | Prédiction à court terme | 44 |
| 2.4.3.3 | Processus de décision | 44 |
| 2.4.4 | Validation quantitative | 45 |
| 2.4.5 | Conclusion | 47 |
| 2.5 | Extension aux environnements semi-contraints | 47 |
| 2.5.1 | Un Environnement semi-contraint | 48 |
| 2.5.1.1 | Topologie de route | 48 |
| 2.5.1.2 | Scénarios | 48 |
| 2.5.2 | Algorithme | 49 |
| 2.5.2.1 | Planification de chemin | 50 |
| 2.5.2.2 | Modèle de prédiction | 50 |
| 2.5.2.3 | Estimation du risque | 51 |
| 2.5.2.4 | Profils de vitesse candidats | 53 |
| 2.5.2.5 | Interactions avec plusieurs véhicules | 54 |
| 2.5.2.6 | Prise de décision | 56 |
| 2.5.2.7 | Module de contrôle | 57 |
| 2.5.3 | Validation | 58 |
| 2.5.3.1 | Cut-in | 58 |
| 2.5.3.2 | Yield | 59 |
| 2.5.3.3 | Merge | 60 |
| 2.5.4 | Résultats | 61 |
| 2.5.4.1 | Cut-in | 61 |
| 2.5.4.2 | Yield | 62 |
| 2.5.4.3 | Merge | 64 |
| 2.5.4.4 | Interprétation générale | 66 |
| 2.5.5 | Remarques sur les limites de la validation | 66 |
| 2.5.6 | Conclusion | 68 |
| 2.6 | Tests sur véhicules réels | 68 |
| 2.6.1 | Implémentation des algorithmes sur les véhicules | 69 |
| 2.6.2 | Expérimentations | 71 |
| 2.7 | Conclusion | 76 |

2.1 Introduction

Dans ce chapitre nous considérons un véhicule automatisé désigné par l'acronyme **véhicule automatisé (VA)** ou nommé *véhicule ego*. Ce véhicule évolue dans un environnement urbain comportant d'autres véhicules en mouvement.

L'environnement urbain envisagé est dit *semi-contraint* pour désigner un environnement de conduite particulier dans lequel l'espace navigable est très large de sorte qu'il n'est pas possible de spécifier une topologie de route vraiment stricte, permettant de séparer la route en voies de circulation par exemple. Ceci est le cas dans certaines larges intersections parisiennes (place de la Bastille, place de l'Étoile...), ou encore au niveau des péages d'autoroutes.

Ce type d'**environnement semi-contraint** n'a pratiquement pas été envisagé par les principales approches de planification existantes comme nous le verrons dans la section 2.2. En effet, la plupart d'entre elles se placent dans des cadres spécifiques (autoroute, intersection contrôlée par des feux de signalisation ou avec des marquages aux sols) qui induisent une topologie stricte de route et permettent de restreindre l'espace navigable. Il est à noter que ces hypothèses permettent non seulement de réduire la dimension du problème de planification mais également d'améliorer les approches de prédiction de mouvement des autres véhicules. Cependant, les familles d'approches qui s'appuient sur un ensemble discret de manœuvres prédéfinies peuvent s'avérer infructueuses dans des situations de conduite où les interactions entre véhicules sont peu codifiées et s'effectuent de manière réactive.

Notre but est d'introduire une méthode permettant de calculer le mouvement du véhicule VA en évitant les collisions avec les autres véhicules dans un environnement semi-contraint. La stratégie retenue dans nos travaux consiste à séparer la planification du chemin et celle de la vitesse, en concevant un algorithme capable d'adapter continuellement la vitesse du VA en fonction des autres véhicules qui croisent son chemin. Nous supposons ici qu'il est préférable dans ce type d'environnement de réagir de façon « ad-hoc », en se basant sur des métriques de risque simples qui s'adaptent facilement à différentes situations.

Une première approche est présentée dans la section 2.4 où l'on considère le VA en interaction avec un autre véhicule dans des intersections simples non supervisées (carrefours giratoires et carrefours en X sans feux de signalisation). La méthode s'inspire du comportement de conducteurs humains évoluant dans le même environnement. Dans les situations étudiées, le VA n'a pas la priorité par rapport à l'autre véhicule, ce qui le force à prendre la décision de s'insérer lorsque c'est possible ou à céder le passage sinon.

Une généralisation de l'approche aux cas d'environnements semi-contraints est proposée dans la section 2.5. L'algorithme s'appuie sur un ensemble de profils de vitesse de référence compatibles avec les capacités du véhicule et le confort de ses passagers. Une modélisation MPC de la réponse du contrôle longitudinal ainsi qu'une métrique quantifiant le risque sont utilisées sur une prédiction à horizon fuyant, ce qui permet d'identifier la vitesse optimale à appliquer.

Des validations quantitatives ont été effectuées sur un grand nombre de scénarios en simulation, démontrant la stabilité et la capacité de l'algorithme à éviter les situations dangereuses. Un outil de simulation dédié a été conçu pour ces travaux, son développement est détaillé dans la section 2.3. Des validations qualitatives de l'approche, décrites dans la section 2.6, ont également été menées avec des véhicules réels sur le campus de l'Inria Rocquencourt.

2.2 État de l'art

Les différentes approches de la littérature en matière de planification de trajectoire pour un véhicule automatisé en présence d'autres véhicules mobiles sont présentées section 2.2.1. Cet état de l'art est suivi de la section 2.2.2 qui rappelle les méthodes de prédiction de trajectoire de véhicules. En effet, l'anticipation de mouvement est un prérequis nécessaire à la planification en présence d'obstacles mobiles. Enfin, la section 2.2.3 détaille et compare les principaux critères quantitatifs permettant de mesurer le risque d'une situation. Ces métriques se retrouvent dans la

plupart des fonctions de coût associées aux différentes méthodes de planification.

2.2.1 Approches en planification de trajectoire pour un véhicule automatisé

L'état de l'art de planification de trajectoire qui suit concerne essentiellement le mouvement du VA en situation d'interaction avec d'autres véhicules mobiles. La question de la planification de chemin optimale d'un véhicule en présence d'obstacles statiques n'est pas traitée, et fait d'ores et déjà l'objet d'articles de revue détaillés tels que [GONZALEZ et collab. \[2016\]](#). Les approches présentées par la suite ne traitent pas uniquement de scénarios urbains mais également de situations de conduite sur autoroute. En effet, toutes les approches de planification ne sont pas applicables à des intersections non supervisées, et a fortiori à des environnements semi-contraints où les manœuvres ne sont pas limitées par la topologie de la route. Nous considérerons ainsi toutes les méthodes de planification dynamique existantes avant de les comparer dans le Tableau 2.2.1.11 et de rappeler leurs éventuelles limitations dans le cadre des environnements semi-contraints.

2.2.1.1 Processus de décision markovien partiellement observable

Les POMDP (*Partially Observable Markov Decision Process*) sont une extension des chaînes de Markov. Un Processus de Décision Markovien (MDP) diffère d'une chaîne de Markov classique du fait que les transitions entre les états sont choisies par un agent, qui est récompensé en fonction de ses actions. Dans un POMDP, l'agent n'a qu'une information partielle de l'état courant : une même observation peut être faite dans plusieurs états, et à l'inverse, un état peut mener à différentes observations. Les différences entre les principaux modèles de Markov sont présentées dans le Tableau 2.1.

| | | Contrôle sur les transitions d'états? | |
|--------------------|-----|---------------------------------------|--|
| | | Non | Oui |
| États observables? | Oui | Chaîne de Markov | Processus de décision markovien (MDP) |
| | Non | Modèle de Markov caché (HMM) | Processus de décision markovien partiellement observable (POMDP) |

TABLEAU 2.1 – Les principaux modèles de Markov

Les approches de prise de décision d'un véhicule recourant aux POMDP permettent notamment de palier l'incertitude lors de la décision. Les auteurs de [BRECHTEL et collab. \[2014\]](#) rappellent que cette incertitude peut provenir :

1. du bruit sur les sorties de la perception
2. de l'occultation de la plupart des éléments de l'environnement
3. du caractère inconnu des intentions des autres conducteurs

Il proposent alors d'avoir recours à un POMDP continu, ce qui évite d'avoir à discrétiser a priori l'espace des états. Le POMDP continu se charge d'apprendre une bonne représentation de l'environnement durant la phase de convergence. Cette représentation, calculée une fois pour toutes « hors-ligne », est alors spécifique du scénario de conduite considéré et les informations statiques comme la topologie de l'intersection sont intrinsèques au modèle. Les décisions prises par le POMDP tiennent compte des différentes incertitudes et de l'observabilité partielle de l'environnement. Les transitions du MDP, responsables de la décision, dépendent directement de la loi de probabilité utilisée pour simuler le comportement des autres véhicules lors de la convergence.

Ainsi, il est difficile de restituer des interactions réalistes entre les véhicules sans avoir recours à des données réelles.

Les auteurs de [KYE et collab. \[2015\]](#) se concentrent particulièrement sur l'incertitude causée par l'intention des autres conducteurs. Ils introduisent un mécanisme « intention-aware » via des variables cachées qui tendent à modéliser l'intention des autres conducteurs. Peu de détails sont donnés sur la résolution du POMDP et la convergence de leur modèle, mais les auteurs mentionnent avoir effectué 100 scénarios de tests en conditions réelles sur un véhicule automatisé sans constater aucune collision.

Plus généralement, les intentions des conducteurs voisins sont souvent intégrées à la prise de décision via un modèle mixte MOMDP ([SEZER et collab. \[2015\]](#)) ou faisant intrinsèquement partie des états d'un modèle POMDP ([SONG et collab. \[2016\]](#), [HUBMANN et collab. \[2017\]](#), [GALCERAN et collab. \[2017\]](#), [BOUTON et collab. \[2017\]](#)). Ces travaux concernent des scénarios de changement de voie sur autoroute ou de traversée d'une intersection non supervisée. Dans [SEZER et collab. \[2015\]](#), les actions sont réduites à deux manœuvres consistant à s'arrêter ou à continuer, en négligeant totalement la dynamique du freinage du véhicule. Dans [GALCERAN et collab. \[2017\]](#), seule la décision de changer de voie est considérée, le profil de vitesse étant ajusté par un contrôle longitudinal qui suit le véhicule précédant à une distance fixe. Dans les autres approches, les actions disponibles forment un ensemble de deux ou trois valeurs d'accélération qui permet d'adapter le profil de vitesse. La métrique de risque très majoritairement utilisée au sein des fonctions de coût qui évaluent les états est le [laps de temps avant qu'une collision n'advienne, ou Time To Collision \(TTC\)](#), dont les sérieuses limites sont discutées en section 2.2.3. On remarque que ces fonction de coût (ou de récompense) sont des pondérations de différentes métriques dont les poids doivent être ajustés empiriquement par les auteurs, en fonction du type de scénario envisagé.

Deux de ces travaux proposent une validation comparant les performances du POMDP avec celles d'une heuristique simple suivant un seuil de TTC. Dans [SEZER et collab. \[2015\]](#) de nombreuses collisions sont constatées pour ces heuristiques lorsque le seuil est inférieur à 4s, contrairement aux performances du POMDP. Cependant il est possible que l'inertie du véhicule ne lui permette pas de s'arrêter instantanément comme le suppose la prédiction sans modèle dynamique, ce qui handicaperait fondamentalement l'heuristique tandis que la stratégie d'apprentissage du POMDP lui permettrait de compenser cette mauvaise modélisation. Dans [BOUTON et collab. \[2017\]](#), le POMDP est légèrement plus efficace que l'approche TTC naïve dans les cas les plus simples (virage à droite, avec une faible densité de véhicules), mais les deux approches semblent raisonnablement sûres. A l'inverse, le POMDP devient moins sûr et moins efficace que l'heuristique TTC sur les autres manœuvres (virage à gauche) et lorsque la densité de véhicules augmente.

2.2.1.2 Ontologies

Une ontologie est l'ensemble structuré des termes et concepts représentant la sémantique dans un champ d'informations. Elle permet de spécifier dans un langage formel les concepts d'un domaine (ex : individus, classes, attributs...) et leurs relations dans une sorte de grammaire propre. Dans le cadre de la prise de décision pour la conduite d'une voiture automatisée, les ontologies permettent de raisonner sur les interactions entre les différents usagers de la route de façon à extraire la manœuvre la plus appropriée. Les relations établies entre les différents usagers statiques ou mobiles et les autres objets de l'environnement sont utilisées afin d'inférer les comportements futurs des agents rationnels. Le raisonnement au niveau du comportement se fait souvent en termes de manœuvres ("s'arrêter" ou "continuer"), en fonction de conditions binaires ("s'il y a d'autres véhicules sur la voie", "s'il y a assez de place pour contourner sur la gauche"). Ces modélisations nécessitent la plupart du temps de présupposer que tous les agents en présence sont rationnels et agissent de manière prévisible. Les ontologies sont très efficaces pour réduire rapidement le nombre d'acteurs ou de paramètres à considérer dans une situation donnée afin de prendre une décision optimale. C'est la raison pour laquelle elles peuvent être combinées à une planification plus fine de la trajectoire une fois que la dimensionnalité du problème a été fortement réduite.

Le concept d'une ontologie adaptée aux interactions entre des véhicules est proposé par **REGELE [2008]**. Cette représentation permet notamment d'inclure la topologie de la route comme un réseau de segments accessibles. L'approche consiste à détecter les interactions critiques avec d'autres véhicules, afin de ne pas avoir à traiter toutes les informations disponibles. Cependant, la question de la prise de décision dans le cadre de ces interactions n'est pas traitée. Les auteurs de **ZHAO et collab. [2015a]** et **ZHAO et collab. [2015b]** utilisent des ontologies pour adapter la vitesse du VA au type de route sur laquelle il circule, en considérant la limite de vitesse légale et la largeur de la voie.

Dans **HULSEN et collab. [2011b]**, une ontologie permet de déduire l'ordre selon lequel plusieurs véhicules devraient passer à une intersection donnée en respectant strictement le code de la route. La méthode présuppose de connaître d'avance la destination de chaque véhicule. Une implémentation temps réel de cette approche est proposée dans **HULSEN et collab. [2011a]**, en vue de notifier le conducteur d'une situation trop dangereuse via un système d'aide à la conduite (ADAS). Dans **ARMAND et collab. [2014]** également, les véhicules sont supposés respecter parfaitement les règles de priorité. La modélisation considérée est réduite à une seule dimension longitudinale, le long du futur chemin suivi par le VA. L'approche permet notamment au véhicule d'anticiper les mouvements des autres véhicules en tenant compte de l'influence indirecte de l'environnement. Dans l'exemple mentionné, le VA anticipe que le véhicule qui le précède va devoir ralentir à l'approche d'un passage piéton car un piéton s'en approche et sera probablement en train de traverser lorsque le véhicule arrivera à sa hauteur.

2.2.1.3 Décomposition chemin vitesse

Certaines approches recourent à la décomposition chemin-vitesse introduite dans **KANT et ZUCKER [1986]** : la planification de trajectoire est alors réduite à une planification du profil de vitesse sur un chemin déjà défini en amont, en fonction de la topologie de la route. Cette stratégie revient à rechercher un profil de vitesse sur le plan chemin-temps, sur lequel les obstacles mobiles à éviter sont représentés par leur intersection avec le couloir du VA lors de leur passage sur son chemin.

Dans ce cadre, les auteurs de **JOHNSON et HAUSER [2012]** démontrent que les ensembles des vitesses atteignables par un véhicule dont la vitesse et l'accélération sont bornées sont toujours des intervalles convexes. Lors de l'évitement d'un obstacle, un de ces intervalles de vitesses peut se voir scindé au maximum en deux intervalles (le premier qui laisse passer l'autre véhicule, l'autre pour passer en premier). En partant de ces considérations, il est possible de construire un arbre dont les nœuds sont placés aux positions des obstacles rencontrés sur le chemin, et les branches représentent ces intervalles de vitesses. La complexité d'une recherche de chemin dans cet arbre est en $O(2^N)$ avec N le nombre d'obstacles sur le parcours du véhicule. Cependant la propriété convexe des intervalles de vitesse permet de fusionner une majorité des nœuds et de réduire drastiquement cette complexité en $O(N)$. La recherche d'un profil de vitesse dans cet arbre peut s'effectuer en $O(N^4)$, ce qui est validé expérimentalement.

Les mêmes auteurs généralisent cette approche pour effectuer un suivi de véhicule lorsqu'un obstacle avance sur le même chemin que le VA durant une durée indéterminée (cf. **JOHNSON et HAUSER [2013]**). Ils formalisent également le calcul de la forme exacte que prennent les obstacles dans l'espace chemin-temps (précédemment cette forme était approximée par un rectangle), afin de mieux rendre compte du type d'interaction avec les autres véhicules.

Dans **BENDER et collab. [2015]**, cette décomposition chemin-vitesse est étudiée dans le cadre d'un environnement dont la topologie est contrainte par des règles de conduite. Les trajectoires possibles permettant d'éviter d'entrer en collision avec les obstacles sont discrétisées en plusieurs classes de solutions : les manœuvres. L'aspect combinatoire de ces manœuvres est considéré comme une opportunité de réduire le domaine de recherche de la trajectoire optimale. Les auteurs soutiennent que cette classification en manœuvres permet de réduire le problème d'optimisation globale de la planification de trajectoire en une optimisation locale dont la convergence est assurée, grâce à la création d'une heuristique qui sélectionnerait en amont la classe de trajectoires

appropriée. Cependant les auteurs reconnaissent que la formulation d'une telle heuristique n'est pas triviale. Il est d'ailleurs à noter que dans beaucoup de cas de figure, les optimisations locales internes à chaque classe devront être systématiquement menées en amont afin de déduire la trajectoire globalement optimale. Le type de manœuvre auquel la trajectoire optimale appartient ne pourra être déduit avant d'effectuer ces multiples optimisations.

Cette explosion combinatoire des manœuvres possibles est soulignée dans [HUBMANN et collab. \[2016\]](#), ainsi que la difficulté de constituer une heuristique sélectionnant une manœuvre. La solution proposée par les auteurs est d'utiliser la décomposition chemin-vitesse dans un arbre de recherche pour effectuer une planification de vitesse compatible avec un calcul se faisant en temps réel. Les nœuds de l'arbre sont placés régulièrement toutes les 1s jusqu'à l'horizon temporel et possèdent chacun 4 branches représentant 4 valeurs d'accélération possibles. Différentes fonctions de coût sont combinées en vue de tenir compte de la sécurité, du confort et du temps de trajet associé à chaque nœud. L'algorithme A* est utilisé pour rechercher le profil de vitesse optimal. On peut relever que les 4 valeurs d'accélération retenues sont plutôt faibles ce qui doit rendre difficile le contrôle d'une insertion réactive devant un autre véhicule. Ce faible nombre de valeurs disponibles est néanmoins nécessaire pour réduire la complexité de l'arbre. De plus, le fait que la planification soit constituée d'une succession de valeurs d'accélération plutôt que d'un profil de vitesse sur le long terme ne permet pas de générer des comportements « naturels », qui respectent les conventions de conduite des humains. Par exemple, le comportement du véhicule cédant le passage à un autre sera de freiner faiblement en amont lorsqu'il détecte une interaction, même si celle-ci n'aura lieu qu'à la fin de l'horizon temporel. En général le freinage n'est pas immédiat s'il n'est pas nécessaire, et le comportement d'un conducteur humain serait plutôt de conserver plus longtemps sa vitesse afin de rester réactif si l'opportunité de s'insérer plus tôt se présente. Des évaluations de type qualitatif sont menées sur deux scénarios en simulation et lors d'un test sur véhicule automatisé.

D'une manière générale, les approches suivant la décomposition chemin-vitesse proposent des planifications de profil de vitesse complètes et optimales avec une résolution efficace (de complexité polynomiale notamment). Pour autant, seul l'évitement d'obstacle est considéré dans [JOHNSON et HAUSER \[2012\]](#) et [JOHNSON et HAUSER \[2013\]](#) : les notions de confort ou de règles de conduite ne sont pas prises en compte. Sur des scénarios plus réalistes, la complexité de résolution mentionnée dans [BENDER et collab. \[2015\]](#) est résolue par [HUBMANN et collab. \[2016\]](#) via une discrétisation des accélérations admissibles. Cependant, cette dernière ne permet pas d'anticiper la réponse dynamique du contrôleur du véhicule, ni de tenir compte de l'aspect réaliste des profils de vitesse planifiés par le véhicule.

Enfin, le cas d'un environnement avec des faibles contraintes sur la topologie de la route est traité dans [MAO et collab. \[2018\]](#). Une approche de planification de profil de vitesse est proposée pour une situation de sortie de péage d'autoroute, où le VA doit s'insérer sur une voie d'autoroute à partir de l'espace navigable très ouvert que constitue le péage, tout en évitant les collisions avec les véhicules qui l'entourent. Un arbre de recherche dont la profondeur dépend du nombre d'obstacles sur le chemin est construit, avec une discrétisation simple des accélérations admissibles (une positive et une négative). Le profil de vitesse le plus rapide qui permet d'éviter les collisions est sélectionné.

2.2.1.4 Planification avec commande prédictive

La commande prédictive (*Model Predictive Control*, ou *MPC*) est une technique de commande consistant à intégrer une modélisation interne du contrôleur dans le module en amont qui génère la consigne afin de minimiser les perturbations et les retards entre la consigne et la réponse effective du module de contrôle en aval. Dans le cas de la planification de trajectoire d'un véhicule, les approches *MPC* modélisent le plus fidèlement possible le contrôleur utilisé (contrôle longitudinal ou latéral du véhicule) et intègrent une simulation de la réponse du véhicule dans le processus de prédiction qui précède la prise de décision. Ces méthodes permettent d'anticiper et de compenser le futur comportement du véhicule lorsqu'une consigne d'entrée lui est fournie.

Dans NILSSON et SJOBERG [2013], la prise de décision lors de la conduite sur une autoroute à deux voies est principalement basée sur cette approche, en combinant un MPC avec différentes fonctions d'utilité pour éviter les collisions et maintenir la vitesse du véhicule. L'approche est testée sur 4 situations où le VA est considéré comme étant ponctuel et pouvant changer de voie instantanément. Les autres véhicules évoluent à vitesse constante sans changer de voie.

La modélisation MPC peut être aisément associée à différentes approches de planification afin d'améliorer leurs performances. Dans GOTTE et collab. [2015], la planification de trajectoire à partir d'une approche à bande élastique (*Timed Elastic Band*) est reformulée afin de pouvoir y appliquer un MPC non linéaire (NMPC). Le problème de planification revient alors à mener un contrôle optimal en tenant compte des contraintes dynamiques du modèle du véhicule. La validation est effectuée sur 2 scénarios de dépassement sur une autoroute à deux voies, et les valeurs de différentes métriques de confort (accélération, braquage, etc) sont discutées.

La méthode présentée dans RASEKHIPOUR et collab. [2017] combine une approche par champs de potentiel avec un MPC. La fonction de potentiel a l'avantage de pouvoir être modulée en fonction du type d'obstacle (notamment dans le cas d'un obstacle infranchissable comme un véhicule ou franchissable tel un dos d'âne) et permet donc d'adapter la planification de manière dynamique en priorisant les obstacles à éviter. Dans un des 7 scénarios de validation, un dos d'âne est contourné à vitesse constante par une manœuvre latérale. Dans un autre scénario où la présence d'un second véhicule empêche de changer de voie, le VA ralentit afin de passer sur le dos d'âne. L'intégration de la dynamique du véhicule dans le modèle de prédiction de trajectoire permet d'optimiser le chemin de consigne afin que le véhicule suive au mieux la force dérivant du potentiel artificiel.

Les auteurs de ZHANG et collab. [2018] combinent le contrôle optimal d'un MPC avec une planification stratégique inspirée de la théorie des jeux. Le VA avance sur une voie se terminant après quelques dizaines de mètres et doit donc s'insérer sur une voie parallèle où circulent trois autres véhicules. Le problème de planification de trajectoire est résolu à partir d'un jeu de Stackelberg entre le VA et un véhicule cible pour lequel il doit être décidé de s'insérer devant ou derrière. La prédiction de mouvement pour les autres véhicules est effectuée par des chaînes de Markov floues. Le rôle du MPC est d'assurer un contrôle bas-niveau optimal de la vitesse longitudinale en intégrant le modèle dynamique du véhicule. La composante latérale de la trajectoire est assimilée à une variable binaire qui ne représente que la voie sur laquelle le véhicule circule.

Dans GRAF PLESSEN et collab. [2018], une approche de *Linear Time-Varying Model Predictive Control (LTV-MPC)* avec contraintes spatiales est proposée dans le contexte de l'évitement d'obstacles sur une autoroute à deux voies. La dynamique du véhicule est assimilée à un modèle bicyclette non linéaire dont les entrées de contrôle comportent l'accélérateur, le frein, le volant et le boîtier de vitesses. Un couloir de navigation est défini en fonction de la voie du véhicule, de sa vitesse et des obstacles en présence. La trajectoire est planifiée par le MPC à partir de ce couloir, et dépend notamment de la vitesse imposée au véhicule et de ses contraintes dynamiques. Une replanification dynamique doit mettre régulièrement à jour cette trajectoire afin de permettre de circuler parmi des obstacles mobiles.

2.2.1.5 Champs de potentiel artificiel

Les travaux de REICHARDT et SHICK [1994] présentent pour la première fois une utilisation des champs de potentiel (*Artificial Potential Fields*, cf. 3.2.2.2) appliquée à la conduite réactive de véhicules. Dans GERDES et ROSSETTER [2001], la conception d'une fonction de potentiel dédiée permet d'inciter le véhicule à se centrer sur une voie lorsque plusieurs voies sont disponibles. Une thèse a par ailleurs été dédiée à l'utilisation des champs de potentiel en vue d'effectuer un suivi de voie (ROSSETTER [2003]). Les auteurs de WOLF et BURDICK [2008] étendent l'approche de suivi de voie à une conduite sur autoroute plus sûre. Ils introduisent notamment un mécanisme poussant le VA à éviter de rester sur la voie du véhicule qui le précède, en étirant son champ de potentiel vers l'avant en fonction de sa vitesse.

2.2.1.6 Bandes élastiques

Le concept de la planification de chemin par bandes élastiques (*Elastic Bands*) ainsi que la planification de trajectoire par une approche similaire (*Timed Elastic Bands*) proviennent du domaine de la navigation robotique. Cette approche adapte un chemin initial à la présence d'obstacles via une fonction de déformation locale du chemin, qui s'adapte aux modifications de l'environnement (cf. QUINLAN et KHATIB [1993]). L'approche des bandes élastiques est appliquée dans le cadre de la conduite de véhicules pour planifier des trajectoires qui évitent les collisions dans des situations de trafic dense. Les travaux de HILGERT et collab. [2003] et SATTEL et BRANDT [2005] mettent en place des manœuvres d'évitement d'obstacle en n'ayant recours qu'à la modification de l'angle de lacet du véhicule. En conséquence, ces approches ne permettent pas de moduler la vitesse du véhicule, celui-ci ne freine jamais y compris dans les situations urgentes. Les auteurs de ROESMANN et collab. [2012] ajoutent une composante temporelle à l'approche des bandes élastiques (*Timed Elastic Bands*), ce qui permet au véhicule d'éviter les collisions en combinant une déviation latérale du chemin et un ralentissement du véhicule. Dans KELLER et collab. [2014], l'approche des bandes élastiques temporelles est spécifiquement utilisée dans des situations de conduite critiques où le risque de collision est élevé.

2.2.1.7 Théorie des jeux

Un état de l'art de l'utilisation de la théorie des jeux dans le domaine de la conduite autonome de véhicule est établi dans ELVIK [2014]. Cette revue mentionne notamment une situation d'insertion sur l'autoroute où un agent doit décider du moment où il quitte sa voie d'accélération afin de rejoindre l'autoroute et d'autres agents ont le choix de faciliter ou non son insertion en ralentissant ou en changeant de voie (cf. KITA [1999]). Cependant, chaque agent a par ailleurs intérêt à maximiser sa vitesse et la situation impose donc une sorte de compétition modélisée par un jeu non coopératif à somme non nulle. Un scénario similaire est étudié dans YOO et LANGARI [2013] sous la forme d'un jeu en duopole asymétrique de Stackelberg entre l'agent qui cherche à s'insérer et le premier véhicule situé derrière ce dernier sur la voie principale.

Le cadre de la théorie des jeux peut également être utilisé dans des situations plus singulières, comme dans le choix d'utiliser les feux de croisement ou les feux de route la nuit à l'approche d'autres véhicules. Cette situation peut être formulée comme un dilemme du prisonnier où chaque agent a pour intérêt personnel de conserver ses feux de route le plus longtemps possible. Le fait que la conduite soit un jeu répété incite à une stratégie de coopération et pousse les agents à basculer en feux de croisement avant que le véhicule d'en face ne soit trop gêné.

Les auteurs de l'état de l'art reconnaissent que la théorie des jeux seule est largement insuffisante pour modéliser fidèlement le comportement des humains en situation de conduite. Cependant ils concluent qu'elle propose un cadre de travail adapté à l'étude des interactions entre agents rationnels et, de ce fait, gagnerait à être intégrée plus fréquemment dans des approches de prises de décision lors de la conduite. C'est également la conclusion de la thèse de MARTIN [2013] sur le sujet : l'équilibre de Nash calculé à partir de la formulation simplifiée d'une situation d'insertion sur une voie reflète globalement le comportement des conducteurs humains, on peut donc supposer qu'ils agissent de façon rationnelle afin de maximiser leur confort et de minimiser le risque de collisions. Cependant le recours à ce type d'approche est pour l'heure limité quant à la variété des situations de conduite traitées, et il n'est pas toujours aisé de formuler un scénario quelconque sous la forme d'un jeu.

2.2.1.8 Apprentissage profond

L'apprentissage par imitation (*Imitation Learning*) est une stratégie d'apprentissage permettant à un agent d'optimiser ses actions en tentant de reproduire des exemples de décisions prises par un expert dans un contexte similaire. C'est un type d'apprentissage supervisé dans lequel les inférences sont des décisions prises de manière séquentielle.

L'apprentissage par imitation est utilisé dans MORTON et KOCHENDERFER [2017] afin de réduire la dimensionnalité des données dans des états cachés et de s'en servir pour apprendre simultanément une politique de décision reposant sur cette couche cachée. Comme attendu, les états cachés du réseau de neurones réussissent à modéliser différentes classes de comportement du conducteur, à partir d'un jeu de données synthétique. Les politiques de décision sont optimisées pour reproduire les comportements donnés en exemple, conformément à l'approche utilisée, dite de « *Behavior Cloning* ». De fait, le principal défaut de cette approche est qu'elle génère des politiques pouvant être sous-optimales, voire contre-productives, lorsque le contexte dans lequel elles sont appliquées diffère trop du contexte dans lequel elles ont été apprises.

Dans KUEFLER et collab. [2017], une approche d'imitation plus complexe utilisant deux réseaux de neurones en compétition est proposée (*Generative Adversarial Imitation Learning*) et la politique est rendue récurrente (i.e. capable de raisonner à partir d'une séquence) via une couche récurrente *Gated Recurrent Unit* ajoutée au réseau de neurones. L'apprentissage est effectué grâce au jeu de données réelles NGSIM et les résultats de la validation montrent une amélioration par rapport à l'état de l'art en matière d'imitation du comportement humain, selon différentes métriques.

Une autre stratégie d'apprentissage permettant à un agent d'optimiser ses actions est l'apprentissage par renforcement (*Reinforcement Learning*). A la différence de l'*Imitation Learning*, le *Reinforcement Learning* n'apprend pas à partir d'exemples mais cherche à maximiser un score de récompense mis à jour en temps réel en fonction de ses actions (BOJARSKI et collab. [2016], SHALEV-SHWARTZ et collab. [2016], ISELE et collab. [2018]). Cette approche est dite semi-supervisée, car la supervision de l'apprentissage provient uniquement de l'attribution de ce score.

Deux paradigmes sont utilisés classiquement par les approches d'apprentissage par renforcement en fonction du type de données en entrée sur le réseau : le signal brut de perception (ex : les canaux RVB d'une image), paradigme dit *end to end* ou *behavior reflex*, ou alors une sémantique plus complète où l'entrée est une collection d'objets classifiés et localisés par un module de perception en amont : ce sont les approches dites *mediated perception*. Les auteurs de CHEN et collab. [2015] proposent un troisième paradigme utile dans le cadre de la conduite autonome par renforcement, qui utilise une sémantique intermédiaire : l'entrée *direct perception* consiste à extraire par un module de perception des descripteurs de haut niveau sans pour autant identifier formellement les objets de la scène.

Le *Inverse Reinforcement Learning* est une autre méthode de renforcement où la fonction de récompense doit également être apprise par le réseau. Cette approche est utilisée dans SADIGH et collab. [2016] où un véhicule autonome doit apprendre quelle stratégie de planification de trajectoire est la plus efficace afin d'influencer le comportement d'un autre véhicule conduit en simulation par un humain. Le réseau de neurone décide alors de se placer en travers de la route dans le but de forcer l'autre véhicule à utiliser une voie prédéterminée ou, à l'inverse, effectue une marche arrière à l'approche d'une intersection pour s'assurer de céder le passage à l'autre véhicule.

2.2.1.9 Décision coordonnée

Certaines approches de planification de trajectoire supposent une décision coordonnée et centralisée, qui est communiquée et respectée par tous les véhicules. Sur route cela supposerait en pratique d'interdire drastiquement l'accès aux conducteurs humains, ou à tout système qui ne respecterait pas un standard précis. Pour autant ce champ d'étude reste très intéressant pour des implémentations plus locales, comme la synchronisation d'un parc privé de véhicule sur un parking ou dans une exploitation minière par exemple. Dans DESARAJU et collab. [2009], une planification de trajectoires coordonnées pour deux véhicules est proposée afin d'éviter les collisions sur deux scénarios : sur un rond-point et sur une voie d'autoroute. La planification est testée sur un banc d'essai avec deux voitures miniaturisées. D'autres approches proposent une solution de coordination de trajectoires pour plusieurs véhicules passant sur une intersection non supervisée. Les auteurs de ALTCHÉ et collab. [2016] et de GUN et collab. [2018] formulent le problème comme une optimisation linéaire en nombres entiers mixtes (*Mixed-Integer Linear Program*, ou MILP) qui

tient compte des contraintes cinématiques des véhicules. L'approche est validée avec une simulation « microscopique » (ex : via le logiciel SUMO), sur des scénarios d'intersection comportant plus de 15 véhicules.

2.2.1.10 Analyse du profil de vitesse des conducteurs humains

Certains travaux se fondent sur l'analyse du comportement humain lors de la conduite afin d'améliorer la prise de décision pour un véhicule automatisé. Dans [QUINTERO M. et collab. \[2012\]](#) et [LEE et JANG \[2017\]](#), un partitionnement des séquences de conduite est effectué afin de détecter les caractéristiques des situations dangereuses. L'analyse est faite à partir des données séquentielles de vitesse, accélération et d'angle de braquage des véhicules. Les scénarios étudiés se déroulent sur autoroute ou sur un circuit qui ne nécessite pas de tenir compte d'un contexte particulier au niveau de la topologie de la route. La première approche consiste en une classification binaire (bonne ou mauvaise conduite) via un apprentissage supervisé sur un jeu de données annotées. Dans la seconde approche, c'est un apprentissage non supervisé qui est effectué avec un auto-encodeur épars sur un jeu de données de trajectoires de conducteurs de taxis. Comme l'indiquent en conclusion les auteurs, les situations sont ainsi groupées selon des caractéristiques comme l'intensité de l'accélération ou le profil de vitesse, qui ne sont pas forcément toujours corrélés avec la dangerosité effective de la situation. D'autres méthodes d'analyse de profil de vitesse des conducteurs humains sont décrites dans l'article de revue [MEIRING et collab. \[2015\]](#).

Certaines approches analysant les profils de vitesse des conducteurs humains permettent de générer un profil de vitesse applicable par un véhicule automatisé. C'est le cas dans [SAMA et collab. \[2018\]](#), où un chemin prédéfini peut être partitionné en différentes classes de comportement à adopter en se basant sur une carte détaillée de la topologie de la route correspondante. Un auto-encodeur (profond et densément connecté) génère une classification en 9 comportements différents à partir des entrées du volant, de l'accélérateur, des freins et de la vitesse des véhicules. L'apprentissage non supervisé se fait à partir d'un jeu de données de trajectoires exécutées par des moniteurs d'auto-école. Pour chaque classe de comportement, une trajectoire est simulée en utilisant l'auto-encodeur comme modèle génératif. Les résultats montrent que les profils de vitesse générés ressemblent raisonnablement à ceux des conducteurs humains, y compris sur des parcours de validation où le modèle génératif n'a pas été entraîné.

Dans [CHEUNG et collab. \[2018\]](#), un partitionnement des comportements en différentes classes permet de générer des profils de vitesse réalistes pour des scénarios sur autoroute. L'approche utilise en entrée les distances et vitesses relatives des véhicules voisins afin de tenir compte des interactions dynamiques. Les auteurs introduisent la métrique de haut niveau *Trajectory to Driver Behavior Mapping* (TDBM), qui permet une bonne linéarisation des comportements entre agressif et prudent. Cette métrique correspond au premier vecteur propre d'une ACP effectuée sur les données. Une approche simple de planification de trajectoire est proposée. Elle consiste à éviter les conducteurs identifiés comme dangereux en privilégiant un changement de voie lorsque le TDBM d'un véhicule voisin dépasse un certain seuil.

2.2.1.11 Tableau récapitulatif

Le tableau synoptique qui suit compare les 8 principales approches présentées ci-dessus, sur la base des 4 critères suivants :

1. *Environnement semi-contraint* : l'approche peut-elle être utilisée dans un environnement avec peu de contraintes en matière de topologie de la route ?
2. *Interactions avec plusieurs véhicules* : l'approche permet-elle de traiter un contexte très dynamique où le VA croise les trajectoires de nombreux autres véhicules ?
3. *Qualité de la trajectoire* : la trajectoire planifiée peut-elle être suivie par un modèle réaliste de véhicule ? Est-il possible de prendre en compte des critères de confort ?
4. *Nécessite un jeu de données* : l'algorithme nécessite-t-il un apprentissage sur des données annotées de trajectoires sur des environnements semi-contraints ?

Deux approches présentées précédemment ne sont pas incluses dans le tableau. En effet, les approches reposant sur une décision coordonnée entre tous les véhicules permettent de trouver une solution optimale, mais sont spécifiques à des scénarios où il est possible de centraliser la décision. Par ailleurs, les approches de *MPC* permettent d'optimiser la réponse du contrôleur en anticipant son action via une modélisation interne, mais ne sont pas en soi des planifications de trajectoire et doivent être intégrées avec une autre stratégie.

Dans le Tableau 2.2, les cellules en rouge indiquent un défaut rédhibitoire pour que l'approche en question puisse être utilisée dans le contexte d'un environnement semi-contraint. Souvent, cela concerne les approches nécessitant un jeu de données ou une modélisation précise des autres véhicules dans un environnement semi-contraint, prérequis qui n'est pas disponible à l'heure actuelle. Les cellules en orange indiquent un défaut notable qui n'empêche cependant pas le recours à l'approche. Au vu des limitations actuelles de la plupart de ces approches, on remarque qu'il est préférable pour l'heure de recourir aux méthodes de décomposition chemin-vitesse pour pouvoir traiter la planification de trajectoire dans un environnement semi-contraint.

| | Environnement semi-contraint | Interactions avec d'autres véhicules | Qualité de la trajectoire | Jeu de données |
|---------------------------------------|---|--|--|---|
| POMDP | Compatible | Approfondit les interactions entre les véhicules | Dépend de la méthode de planification utilisée | Modélisation réaliste du comportement des autres véhicules |
| Ontologie | Nécessite une topologie de route et un contexte sémantique fort | Approfondit les interactions entre les véhicules en utilisant le contexte sémantique | Dépend de la méthode de planification utilisée | Non |
| Apprentissage Profond | Compatible | Nécessite une modélisation réaliste du comportement des autres véhicules | Approches réactives, encore peu matures en matière de qualité de trajectoire | Approches par imitation, inutilisables sans données d'apprentissage |
| Champ de potentiel | Compatible | Approche réactive, ne tient pas compte explicitement des interactions | Approche robotique, qualité du chemin trop faible pour un véhicule | Non |
| Bandes élastiques | Compatible | Approche réactive, ne tient pas compte explicitement des interactions | Approche robotique, qualité du chemin trop faible pour un véhicule | Non |
| Théorie des jeux | Mouvement décomposé en manœuvres, contexte autoroutier | Prise en compte des intentions des autres véhicules, interactions complexes | Dépend de la méthode de planification utilisée | Non |
| Analyse du comportement humain | Compatible | Difficile prise en compte de l'aspect dynamique du contexte | Qualité de la trajectoire au cœur de ces approches | Profils de vitesse |
| Décomposition chemin / vitesse | Compatible | Planification du profil de vitesse possible en présence d'une flotte de véhicules | L'hypothèse de chemin statique semble proche du comportement naturel d'un véhicule | Non |

TABLEAU 2.2 – Tableau récapitulatif des approches de planification de trajectoire pour un véhicule automatisé

2.2.2 Prédiction de mouvement

Les approches de planification dynamique de trajectoire reposent toutes sur une modélisation des mouvements futurs des autres véhicules. Cette section présente donc un état de l'art sur les méthodes de prédiction de trajectoires, puis discute de la possibilité d'y recourir dans le cadre d'un environnement semi-contraint.

Une revue très complète de la littérature sur la prédiction de mouvements des véhicules a été effectuée dans [LEFÈVRE et collab. \[2014\]](#). Les auteurs distinguent trois grandes catégories d'approches de prédiction (cf. Fig. 2.1) :

1. les prédictions basées sur une extrapolation physique
2. les prédictions basées sur les manœuvres
3. les prédictions basées sur les interactions entre les usagers de la route

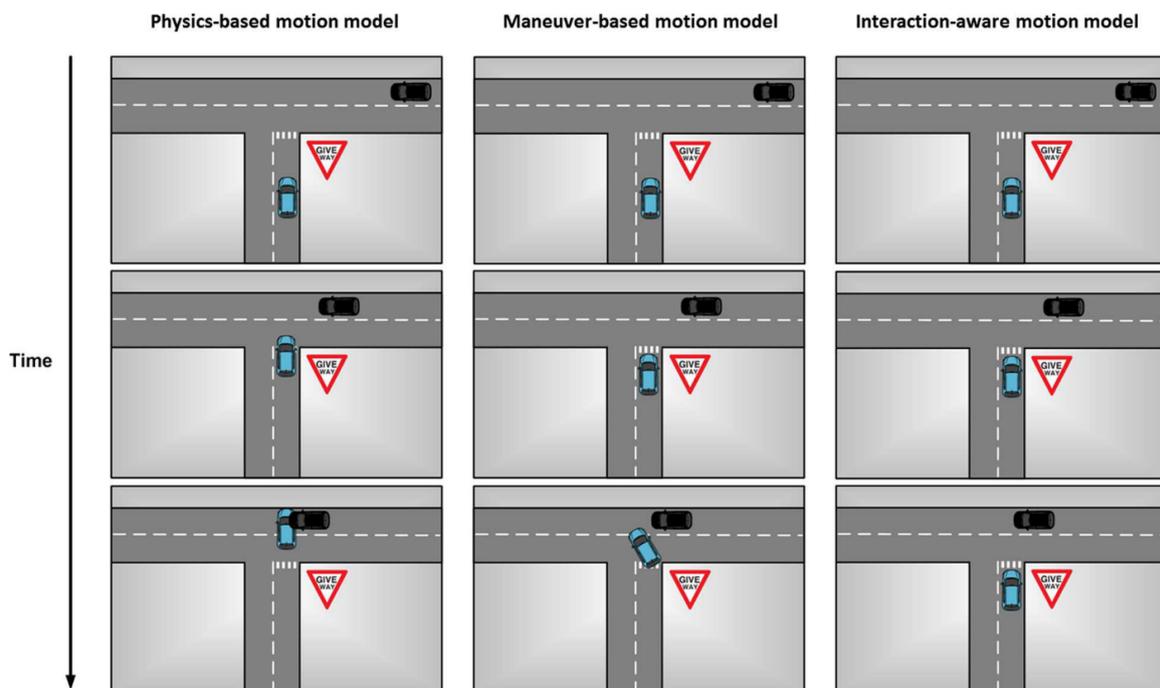


FIGURE 2.1 – Les différents types de modélisation du mouvement (Fig. issue de [LEFÈVRE et collab. \[2014\]](#)).

Cependant afin d'obtenir une partition exhaustive de la littérature nous proposons de les regrouper comme suit :

1. les prédictions basées sur une extrapolation physique
2. les prédictions tenant compte du contexte statique (topologie de la route, marquage au sol)
3. les prédictions tenant compte du contexte dynamique (présence des autres véhicules, prédiction de leurs intentions)

Ce qui permet d'inclure des approches qui tiennent compte du contexte statique sans pour autant être basées sur des manœuvres, ainsi que les récentes approches en apprentissage machine qui tiennent compte du contexte dynamique sans expliciter les interactions entre les véhicules.

Il est à noter qu'il est difficile de comparer ces approches, quand bien même elles s'appliqueraient dans des scénarios de même type. En effet, l'absence d'un benchmark de test reconnu par la communauté scientifique a pour conséquence des validations indépendantes des approches de prédiction par leur concepteurs, sur des jeux de données différents. Le projet CommonRoad ([ALTHOFF et collab. \[2017\]](#)) a justement eu pour objet de créer des scénarios types sur un simulateur en ligne, qui permet d'exécuter différentes approches de prédiction et de planification afin

de les comparer dans le même contexte. Les environnements de conduite concernés se situent principalement sur autoroute, certains ont lieu sur de larges intersections contrôlées par des feux de signalisation. Les véhicules obstacles suivent des trajectoires pré-enregistrées, extraites du jeu de données NGSIM (cf. COLYAR et HALKIAS [2007]).

2.2.2.1 Prédictions basées sur une extrapolation physique

Dans le cadre de la planification de trajectoire, les modèles de prédiction basés sur la physique sont pour la plupart des approches cinématiques (cf. LEFÈVRE et collab. [2014]), qui offrent des approximations raisonnables sur le mouvement des véhicules. Ces modèles cinématiques simulent l'évolution d'une trajectoire à partir d'un état initial, et peuvent être déterministes ou stochastiques.

Différents modèles cinématiques déterministes sont détaillés dans SCHUBERT et collab. [2008]. Parmi eux, les plus courants sont :

1. Mouvement rectiligne uniforme, où la vitesse longitudinale du véhicule est supposée constante (*Constant Velocity*)
2. Mouvement rectiligne avec accélération longitudinale supposée constante (*Constant Acceleration*)
3. Mouvement circulaire uniforme, où l'angle de braquage est supposé constant (*Constant Turn Rate and Velocity*)
4. Mouvement circulaire avec accélération longitudinale constante, qui est une combinaison des deux précédents (*Constant Turn Rate and Acceleration*)

Ces modèles peuvent également être utilisés pour modéliser des densités de probabilité sur les positions et vitesses afin de mieux rendre compte des incertitudes de mesure des capteurs. Une méthode simple consiste à ajouter un bruit gaussien sur un modèle déterministe (cf. AMMOUN et NASHASHIBI [2009]), ce qui produira une prédiction similaire à celle fournie par un filtre de Kalman. Une autre approche est d'utiliser la méthode de Monte-Carlo en effectuant des tirages sur certaines variables d'entrée (variation de l'angle de braquage, accélération, déviation latérale...) afin de générer une distribution. Il est également possible de filtrer les états obtenus afin d'appliquer certaines hypothèses, comme le fait que les véhicules évitent les collisions ou restent dans les limites imposées par le marquage au sol (cf. EIDEHALL et PETERSSON [2008], BROADHURST et collab. [2005]).

2.2.2.2 Prédictions incluant le contexte statique

Des modèles stochastiques infèrent la trajectoire future d'un véhicule à partir d'une séquence de positions passées. Ces approches (VASQUEZ et FRAICHARD [2004], VASQUEZ et collab. [2004]) utilisent une large base de données constituée des différentes manœuvres qui ont été observées. La trajectoire du véhicule est classifiée en lui attribuant une probabilité par manœuvre. Pour cela, les auteurs de VASQUEZ et FRAICHARD [2004] partitionnent les trajectoires afin de les regrouper selon des motifs de base. L'approche décrite dans HERMES et collab. [2009] utilise une recherche de plus longue sous-séquence commune sur les trajectoires, la méthode *Quaternion-based Rotationally Invariant LCS (QRLCS)*. Ces approches requièrent néanmoins de constituer une base de données de taille très conséquente afin de couvrir les cas de figures possibles. Elles ne tiennent pas compte de la situation du véhicule par rapport à un contexte sémantique (par exemple, sa position sur une route) ou des interactions possibles avec les véhicules voisins.

D'autres méthodes infèrent les manœuvres futures d'un véhicule en se basant principalement sur la topologie de route. Pour cela des modèles de Markov cachés peuvent être utilisés, notamment pour inférer le chemin que prendra un véhicule à une intersection (STREUBEL et HOFFMANN [2014]). Le même problème est étudié dans LIEBNER et collab. [2013] à partir des profils de vitesse

seuls et un réseau bayésien simple. Une classification par machine à vecteurs de support (*Support Vector Machine*, SVM) couplée à un filtre bayésien peut également prédire les intentions de changement de voie sur autoroute (KUMAR et collab. [2013]).

Une carte de l'environnement peut être utilisée pour guider la prédiction de chemin d'un véhicule en fonction du contexte statique. A cet égard, dans PETRICH et collab. [2013] la topologie de route a été modélisée sous la forme de variables aléatoires multidimensionnelles qui sont intégrées dans un filtre de Kalman étendu. Cette topologie peut également être représentée sous forme de graphe sur lequel les chemins les plus probables au vu du profil de vitesse du véhicule sont identifiés via une classification ascendante hiérarchique (cf. QUEHL et collab. [2018]).

Des méthodes d'apprentissage en ligne sont également utilisées (VASQUEZ et collab. [2009], GOVEA [2010] et GRAF et collab. [2014]). Ces approches sont capables d'étendre leur modélisation stochastique en ligne, ce qui leur permet d'apprendre progressivement de nouveaux motifs de trajectoires y compris durant la phase d'inférence. Les deux premiers travaux portent sur des modèles de Markov cachés (*Growing Hidden Markov Models*). Dans GRAF et collab. [2014], ils utilisent l'apprentissage machine en ligne afin de prédire la trajectoire des véhicules aux intersections. Ils étudient notamment les effets du changement des obstacles statiques de l'environnement sur le style de conduite adopté afin d'améliorer la prédiction de mouvement des véhicules.

Dans GUNNARSSON et collab. [2006] le modèle de prédiction inclut l'intention du conducteur afin de prédire la trajectoire du véhicule sur le plus long terme. Cet état interne est estimé via un filtre à particules et une classification prédéterminée des manœuvres possibles sur autoroute. Le style de conduite du conducteur est également inféré à partir des observations passées. Ces travaux sont étendus dans SORSTEDT et collab. [2011] afin de pouvoir inclure les incertitudes de mesures dans les paramètres de la fonction de coût. A partir de ces états internes, il est possible de prédire plus précisément la trajectoire future des véhicules.

2.2.2.3 Prédiction incluant le contexte dynamique

D'autres travaux incluent les interactions entre les véhicules dans le modèle de prédiction, afin d'anticiper les futures manœuvres des autres véhicules en tenant compte des éventuelles rétroactions. Dans LAWITZKY et collab. [2013], l'influence mutuelle que les conducteurs ont entre eux est formulée pour un ensemble restreint de manœuvres sur autoroute, avec une résolution en trois étapes. Tout d'abord, une répartition des probabilités pour chaque manœuvre est donnée a priori. Ensuite la probabilité de collision pour chaque paire de manœuvres entre deux véhicules est calculée. Enfin, la répartition des probabilités des manœuvres est mise à jour en formulant l'hypothèse que chaque conducteur évite les manœuvres associées à une probabilité de collision trop élevée. Les 9 manœuvres disponibles correspondent aux changements de voie sur l'une des 3 voies parallèles et à 3 profils de vitesse (freinage, conservation de la vitesse et accélération) sur chacune d'entre elles. Présupposer que les conducteurs évitent d'effectuer une manœuvre trop risquée semble en effet pertinent, et s'avère très efficace pour restreindre l'ensemble des combinaisons possibles. Cependant, on peut également critiquer cette hypothèse précisément car elle sous-estime le risque, c'est à dire la probabilité qu'un autre véhicule effectue une manœuvre dangereuse.

Cette interaction entre les conducteurs des différents véhicules peut également être modélisée via un réseau bayésien dynamique (*Dynamic Bayesian Network*). Dans GINDELE et collab. [2010], un filtre de haut niveau est introduit, permettant de lisser et d'extrapoler les positions détectées des véhicules via des variables aléatoires. Chaque véhicule est décrit par son vecteur d'état, des descripteurs contextuels par rapport aux autres véhicules, une classification symbolique de la situation dans laquelle il se trouve, la manœuvre correspondante qu'il doit accomplir, et la prédiction de trajectoire qui en résulte. Les auteurs indiquent que les relations entre ces variables aléatoires sont modélisées manuellement, sans donner les détails de leur implémentation. Les résultats démontrent une bonne précision du filtre, mais le nombre et la variabilité des scénarios de test ne sont pas mentionnés. Une approche similaire est proposée dans LEFÈVRE et collab. [2012], où le risque à une intersection est directement évalué à partir des manœuvres disponibles pour

chaque véhicule, sans nécessiter la génération des prédictions de trajectoires futures. Le mouvement des véhicules est modélisé par la répartition de probabilité des manœuvres applicables sous la forme d'un réseau bayésien dynamique. Pour chaque véhicule, son intention (i.e. la manœuvre en cours, identifiée à partir du mouvement du véhicule) peut être comparée avec ce qu'il devrait faire (*expectation*) compte tenu du code de la route. Chaque véhicule a le choix entre deux manœuvres applicables : traverser l'intersection en suivant son chemin, ou s'arrêter pour céder le passage. Le risque de la situation est ensuite inféré à partir de cette comparaison via un filtre à particules.

On peut également constater depuis quelques années la prépondérance des approches d'apprentissage profond dans les travaux qui cherchent à anticiper la trajectoire future des véhicules. Les réseaux de neurones récurrents (*Deep Recurrent Neural Network*) sont notamment utilisés, afin d'intégrer l'aspect temporel des trajectoires dans la structure du réseau de neurones. Le type de neurone LSTM (*Long Short-Term Memory*) est le plus couramment utilisé dans ces architectures (XIN et collab. [2018], ALCHÉ et DE LA FORTELLE [2017]). À l'inverse, certains travaux utilisent à dessein un réseau de neurones à propagation avant (*Feedforward Neural Network*) vérifiant la propriété de Markov afin de pouvoir mieux l'étudier et d'en extraire les caractéristiques, par exemple en générant un GMM (*Gaussian Mixture Model*) dont les paramètres sont ajustés par le réseau de neurone (LENZ et collab. [2017]).

Les travaux les plus récents tendent à inclure la notion d'interaction entre les véhicules au sein d'une structure de réseau de neurones récurrents. Certaines approches modélisent un état interne pour chaque véhicule avec une couche LSTM (KIM et collab. [2017]) afin d'anticiper les interactions entre le véhicule ego et chaque autre véhicule, pris deux à deux. Un mécanisme de pooling localisé permet à des couches LSTM simulant chacune la trajectoire d'un agent d'utiliser uniquement les informations concernant ses proches voisins. Social LSTM a été initialement introduit afin de prédire les trajectoires de piétons en interaction dans une foule dense (ALABI et collab. [2016]). Le *Social Pooling* a ensuite été utilisé sur des véhicules. Dans un scénario sur autoroute la prédiction de trajectoire pour chaque véhicule est effectuée conjointement avec celles de ses voisins proches (DEO et TRIVEDI [2018]). Les auteurs de MESSAOUD et collab. [2019] introduisent un mécanisme d'attention et le combinent avec le *Social Pooling* pour saisir l'importance relative de chaque véhicule voisin, proche ou non du véhicule ego, dans la prévision du mouvement. Une autre stratégie pour considérer le voisinage de chaque véhicule est d'utiliser un réseau de convolution sur une représentation de la scène vue de haut. Les auteurs de JEON et collab. [2018] introduisent une grille d'occupation *Multi-Channel Occupancy Grid Map* en entrée d'un réseau de neurones récurrent combinant des couches de convolution avec des couches LSTM. La convolution permet de réduire la dimensionnalité de la grille d'occupation tout en résonnant de façon spatiale.

On peut remarquer que le principal facteur limitant ces approches *Deep Learning* est la rareté de bases de données de bonne qualité sur des trajectoires de véhicules, de taille suffisamment conséquente, sur des scénarios variés, et annotées avec une vérité terrain. À cet égard, les publications des jeux de données NGSIM (COLYAR et HALKIAS [2007]) et highD (KRAJEWSKI et collab. [2018]) de scénarios sur autoroute ont rapidement donné suite à l'apparition dans la littérature d'approches inédites pour l'inférence de trajectoires de véhicules. On constate l'absence notable de base de données de qualité sur des scénarios urbains d'intersections non supervisées, ce qui est sans doute la principale raison pour laquelle aucun modèle RNN efficace n'a encore été proposé dans ces situations.

2.2.2.4 Discussion sur les environnements semi-contraints

D'après LEFÈVRE et collab. [2014], le seul inconvénient des approches les plus complexes qui incluent les interactions entre les véhicules est qu'elles nécessitent une grande puissance de calcul, et peuvent être difficiles à implémenter sur un système temps réel. Or ces prédictions sophistiquées présupposent également que certaines hypothèses de départ soient respectées :

1. que le contexte du scénario soit bien identifié : en effet, plus la prédiction est sophistiquée plus elle est sensible à la détection des différents éléments de l'environnement.

2. que le jeu de données pour l'apprentissage ne soit pas biaisé : il doit être suffisamment conséquent et équilibré pour éviter que la modélisation des trajectoires courantes ne soit pas centré sur la moyenne des comportements en situation réelle.
3. que la variance des trajectoires ne soit pas sous-estimée : le modèle doit pouvoir raisonnablement gérer des situations spécifiques (e.g. un comportement inattendu de la part d'un conducteur), sans présupposer que les conditions de conduite du jeu de données sont toujours valables en situation réelle.

Pour revenir à la classification en 3 types de prédictions, les approches qui infèrent l'intention des autres véhicules nécessitent une modélisation adaptée des autres conducteurs, ce qui n'est pas forcément possible dans toutes les situations de conduite. Parmi elles, les approches basées sur de l'apprentissage en particulier requièrent des bases de données conséquentes qui ne sont pas toujours accessibles. Les approches qui incluent le contexte statique requièrent un environnement connu dans lequel un ensemble limité de manœuvres peut être établi d'avance. Ces hypothèses ne sont pour la plupart jamais validées dans les environnements de type semi-contraints que l'on aborde ici.

Durant ce doctorat, on s'est efforcé de constituer une base de données des trajectoires de véhicules en interaction dense sur un environnement semi-contraint à partir de la place de l'Étoile à Paris. Une captation vidéo a été effectuée plusieurs heures durant depuis le haut de l'Arc de Triomphe. Au vu de la densité de véhicules sur ce rond-point on peut estimer que plus de 5000 trajectoires différentes ont été captées (i.e. passages d'un véhicule circulant en moyenne 30s sur le rond-point, avec 60 images par seconde). Le travail en traitement d'image effectué afin d'isoler les trajectoires n'a pas pu aboutir à un jeu de données suffisamment annoté pour pouvoir être utilisé et entraîner un réseau de neurones.

En l'absence de base de données adaptée pour pouvoir traiter la prédiction de mouvement dans un environnement semi-contraint à partir d'une approche de type machine learning, une prédiction plus simple s'appuyant sur une extrapolation physique du mouvement a été utilisée. La plupart du temps, cette prédiction de mouvement est plus conservatrice que les approches basées sur des manœuvres, car ces dernières peuvent anticiper les règles de priorité afin d'en tirer profit pour mieux insérer le véhicule autonome dans la circulation. Mais il est impossible d'utiliser ces approches dans un environnement où les règles de priorité sont floues et où on ne peut réduire la trajectoire d'un véhicule à un ensemble de manœuvres. Une planification de trajectoire fonctionnelle peut néanmoins être conçue pour un tel environnement à partir d'une prédiction physique, à condition que la planification soit suffisamment réactive pour compenser rapidement le manque d'anticipation de ce type de prédiction.

2.2.3 Métriques d'évaluation du risque

Un des principaux critères définissant le succès d'une planification de trajectoire est de pouvoir éviter les situations qui présentent un trop grand risque de collision avec un obstacle.

Cette section décrit et compare les principales métriques permettant de quantifier ce risque, d'abord d'une façon générale, puis dans des situations de conduite plus spécifiques.

2.2.3.1 Présentation de métriques courantes

L'évaluation du risque consiste à quantifier le risque de collision entre le VA et un véhicule obstacle à chaque pas de temps de la prédiction sur l'intervalle de temps $[t, t+T_H]$. Trois métriques principales peuvent être retrouvées dans la littérature afin de quantifier le risque d'une situation de conduite sur une intersection :

1. Le TTC (*Time To Conflict*, ou *Time To Collision*)
2. Le TG (*Time Gap*, also called *Time Headway*)
3. Le PET (*Post Encroachment Time*, ou *Trailing Buffer*)

Le TTC entre deux véhicules $V1$ et $V2$ est le laps de temps restant avant qu'une collision ne se produise, en considérant que les deux véhicules conservent une vitesse constante. Le TTC ne peut donc être défini que lorsque le véhicule de devant ($V1$) est plus lent que celui qui le suit ($V2$). $TTC = \frac{dist(V1,V2)}{vitesse_{V2}-vitesse_{V1}}$. Lorsque les chemins des véhicules se croisent, le TTC est seulement défini en cas de collision future, c'est à dire si $\frac{dist(V1,zoneDeCroisement)}{vitesse_{V1}} = \frac{dist(V2,zoneDeCroisement)}{vitesse_{V2}}$. Dans le cas contraire le TTC n'est pas défini, même pour des trajectoires faisant passer les véhicules très près l'un de l'autre à des vitesses élevées, ce qui traduit pourtant bien une situation dangereuse. C'est là la principale faille de cette métrique, comme expliqué en détails dans [VOGEL \[2003\]](#).

Le **intervalle de temps entre les trajectoires, ou Time Gap (TG)** entre deux véhicules est défini lorsqu'un véhicule $V1$ est situé sur le chemin d'un autre véhicule $V2$ et devant celui-ci. C'est le temps que nécessite $V2$ pour atteindre la position initiale de $V1$ en suivant un profil de vitesse supposé connu. Dans le cas où les deux futurs chemins se croisent comme à une intersection, un TG minimum peut être calculé à partir des deux trajectoires prédites. En cas d'intersection, la valeur du TG minimum est toujours définie et mesure le temps qui a séparé les passages des véhicules $V1$ et $V2$ à l'intersection. Étant donné qu'une valeur de TG est définie pour chacune des positions de $V1$ en contact avec le futur chemin de $V2$, le TG minimum permet de ne retenir qu'une seule valeur pour quantifier le risque maximum causé par l'interaction entre les deux trajectoires. A notre connaissance, cette métrique n'est que très rarement utilisée pour mesurer le risque aux intersections.

Le **décalé après croisement, ou Post-Encroachment Time (PET)** est une métrique créée spécifiquement pour les intersections en T ou en X. Il est défini comme étant le décalé entre l'instant auquel le premier véhicule sort de l'intersection et l'instant auquel le second y entre. Ainsi, un cas de figure où $PET = 0s$ est censé correspondre à une collision, ou à une situation qui s'en approche. Plusieurs travaux recommandent nettement l'utilisation du PET pour évaluer les situations de conduite sur intersection ([VOGEL \[2003\]](#), [BÄRGMAN et collab. \[2015\]](#)).

2.2.3.2 État de l'art des métriques de risque en situation de suivi

De nombreux travaux concernent les métriques de risque pour des scénarios longitudinaux où les véhicules se suivent. Ces scénarios ont principalement lieu sur autoroute, et se distinguent notamment par la limitation des manœuvres possibles : suivi de voie, changements de voie lors d'un dépassement, éventuellement insertion depuis une bretelle d'accélération.

Les systèmes régulateurs d'allure et d'espacement (*Adaptive Cruise Control*, ou ACC) utilisent couramment un TG constant pour adapter la vitesse d'un véhicule qui en suit un autre. Dans [WANG et RAJAMANI \[2004\]](#), les auteurs étudient les limites de cette métrique en terme de stabilité de la fluidité du trafic. Sur autoroute, les scénarios de jonction de la voie principale avec une voie d'accélération ou de sortie sont particulièrement problématiques. Dans [POLYCHRONOPOULOS et collab. \[2004\]](#), le TG est remplacé par de nouvelles métriques introduites pour prendre en compte une prédiction de trajectoire future pour les autres véhicules. Le *predicted minimum distance* (PMD) et le *time to predicted minimum distance* (TPMD) sont calculés en fonction des futures positions prédites pour les deux véhicules. Le PMD est la distance minimale entre les deux véhicules durant l'horizon temporel considéré, et le TPMD est le laps de temps restant avant que le PMD soit atteint. D'après les résultats montrés, le TPMD est bien plus performant que le TTC afin d'anticiper l'entrée d'un véhicule sur l'autoroute. Cependant, on peut remarquer trois éléments clés qui différencient les métriques TPMD et TTC :

1. Le TPMD est mesuré entre le VA et chaque autre véhicule, tandis que le TTC prend pour cible un seul autre véhicule, en l'occurrence celui qui est directement situé devant le VA.
2. Le TPMD est calculé en considérant les positions des deux véhicules à chaque pas de temps jusqu'à l'horizon temporel, tandis que le TTC semble être calculé directement depuis la vitesse instantanée relative du VA par rapport à celle du véhicule cible.
3. Enfin, le TPMD considère la distance minimale entre les deux véhicules et le TTC mesure le laps de temps restant avant qu'ils n'entrent en collision.

Les auteurs ne distinguent pas ces trois propriétés, mais on peut présumer que seule la première suffit à rendre le TTC inopérant dans le scénario abordé en simulation. De même, le TTC minimum calculé à chaque pas de temps jusqu'à l'horizon temporel serait plus précis que le TTC instantané dans le cas où les vitesses des véhicules ne sont pas constantes. Enfin, les auteurs de [JIMÉNEZ et collab. \[2013\]](#) proposent un calcul en 2D plus précis du TTC en considérant la boîte englobante rectangulaire de chaque véhicule, au lieu d'une approximation en une seule dimension. Deux nouveaux indicateurs de risque sont présentés dans [MINDERHOUD et BOVY \[2001\]](#), établis à partir du TTC. Le *Time Exposed Time-to-collision* (TET) mesure, sur une fenêtre temporelle donnée, le temps durant lequel le TTC demeure en dessous d'un seuil critique. Le *Time Integrated Time-to-collision* (TIT) intègre, sur tous les instants où le TTC est inférieur au seuil critique, la différence entre le TTC et le seuil. Cette valeur exprimée en s^2 augmente donc lorsque le TTC est faible, en ne quantifiant que les instants où le TTC est sous le seuil critique. D'après les auteurs, ces métriques permettent d'établir le risque encouru durant une séquence de conduite plus précisément que le TTC. Cependant elles semblent n'être utilisables que pour une évaluation du risque effectuée a posteriori, du fait qu'elles nécessitent des mesures sur toute la séquence de conduite. Par ailleurs, les auteurs indiquent qu'une valeur de TET égale à 0s signifie que la situation est sans danger (« A value of 0 implies that there is no safety concern »), alors que le contre-exemple de deux véhicules se suivant à une distance arbitrairement proche et ayant une même vitesse arbitrairement élevée semble indiquer le contraire.

2.2.3.3 État de l'art des métriques de risque en situation de croisement

D'autres travaux concernent l'évaluation du risque sur des scénarios d'intersection, notamment en environnement urbain. De nombreuses combinaisons de manœuvres sont possibles en fonction des destinations des différents véhicules en présence. Par ailleurs, on peut constater que la littérature à propos des métriques de risque sur cet environnement se trouve être moins fournie que celle à propos de situations longitudinales, réputées moins complexes.

Les auteurs de [CHING-YAO CHAN \[2006\]](#) proposent une nouvelle métrique de risque pour les scénarios de virage à gauche sur une intersection (*Left-Turn-Across-Path Opposite-Direction*) : $CriticalityIndex = \frac{V^2}{TTC}$. La vitesse V est celle du véhicule obstacle arrivant en face, et le TTC est le laps de temps avant que ce dernier ne croise le chemin du véhicule qui effectue le virage à gauche. On constate que cette métrique ne prend pour entrée que la position et la vitesse du véhicule obstacle, sans tenir compte de l'état du véhicule effectuant le virage. Elle mesure donc le risque de se trouver sur l'intersection, dans le cas où le véhicule est à l'arrêt et doit décider d'effectuer son virage ou non. Elle ne semble en revanche pas adaptée pour pouvoir être intégrée dans un algorithme de prise de décision plus générique, qui nécessite également d'évaluer le risque tout au long de la phase d'approche de l'intersection.

Une approche probabiliste de métrique de risque est proposée dans [ZHU et collab. \[2009\]](#). La probabilité de collision à une intersection est calculée à partir d'une distribution des variables aléatoires de vitesse et d'accélération calibrée sur des données réelles en situation urbaine. Cependant il est difficile de distinguer l'apport de la métrique probabiliste en elle-même de l'intérêt d'utiliser des lois probabilistes pour décrire le mouvement des véhicules. Il eut été intéressant de comparer les résultats à ceux qui seraient obtenus par une métrique plus classique (ex : TTC) sur des trajectoires générées par les mêmes variables aléatoires de vitesse et d'accélération.

Dans [VOGEL \[2003\]](#), l'auteur étudie la distinction entre TTC et TG, théoriquement, puis empiriquement en effectuant des mesures sur un carrefour urbain. Bien que le TTC tienne compte de la vitesse du premier véhicule, ce qui représente une information supplémentaire par rapport au TG, il ne semble pas être adapté pour mesurer continuellement le risque d'une situation (cf. Tab. 2.3). En effet, le TTC demeure très élevé voire n'est pas défini dans un grand nombre de situations dangereuses où le TG peut être faible. Par exemple, lorsque deux véhicules se suivent de très près avec la même vitesse, ou encore lorsque deux véhicules se croisent dans un laps de temps très faible à une intersection, sans toutefois entrer en collision.

Cette conclusion sur la pertinence relative d'utiliser le TTC seul pour évaluer le risque de colli-

| | | TG | |
|-----|-------------------------|------------------|------------------------|
| | | Faible | Élevé |
| TTC | Faible | Danger imminent | (Situation impossible) |
| | Élevé (voire infini) | Danger potentiel | Pas de danger |

TABEAU 2.3 – Relations entre TTC, TG et le danger réel de la situation (Tab. issu de [VOGEL \[2003\]](#))

sion à une intersection peut également être tirée en observant les résultats de [YAN et collab. \[2018\]](#), qui comparent le TTC et le PET sur des situations conflictuelles dans un environnement urbain. Les auteurs ne le mentionnent pas directement, mais leurs résultats montrent que ces deux métriques ne sont pas corrélées, et que certaines situations très risquées au regard du PET ont un TTC situé dans la moyenne. Pour autant, ils estiment que le TTC permet de relativiser le risque lorsqu’une collision dangereuse (PET faible en valeur absolue) n’aura lieu qu’après un délai assez long (TTC suffisamment élevé).

Les auteurs de [BÄRGMAN et collab. \[2015\]](#) étudient la zone de confort des conducteurs lors d’un virage à gauche sur un carrefour en T. Une étude contrôlée a été mise en place impliquant 22 participants qui conduisent à tour de rôle un véhicule traversant une intersection. Ils doivent tourner à gauche alors qu’un autre véhicule traverse l’intersection en allant tout droit. Ce véhicule est un robot autonome entouré d’un ballon qui prend la forme de la carrosserie d’une voiture afin de maximiser le réalisme tout en minimisant les risques en cas de collision. Les auteurs identifient le PET comme le critère de risque le plus pertinent pour le scénario considéré, et mesurent sa valeur dans les scénarios de deux configurations distinctes : conduite normale et conduite sous l’injonction de passer au plus vite l’intersection. Cette étude permet d’identifier les valeurs moyennes de PET lors d’une conduite naturelle sur intersection. Les conducteurs tendent à préférer un $PET > 2.5s$ pour une conduite confortable, mais peuvent accepter de prendre plus de risque avec un $PET > 1.0s$ lorsqu’ils sont pressés.

2.2.4 Conclusion

L’étude de l’état de l’art en matière de planification de trajectoire et de prise de décision pour une voiture autonome permet de constater que le nombre d’approches applicables varie fortement en fonction de l’environnement considéré. On observe notamment la prévalence des approches dédiées aux situations sur autoroute, où la décision est restreinte à la planification d’un changement de voie. De nombreuses approches sont également proposées pour des scénarios urbains, notamment sur des intersections non supervisées par des feux de circulation. Dans les situations où les interactions avec les autres véhicules sont plus complexes, il s’avère pertinent de recourir à des modèles statistiques pour prédire les trajectoires futures des autres véhicules et améliorer ainsi la planification pour le VA. Il peut également être judicieux de préférer une planification dynamique qui met à jour régulièrement une trajectoire de référence plutôt qu’effectuer une décision de façon sporadique, décision qui fixerait un enchaînement de manœuvres prédéterminées après avoir analysé la situation.

On peut remarquer que les plus récentes approches statistiques de prédiction de mouvement qui permettent d’anticiper les trajectoires des véhicules sont rarement applicables dans des environnements non contraints comme un espace navigable ouvert, sans marquage au sol ni voies de circulation. A cette constatation s’ajoutent les limites des approches probabilistes, en matière d’identification du contexte et de biais éventuels dans le jeu de données sur lequel la modélisation est établie. Pour toutes ces raisons, on aura par la suite recours à des prédictions s’appuyant sur des extrapolations physiques des mouvements des véhicules.

Le type d’environnement semi-contraint qui nous concerne ici n’est quasiment pas envisagé dans les méthodes de planification de trajectoire d’un véhicule, comme l’indique le [Tableau 2.2](#).

Les algorithmes effectuant une décomposition chemin-vitesse semblent les plus adaptés pour ce type d'environnement, puisqu'ils ne nécessitent pas de disposer d'une grande quantité de données et d'un mécanisme d'apprentissage pour planifier une trajectoire sûre. De plus, dans ce type d'approches, la qualité du chemin obtenu est suffisamment correcte pour qu'il puisse être suivi par le véhicule. On peut noter par ailleurs que la seule méthode existante qui mentionne explicitement un environnement sans contraintes fortes de topologie de route appartient à cette famille d'approches (MAO et collab. [2018]).

Nous avons donc décidé de proposer dans cette thèse une approche de type décomposition chemin-vitesse pour calculer le mouvement d'un véhicule en interaction avec d'autres véhicules et circulant dans un environnement semi-contraint. Elle consiste à suivre un chemin statique établi d'avance, sans considérer les obstacles mobiles, et à adapter dynamiquement le profil de vitesse du véhicule sur ce chemin. Dans ce cadre :

- Différentes hypothèses de profils de vitesse seront évaluées. Tous les profils de vitesse devront éviter les collisions et permettre au véhicule d'adopter une conduite souple et confortable, se rapprochant du comportement « naturel » d'un conducteur humain.
- Afin de mieux anticiper l'action du module de contrôle longitudinal du véhicule, une modélisation interne de type MPC sera ajoutée à la stratégie de planification.
- Enfin, deux métriques de risque seront utilisées en fonction du contexte, selon les conclusions établies à partir de l'étude de la littérature : le PET sera utilisé dans des scénarios d'intersections simples, et le TG sera appliqué lorsque la topologie de la route est moins bien définie.

2.3 Conception et développement d'un outil de simulation

Pour tester les algorithmes de prise de décision de véhicules automatisés, un outil de simulation est nécessaire. Les scénarios de tests qui sont envisagés présentent un risque de collision et de dommages matériels trop important pour pouvoir être effectués en conditions réelles, y compris sur une zone de test dédiée.

Plusieurs outils de simulation existants ont été comparés, mais aucun n'ayant pu remplir tous les critères nécessaires, un nouvel outil de simulation a été conçu. Cette section détaille les différentes caractéristiques de cet outil de simulation.

2.3.1 Critères retenus pour le simulateur

Afin de pouvoir couvrir des scénarios de trafic urbain où les véhicules sont amenés à interagir de façon complexe, le simulateur doit intégrer les caractéristiques suivantes :

1. Le simulateur doit être capable de laisser un humain conduire dans un environnement contrôlé (driver-in-the-loop). De cette façon, les comportements de plusieurs conducteurs différents peuvent être testés face aux mêmes scénarios. Les commandes du conducteur ainsi que les états physiques des véhicules présents dans un scénario sont enregistrés, afin d'être analysés par la suite.
2. Les algorithmes de prise de décision peuvent être implémentés et testés dans l'environnement de simulation. Ces tests peuvent être de nature objective (avec des métriques de risque, de confort...) ou subjective (soumis à une évaluation par un observateur humain). Ces évaluations peuvent être obtenues pour tout type de scénario y compris les plus dangereux, ce qui est pratiquement impossible à obtenir dans le cas d'un test réel.
3. Plusieurs véhicules conduits par différents algorithmes de décision peuvent être placés en même temps dans un scénario de conduite en simulation. Ceci permet d'automatiser la validation objective de ces algorithmes en présence d'autres véhicules réagissant à leur environnement.

Finalement, un simulateur réaliste, polyvalent, générique et multi-agents est requis. « Réaliste » car il doit recourir à une modélisation physique de l'environnement et des véhicules assez juste pour que les trajectoires testées numériquement soient faisables en pratique. « Polyvalent » en matière d'environnement de conduite. Le logiciel doit permettre de simuler facilement différentes topologies de route : différents types d'intersections (cf Fig. 2.2), avec ou sans panneaux de circulation (cf Fig. 2.3), les largeurs des voies et les angles aux intersections doivent être personnalisables. Le but étant de permettre de reproduire facilement et rapidement tout type de topologie d'intersection urbaine. Pour générer des interactions complexes entre conducteurs, une simulation multi-agents doit permettre de réunir plusieurs conducteurs humains en les plaçant simultanément sur le même scénario. Cela permet d'étudier dans un environnement contrôlé comment les humains prennent des décisions lorsqu'ils sont en interaction avec d'autres usagers de la route. L'attribut « générique » fait référence à la faculté de cet outil de simulation de pouvoir confronter dans le même scénario des conducteurs humains, des véhicules conduits par des algorithmes et d'autres reproduisant des trajectoires pré-enregistrées. Il permet donc également de valider des algorithmes comportementaux selon différents critères de performance et sur des scénarios standardisés.

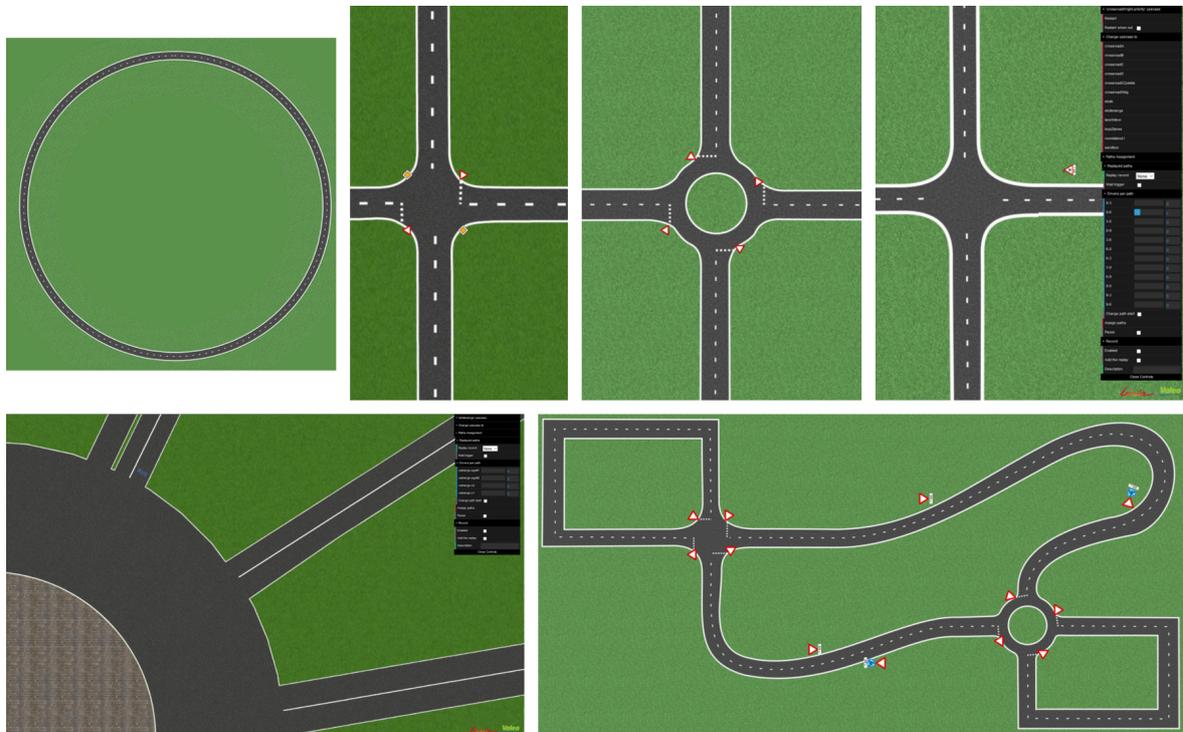


FIGURE 2.2 – Topologies de route principalement utilisées avec le simulateur. En haut à gauche : une route de deux voies permettant de conduire en boucle. En haut à droite : deux intersections simples et un carrefour giratoire. En bas à gauche : un large rond-point où l'espace navigable est très large. En bas à droite : un scénario de test utilisé pour habituer les participants avant une étude.

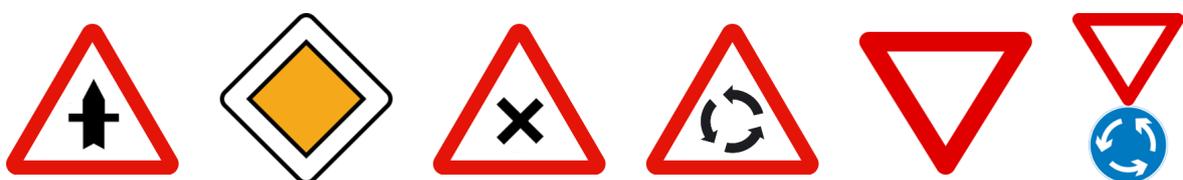


FIGURE 2.3 – Panneaux de signalisation disponibles dans le simulateur, pouvant être intégrés sur une carte.

2.3.2 Architecture en réseau du simulateur

Le simulateur multi-agents suit une architecture serveur/clients qui permet de centraliser les calculs d'interactions physiques sur un terminal en mode serveur et de déporter la visualisation et le contrôle manuel sur plusieurs machines en réseau, via un simple navigateur web. Afin de faciliter une telle implémentation, nous avons choisi de développer en nodeJS –un interpréteur de langage JavaScript qui est optimisé pour s'exécuter en mode serveur. Node a l'intérêt de faciliter l'utilisation de sockets et d'événements réseau par sa structure fortement orientée asynchrone.

La visualisation de la scène est générée par la bibliothèque logicielle ThreeJS, qui permet d'effectuer un rendu 3D en temps réel. La simulation des interactions physiques est effectuée par la bibliothèque Bullet, une référence en la matière (notamment utilisée par le projet open-source Blender). Bullet est développé en C++ mais c'est un wrapping JavaScript, AmmoJS, qui est utilisé afin d'exposer les méthodes de Bullet en JavaScript.

Enfin, la bibliothèque JavaScript PhysiJS utilise AmmoJS et implémente directement des modèles de véhicules qui interagissent physiquement avec le sol et sont pilotés par des commandes au niveau de la dérivée de l'accélération (i.e. « jerk » en anglais, équivalent à l'action sur les pédales d'accélérateur et de frein) et au niveau du contrôle de la vitesse angulaire de lacet (équivalent à l'action sur le volant). Tout d'abord, ces bibliothèques JavaScript ont été adaptées afin de les rendre compatibles avec nodeJS. Puis une synchronisation ad-hoc a été développée entre le serveur et les clients : ceux-ci envoient régulièrement les commandes manuelles des conducteurs humains vers le serveur, celui-ci calcule le nouvel état de la simulation, et finit par mettre à jour les clients en leur partageant les nouveaux états des différents objets mobiles de la scène.

Les véhicules en simulation peuvent être conduits par un humain latéralement via le mouvement horizontal de la souris ou via un volant USB, et longitudinalement via le mouvement vertical de la souris ou via des pédales USB.

2.3.3 Modèle physique des véhicules

Les mises à jour de l'état de la scène sont effectuées régulièrement toutes les 100ms, à partir d'équations physiques reproduisant la gravitation et les différentes forces de frottement s'appliquant aux objets mobiles de la scène, notamment en fonction de la friction du sol et du modèle de véhicule (« raycast vehicle model »). Ce dernier comporte quatre systèmes de roue avec suspensions ainsi qu'une masse de châssis reposant dessus. Cette modélisation reproduit le tangage et le roulis du véhicule ainsi que les rebonds dus aux suspensions en fonction des virages, accélérations et décélérations, ce qui donne un meilleur réalisme visuel pour le conducteur. En plus de la masse du châssis du véhicule, les systèmes de suspension sont décrits par leurs paramètres de rigidité, compression, amortissement et déflexion maximale.

2.3.4 Génération de topologie de route

Une des caractéristiques de ce simulateur est de faciliter la création de nouvelles topologies de route et de les intégrer dans des scénarios spécifiques. La représentation de l'environnement est principalement décrite par une image vue de haut qui précise avec différentes textures les espaces navigables (route de bitume), les espaces à éviter (herbe du bas-côté), ainsi que les marquages au sol. Cette carte est également complétée avec un fichier de configuration de format json qui permet d'ajouter des panneaux de circulation (type, position, orientation) qui sont intégrés dans l'environnement 3D, ainsi qu'une liste de déclencheurs pouvant déterminer le comportement de certains algorithmes lorsqu'un conducteur humain approche une certaine position. Une liste de trajectoires cibles peut également être établie pour une scène. Une trajectoire est définie par un échantillonnage régulier de points (x, y, vitesse) au cours du temps, stockés dans un fichier csv. Ces coordonnées peuvent être assignées à des véhicules en tant que trajectoire complète (les contrôleurs longitudinal et latéral du véhicule reproduisent alors au plus proche cette trajectoire), ou bien en tant que chemin (seul le contrôleur latéral suit le chemin, le choix de la vitesse étant laissé à un algorithme de planification ou à un conducteur humain).

La création manuelle de la carte peut être assistée par des scripts de génération automatique à partir de l'API d'OpenStreetMap. Les scripts sont écrits en python et exécutés par le logiciel d'édition d'images GIMP. Le premier prend en paramètre d'entrée deux coordonnées GPS délimitant un rectangle, récupère sur OpenStreetMap les portions de route qui s'y trouvent, et s'en sert pour générer une vue de haut prête à être utilisée par le simulateur. Plusieurs calques sont créés afin de faciliter l'édition manuelle (ajout/suppression de voies). Cette fonctionnalité est pratique pour s'inspirer d'une intersection existante afin de garder un réalisme notamment au niveau de l'échelle des voies de circulation. Cela permet également un gain de temps considérable par rapport à une réplique en partant de zéro (ou d'un calque d'image satellite). Un exemple d'utilisation de ce script est illustré Fig. 2.4.



FIGURE 2.4 – Exemple d'utilisation du script de génération de carte. De gauche à droite : carte OpenStreetMap, carte générée avec le script, et vue 3D du simulateur. Seuls les panneaux de cédez-le-passage et le marquage au sol associé ont été ajoutés manuellement.

Le deuxième script planifie un chemin navigable en s'appuyant sur les coordonnées des voies. Le chemin généré est une succession de courbes de Bézier cubiques jointes par [continuité G1](#). Avant d'être exporté en csv pour le simulateur, ce chemin peut être adapté manuellement avec l'outil chemin de GIMP. Cet outil utilise en interne des courbes de Bézier cubiques et est donc parfaitement adapté.

2.3.5 Visualisation de la scène et mode moniteur

Le simulateur permet deux modes de visualisation : une vue « ego » depuis l'intérieur d'un véhicule (pour le conduire ou visualiser le comportement d'un algorithme en tant que passager) et un mode « moniteur » permettant de contrôler le déroulement du scénario depuis une vue de haut globale de la scène. Pour les deux modes le moteur 3D ThreeJS effectue le rendu de la scène en utilisant la carte vue de haut comme texture de sol, en ajoutant les modèles 3D des différents véhicules ainsi que des panneaux de signalisation.

Dans la vue « ego », le rendu s'adapte à la taille de la fenêtre, ainsi qu'au nombre d'écrans utilisés : jusqu'à 3 écrans pour une vue panoramique de 180° (cf Fig. 2.5). Lorsqu'un seul écran est utilisé, la focale de la caméra de rendu est modifiée afin de couvrir à elle seule 135°. Ceci permet de compenser la réduction de champ de vision, mais déforme légèrement l'image. Il est préférable de recourir à 3 écrans lorsque l'évaluation subjective d'un conducteur humain ou a fortiori son comportement en situation sont étudiés. Un polygone avec une texture de ciel est ajouté pour rendre la scène plus réaliste et donc moins gênante pour l'immersion des conducteurs humains. Un compteur de vitesse est ajouté, ainsi que quelques contrôles simples permettant de basculer entre une vue globale de haut ou la vue du pare-brise de la voiture. Le conducteur peut également ajouter l'affichage de rétroviseurs.

Le mode « moniteur » permet de superviser les scénarios via un menu graphique (cf Fig. 2.7), en affectant par exemple les conducteurs disponibles (humains ou algorithmes) sur des chemins à suivre prédéfinis, ou de redémarrer le scénario à son état initial. Il permet également de lancer l'enregistrement d'une simulation complète en stockant toutes les 100ms l'état de chaque véhicule



FIGURE 2.5 – Poste de conduite en simulation avec vue sur 3 écrans. Des pédales ainsi qu'un volant USB sont utilisés pour la conduite.

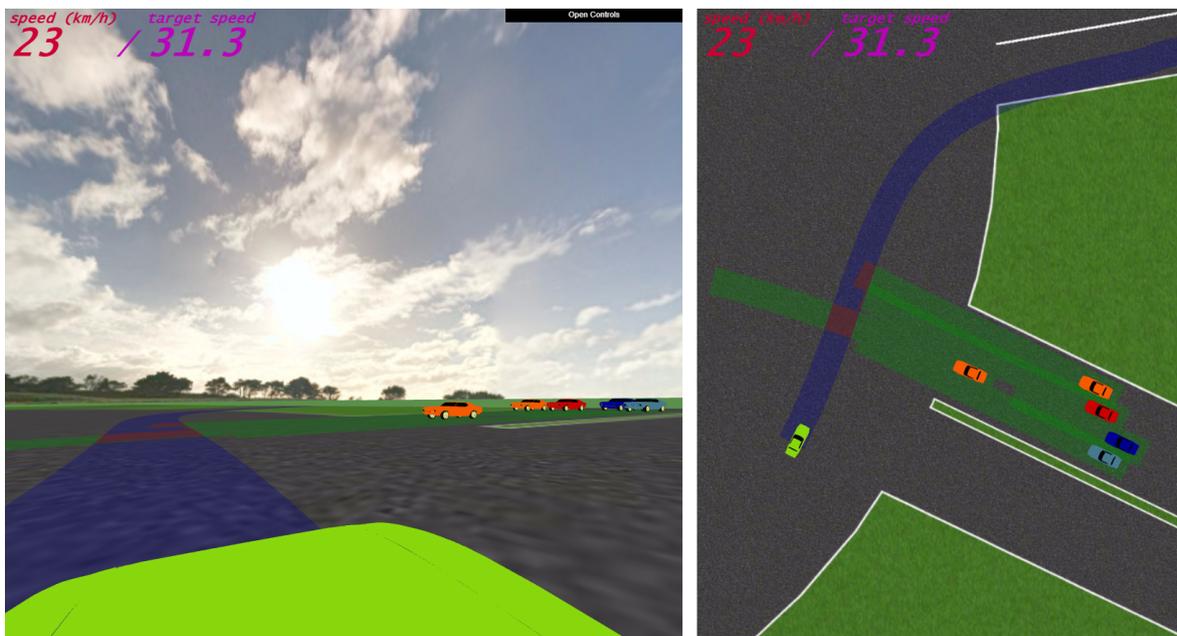


FIGURE 2.6 – Aperçu du rendu de simulation. La même scène est affichée avec le point de vue 3D depuis un véhicule et une vue de haut.

(position en 3D, vitesse, rotation sur 3 axes, commandes de volant et pédales) dans un fichier json, ainsi que d'autres informations sur la séquence enregistrée. Un enregistrement peut être rejoué dans le simulateur de deux façons. Premièrement en assignant à chaque véhicule un algorithme qui va s'asservir en position (contrôleur latéral) et en vitesse (contrôleur longitudinal) sur les trajectoires sauvegardées. Cette méthode permet d'ajouter également de nouveaux véhicules véritablement conduits par des humains ou par des algorithmes décisionnels. Les véhicules asservis sur les trajectoires mémorisées peuvent dériver très légèrement par rapport à la cible, c'est un aléa classique dû au module de contrôle. La deuxième méthode consiste à visionner simplement les enregistrements image par image depuis le moniteur. Cette fois aucune erreur de positionnement n'est possible, et aucun autre véhicule ne peut être ajouté. Cette méthode permet de reVISIONNER précisément un scénario en vue de l'analyser.

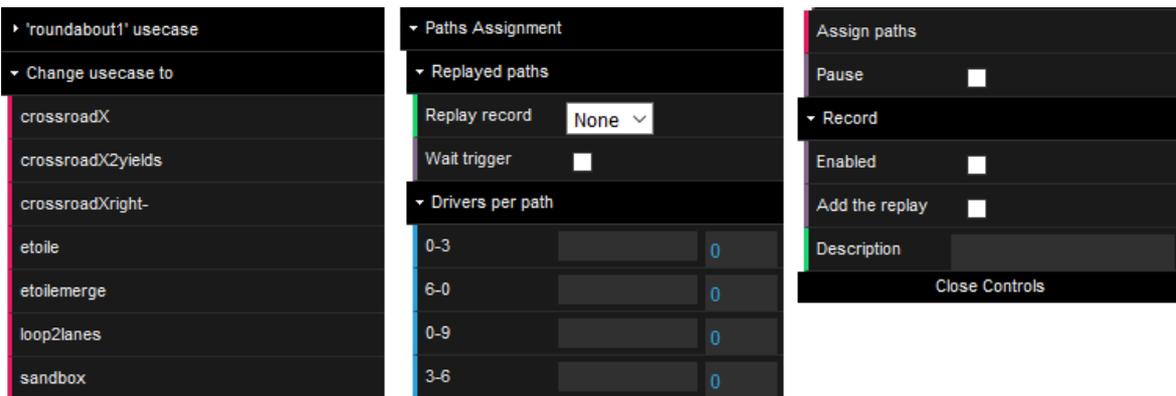


FIGURE 2.7 – Menu graphique du mode moniteur. À gauche le sous-menu pour changer de carte, au milieu le sous-menu pour affecter des conducteurs humains à des chemins prédéfinis, ainsi qu'une option pour charger un scénario sauvegardé, et à droite le sous-menu pour enregistrer le scénario en cours.

2.3.6 Systèmes d'asservissement longitudinal et latéral

Les deux contrôleurs d'asservissement suivants sont disponibles sur les véhicules en simulation.

Le contrôleur latéral permet de suivre un chemin de référence et ainsi de rester dans la voie assignée au véhicule. Il est utilisé quasiment systématiquement dans les travaux qui suivent, puisque le chemin est prédéfini et que le processus de décision est simplifié à un ajustement de la vitesse. Il permet également aux conducteurs humains de se concentrer plus aisément sur leur profil de vitesse, étant donné que la tenue de route est difficile lorsque l'environnement de simulation n'est affiché que sur un seul écran. L'angle de correction de lacet appliqué sur le volant est déduit de deux écarts du véhicule au chemin de consigne. Le premier écart (ou erreur) est la distance du point de contrôle au chemin et le second est l'écart angulaire entre l'orientation du véhicule et la direction du chemin (direction prise au projeté du point de contrôle sur le chemin). Le schéma 2.8a illustre graphiquement ces deux erreurs par rapport à la consigne et détaille leur somme pondérée qui forme la correction angulaire. Les poids ka et kd sont empiriquement fixés respectivement à 0.5 et 0.05.

Le contrôleur longitudinal ajuste les commandes virtuelles du véhicule simulé (pédales d'accélérateur et de freins) afin de l'asservir en vitesse sur une consigne donnée. Les algorithmes de décision donnent en sortie une trajectoire cible dont le profil de vitesse est utilisé comme consigne par le contrôleur longitudinal. Ce contrôleur de vitesse est également utilisé lorsqu'un humain conduit manuellement un véhicule via la souris : la position verticale de la souris donne une référence absolue de vitesse que va cibler le contrôleur. Cette méthode (cf en bas de la Fig. 2.8b) a supplanté l'implémentation précédente (en haut de la figure), où la position verticale de la souris était une référence relative d'accélération afin d'imiter le fonctionnement des pédales. Avec cette précédente méthode, une position statique de la souris induisait soit une accélération constante

soit un freinage, ce qui ne permettait pas de garder une vitesse constante et perturbait notablement la conduite manuelle. L'équation reliant la consigne de vitesse avec l'accélération appliquée au véhicule est décrite par l'équation 2.1.

$$Acc = Acc_{max} \cdot \left(1 - \left(\frac{Speed_{current}}{Speed_{target}} \right)^3 \right) \quad (2.1)$$

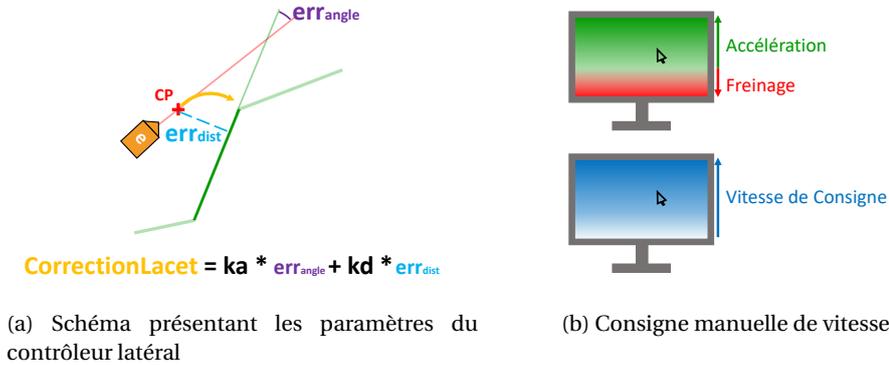


FIGURE 2.8 – Schémas des contrôleurs

2.3.7 Conclusion

Le simulateur de conduite détaillé dans cette section regroupe dans un même environnement des véhicules conduits par des humains et d'autres par des algorithmes. Il présente les caractéristiques suivantes :

1. La dynamique des véhicules est générée par un moteur physique à partir d'une modélisation complète des composants mécaniques du châssis, ce qui permet d'obtenir des trajectoires plus réalistes que si elles faisaient l'objet d'une modélisation cinématique.
2. Différents environnements de conduite sont intégrés dans le simulateur et il est possible d'en générer rapidement de nouveaux à partir des données cartographiques d'OpenStreet-Map.
3. Les trajectoires suivies par les véhicules sont enregistrées afin de pouvoir être analysées à l'issue de la simulation.

Cet outil est utilisé dans le reste du chapitre : pour obtenir des données sur le comportement des conducteurs humains dans des situations contrôlées où ils doivent prendre une décision à une intersection, et comme outil de validation des approches d'adaptation dynamique de vitesse dans différents environnements.

Les contrôles longitudinal et latéral des véhicules sont séparés dans le simulateur, de sorte qu'il est possible d'asservir le véhicule à une voie de circulation en utilisant directement le module de contrôle latéral. Dans ce cas, seule la possibilité d'ajuster la consigne de vitesse du véhicule est laissée à l'algorithme intelligent ou au conducteur humain. En pratique ce fonctionnement a été systématiquement utilisé, puisque les travaux qui suivent traitent uniquement du problème de la planification du profil de vitesse le long d'un chemin déterminé.

2.4 Planification de vitesse aux intersections

Ces travaux concernent la prise de décision d'un véhicule automatisé sur des topologies d'intersections en présence d'autres véhicules non communiquant. L'objectif est ici de s'inspirer du comportement humain afin de réaliser une manœuvre qui semble la plus « naturelle » possible, tout en évitant les collisions grâce à une analyse prédictive du risque encouru.

Le véhicule étudié dans différentes configurations d'intersection, pouvant être dirigé en simulation soit par un conducteur humain soit par un algorithme, sera désigné dans un premier temps par « le véhicule ego ». La notation « VA » fera ensuite référence au même véhicule, une fois automatisé par l'approche de planification.

La perception de l'environnement est supposée toujours disponible et les erreurs ou imprécisions des capteurs sont négligées. La prédiction de mouvement est effectuée par une modélisation physique simple en faisant à chaque instant l'hypothèse que la vitesse des autres véhicules est constante. Cette hypothèse peut sembler simpliste par rapport à un modèle de comportement plus complexe qui tiendrait compte du contexte précis, mais elle permet néanmoins d'appréhender la situation « pire cas », dans le cas où les autres véhicules ne réagiraient pas à la présence du véhicule ego. Cela se justifie particulièrement sur ces scénarios où le véhicule ego n'a pas la priorité sur l'intersection.

La planification de trajectoire pour le véhicule automatisé est subdivisée en deux étapes. Tout d'abord, un chemin est planifié en définissant une suite de courbes de Bézier de [continuité G1](#). Un module de planification dédié pré-calcule le chemin de façon statique, en considérant uniquement la topologie de la route et sans tenir compte des obstacles mobiles. La trajectoire du véhicule est ensuite complétée par un profil de vitesse calculé en temps réel en fonction des autres véhicules.

Au préalable, une étude sur simulateur permet d'enregistrer les profils de vitesse de conducteurs humain sur différents scénarios d'intersections. L'algorithme d'adaptation de vitesse tend à reproduire ces profils de vitesse afin d'assurer des accélérations confortables, et de modéliser ainsi un comportement « naturel ». Une prédiction à horizon fuyant évalue en permanence le risque de plusieurs hypothèses de profils de vitesse, en utilisant notamment comme métrique le [PET](#) introduit section [2.2.3.3](#). Les profils trop risqués sont ainsi écartés, et le plus optimal en temps de parcours est retenu.

En somme, cette approche combine des profils de vitesse inspirés de la conduite humaine avec des critères robustes d'évaluation de risque, afin d'adapter en temps réel la vitesse d'un véhicule automatisé sur une intersection.

2.4.1 L'environnement des intersections

Dans cette étude, les environnements d'intersections urbaines sont étudiés au travers de différents scénarios implémentés dans le simulateur. Ces scénarios se déroulent sur des topologies de carrefours en X ou de carrefours giratoires sur lesquels le véhicule ego et l'autre véhicule ne peuvent circuler que sur une seule voie. La prise de décision est ainsi restreinte à une adaptation de la vitesse, qui dépend des interactions avec l'autre véhicule. Afin de favoriser ces interactions, un mécanisme d'adaptation de vitesse simple est intégré sur l'autre véhicule.

2.4.1.1 Scénarios

Pour cette étude, les trois topologies d'intersection urbaine décrites ci-dessous et affichées [Fig. 2.9](#) ont été retenues :

1. un carrefour en X : une route principale prioritaire coupe orthogonalement une route non prioritaire. Deux voies opposées débouchent donc sur des « cédez-le-passage ».
2. un carrefour en X sans priorités particulières : sans marquage au sol, la priorité à droite s'applique.
3. un [carrefour giratoire](#) large d'une seule voie et muni de quatre voies d'entrée/sortie. Chaque voie d'entrée comporte un panneau « cédez-le-passage ».

Au total, sept scénarios ont été retenus : six pour les carrefours en X et un pour le carrefour giratoire. Dans tous les scénarios, le véhicule ego est censé céder le passage à l'autre véhicule, qui a la priorité. Lors du scénario sur carrefour giratoire, le véhicule ego s'approche d'un « cédez-le-passage » après lequel il doit s'insérer sur l'anneau central alors qu'un autre véhicule situé sur une autre voie d'entrée sur la gauche s'est inséré en premier sur le carrefour et a donc la priorité.

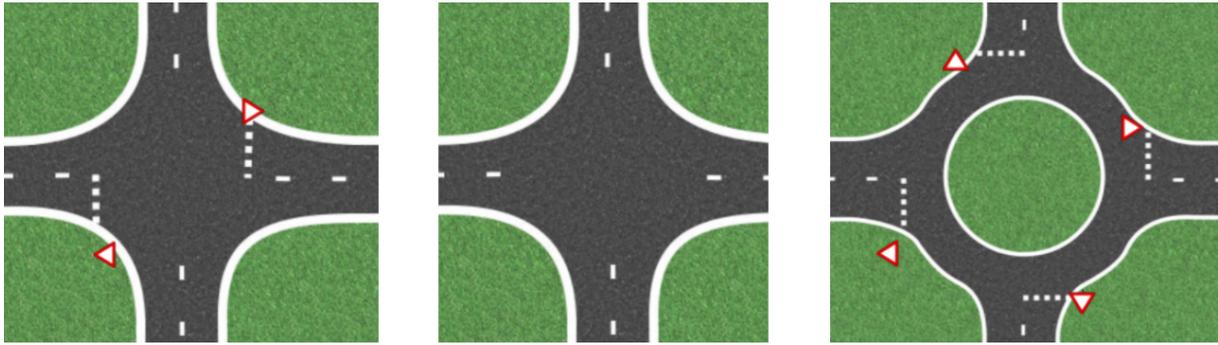


FIGURE 2.9 – Les trois topologies d’intersection urbaine utilisées dans les différents scénarios.

Sur le carrefour en X on distingue six scénarios différents, en fonction des manœuvres effectuées par le véhicule ego et l’autre véhicule. Afin de faciliter leur nommage, leur préfixe indique la manœuvre du véhicule ego :

1. LTAP lorsqu’il croise le chemin de l’autre véhicule en faisant un virage à gauche (*Left Turn Across Path*)
2. RTAP pour un virage à droite (*Right Turn Across Path*)
3. SAP lorsqu’il croise l’autre véhicule en allant tout droit (*Straight Across Path*)

Le suffixe indique la direction de provenance de l’autre véhicule en sachant que celui-ci continue toujours sa course en allant tout droit :

1. LD lorsqu’il provient de la gauche (*Left Direction*)
2. RD pour la droite (*Right Direction*)
3. OP s’il provient de la direction opposée (*Opposite Direction*)

Il y a donc six scénarios pour la topologie de carrefour en X (illustrés Fig. 2.10) : trois pour effectuer un virage à gauche (LTAP/LD, LTAP/OD, LTAP/RD), deux pour avancer tout droit (SAP/LD, SAP/RD) et un pour effectuer un virage à droite (RTAP/LD). Les autres scénarios n’impliquant pas d’interaction directe entre les deux véhicules, ils sont négligés.



FIGURE 2.10 – Les six scénarios pour la topologie de carrefour en X. Le chemin du véhicule ego est en vert.

2.4.1.2 Configurations

Chaque scénario est répété avec 8 configurations différentes, dans lesquelles le comportement du second véhicule change pour que les interactions entre les deux véhicules soient les plus variées possible. Dans tous les scénarios, les véhicules sont au départ à l’arrêt et situés à 200m de l’intersection. Lorsque le véhicule ego arrive au niveau de l’intersection, l’autre véhicule est censé arriver au niveau d’un point P_s dont la position est située à $P_{s_{diff}}$ mètres avant l’intersection, et ce avec une vitesse V_s . Les valeurs des paramètres pour les 8 configurations sont données dans le Tableau 2.4.

| | config 1 | config 2 | config 3 | config 4 | config 5 | config 6 | config 7 | config 8 |
|------------------------|----------|----------|----------|----------|----------|----------|----------|----------|
| $P_{s_{diff}} (m)$ | 60 | 50 | 40 | 30 | 20 | 15 | 10 | 5 |
| $V_s (m \cdot s^{-1})$ | 11.1 | 11.1 | 11.1 | 11.1 | 25 | 25 | 25 | 25 |

TABEAU 2.4 – Paramètres des configurations

2.4.1.3 Algorithme de synchronisation

Puisque les conducteurs du véhicule ego (participants humains ou algorithmes autonomes) ne vont pas agir avec le même comportement dans la scène, ils peuvent ne pas atteindre l'intersection avec le même délai. Afin de standardiser les interactions au niveau de l'intersection entre les différents conducteurs, il est nécessaire de réduire cette variance de délai en modifiant le comportement de l'autre véhicule (appelé par la suite véhicule de suivi) en fonction de l'avancée du véhicule ego. Cette adaptation de vitesse permet de normaliser les différentes configurations, et de les comparer entre elles, indépendamment des profils de vitesse des conducteurs en début de scénario.

Cette adaptation de vitesse du véhicule de suivi est effectuée comme suit. Le véhicule ego (e) approche du point (Pe) où il entrera dans l'intersection (cf Fig. 2.11). Avec une hypothèse de vitesse constante pour le véhicule ego on peut estimer que la distance entre e et Pe , dénotée $DistPe$, sera parcourue durant l'intervalle de temps Tp . L'algorithme de synchronisation du véhicule de suivi (s) adapte sa vitesse ($V_s(t)$) jusqu'à l'intersection, afin qu'à l'instant Tp il atteigne un point donné (Ps) sur l'intersection avec une vitesse donnée (Vp). Le problème consiste donc à construire le profil de vitesse $V_s(t)$ du véhicule de suivi afin de vérifier l'équation 2.2.

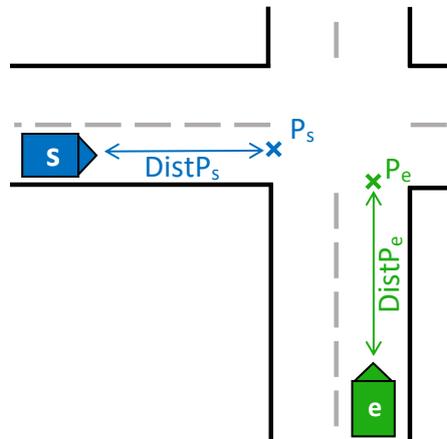


FIGURE 2.11 – Schéma de l'intersection durant la synchronisation

$$\begin{cases} V_s(0) \text{ est réajusté à chaque itération} \\ V_s(Tp) = Vp \\ \int_0^{Tp} V_s(t) \cdot dt = DistPs \\ \frac{dV_s(t)}{dt} \in [\maxFrein, \maxAccélérateur] \end{cases} \quad (2.2)$$

Cette équation traduit la contrainte suivante : le véhicule s , qui a la vitesse instantanée $V_s(0)$ à chaque itération, doit atteindre à l'issue d'un délai TP le point Ps avec la vitesse Vp . Ps est situé à $DistPs$ mètres de s , d'où l'intégrale de l'équation. De plus, l'accélération $\frac{dV_s(t)}{dt}$ est bornée dans l'intervalle des accélérations acceptables.

Pour résoudre l'équation 2.2, on fait l'hypothèse que seules les accélérations maximales sont utilisées. En fonction de la situation il y a deux cas possibles, illustrés par les profils de vitesse de la Fig. 2.12 :

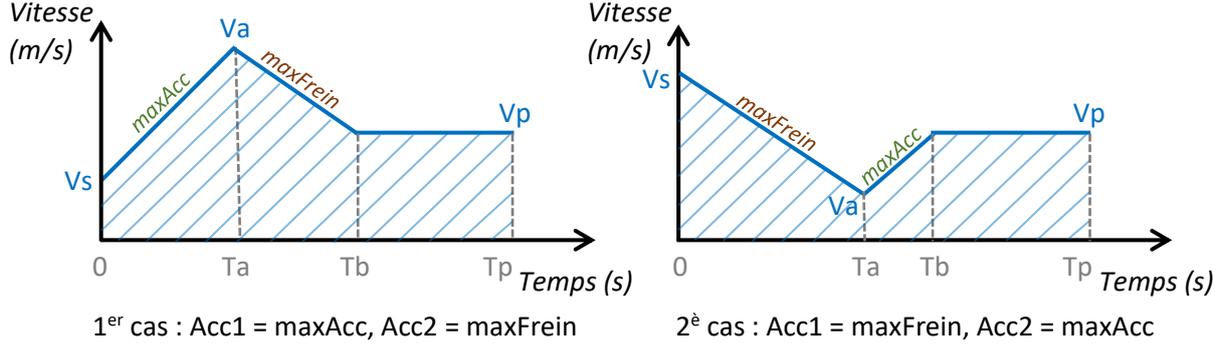


FIGURE 2.12 – Consigne du profil de vitesse lors du suivi.

1. si $V_S < V_p$, le véhicule de suivi accélère jusqu'à l'instant T_a où il atteint une vitesse V_a supérieure ou égale à V_p , puis freine jusqu'à l'instant T_b où il atteint la vitesse V_p et enfin circule à vitesse constante V_p entre T_b et T_p
2. si $V_S > V_p$, le véhicule de suivi freine jusqu'à l'instant T_a puis accélère de T_a à T_b et conserve enfin la vitesse V_p de T_b à T_p .

En utilisant les vitesses V_a , V_p et les instants T_a et T_b on obtient l'équation 2.3, qui peut se réécrire uniquement en fonction de T_a avec l'équation du second degré 2.4.

$$\begin{cases} V_a &= V_S + T_a \cdot Acc1 \\ T_b &= T_a + \frac{V_p - V_a}{Acc2} \\ DistPs &= T_a \cdot \frac{V_S + V_a}{2} + (T_b - T_a) \cdot \frac{V_a + V_p}{2} + (T_p - T_b) \cdot V_p \end{cases} \quad (2.3)$$

$$\begin{cases} 0 &= A \cdot Ta^2 + B \cdot Ta + C \\ A &= \frac{Acc1}{2} \cdot \left(1 - \frac{Acc1}{Acc2}\right) \\ B &= \left(1 - \frac{Acc1}{Acc2}\right) \cdot (V_S - V_p) \\ C &= \frac{-1}{2 \cdot Acc2} \cdot (V_S - V_p)^2 + V_p \cdot T_p - DistPs \end{cases} \quad (2.4)$$

Parmi les solutions $T_a = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$ en considérant les deux cas ($Acc1 = \text{maxAccélérateur}$, $Acc2 = \text{maxFrein}$) et ($Acc1 = \text{maxFrein}$, $Acc2 = \text{maxAccélérateur}$), au maximum 4 solutions sont possibles. Les solutions physiquement impossibles telles que $T_a < 0$ sont retirées et la solution qui minimise T_a est retenue. Enfin, le profil de vitesse sur lequel le véhicule de suivi doit s'asservir est obtenu. Le contrôleur longitudinal reçoit ce profil afin d'adapter le mouvement du véhicule automatisé, et ce processus est répété à chaque itération.

2.4.2 Acquisition de données

L'environnement de simulation est utilisé ici pour agréger des données sur le comportement de conducteurs humains dans les scénarios urbains avec intersections présentés ci-dessus. Une expérience est menée avec 10 participants conduisant leur véhicule (ego) dans les 56 configurations différentes en présence d'un autre véhicule, automatisé, avec lequel le participant doit interagir sur l'intersection.

Avant de débiter la conduite sur les scénarios d'intersection, les 10 participants remplissent un questionnaire général où ils mentionnent un certain nombre d'informations (âge, genre, nombre d'années depuis l'obtention du permis de conduire, nombre de kilomètres conduits en moyenne dans l'année, et le ratio des kilomètres parcourus dans un environnement urbain). Ils s'entraînent pendant 5 min (ou plus si souhaité) dans l'environnement de simulation, afin de s'habituer aux commandes du véhicule. Il est à noter qu'un circuit d'entraînement différent des scénarios enregistrés a été utilisé, afin que le participant ne s'accoutume pas trop aux types d'intersections pour lesquelles l'étude est menée.

Il est demandé à chaque utilisateur de conduire le véhicule ego pour chacune des 56 configurations des 7 scénarios. Pour chaque scénario les configurations sont présentées dans le même ordre, tiré aléatoirement afin d'éviter toute accoutumance à une variation progressive des paramètres qui pourrait biaiser l'expérience. A la fin de chaque configuration, le participant doit remplir un questionnaire afin de donner son évaluation subjective du risque encouru. Aucune instruction spécifique ni aucun détail sur le but exact de l'étude ne sont donnés aux participants. Il leur est juste demandé de respecter le code de la route et de conduire le plus normalement possible. Les trajectoires des véhicules sont enregistrées dans un fichier qui stocke toutes les 100ms toute information utile sur l'état de chaque véhicule (position, orientation, vitesse, et commandes manuelles appliquées via les pédales et le volant).

Après le dernier scénario, un dernier questionnaire est présenté au participant, afin d'obtenir son ressenti général sur l'exercice en simulation (Fig. 2.13).

How challenging was the driving task?
(very unchallenging) 1 2 3 4 5 (very challenging)

How efficient (considering time) did you drive?
(very inefficient) 1 2 3 4 5 (very efficient)

How safe did you drive? (5 is very safe)
(very unsafe) 1 2 3 4 5 (very safe)

How natural did you drive? (5 is very natural)
(very unnatural) 1 2 3 4 5 (very natural)

Did the simulation environment influence your normal driving behavior?

Have there been critical situations? If yes, please describe briefly.

Which scenario was especially stressful?

Which scenario required highest concentration?

The End

FIGURE 2.13 – Questionnaire de fin présenté aux participants après la simulation.

Les données extraites de cette étude vont être utilisées pour établir une modélisation des profils de vitesse de conducteurs humains. Cette modélisation sera fournie à l'algorithme d'adaptation de vitesse décrit en section 2.4.3 afin d'améliorer le confort et l'aspect naturel du profil de vitesse choisi par le véhicule.

2.4.3 Conception de l'algorithme

La planification de mouvement se décompose en deux étapes : la planification de chemin en fonction de la topologie de la route et de la manœuvre effectuée, qui est pré-calculée une fois pour toutes, et le calcul du profil de vitesse sur ce chemin en fonction des interactions avec les autres véhicules, qui est effectué dynamiquement toutes les 100ms. Le processus de décision calcule donc un profil optimal de vitesse qui est fourni au module de contrôle longitudinal afin d'adapter la vitesse instantanée du véhicule ego, qui est désormais automatisé (VA).

Un ensemble de profils de vitesse de référence est créé hors-ligne avant exécution, à partir d'une modélisation du comportement des conducteurs humains en simulation (cf. section 2.4.3.1). L'algorithme itère sur ces références en effectuant pour chacune une courte prédiction (horizon temporel de 5 sec) simulant les interactions du VA qui suit un profil de vitesse donné avec une prédiction de mouvement pour l'autre véhicule. Cette simulation rapide permet d'attribuer un score de risque à chaque profil de référence en fonction d'un contexte réactualisé à chaque instant. Le meilleur profil de vitesse est ensuite envoyé au module de contrôle longitudinal. L'hypothèse de mouvement pour l'autre véhicule est une prédiction basée sur une modélisation physique simple : le véhicule continue d'avancer tout droit à vitesse constante. Cette modélisation pourrait être plus sophistiquée, notamment avec un modèle stochastique tenant compte du contexte environnant. Mais dans le cadre d'intersections simples comme ici, cette prédiction se révèle suffisante, en particulier car elle est réajustée à chaque itération.

Dans cette approche, le PET est utilisé pour mesurer le risque encouru lors du passage sur l'intersection. Il correspond au délai entre l'instant où le premier véhicule sort de la zone de collision et l'instant où le deuxième y entre (voir Fig. 2.14). Comme les chemins sont prédéterminés,

la forme de cette zone de collision est également prédéfinie : par un rectangle pour les carrefours en X et par une portion de couronne pour les carrefours giratoires.

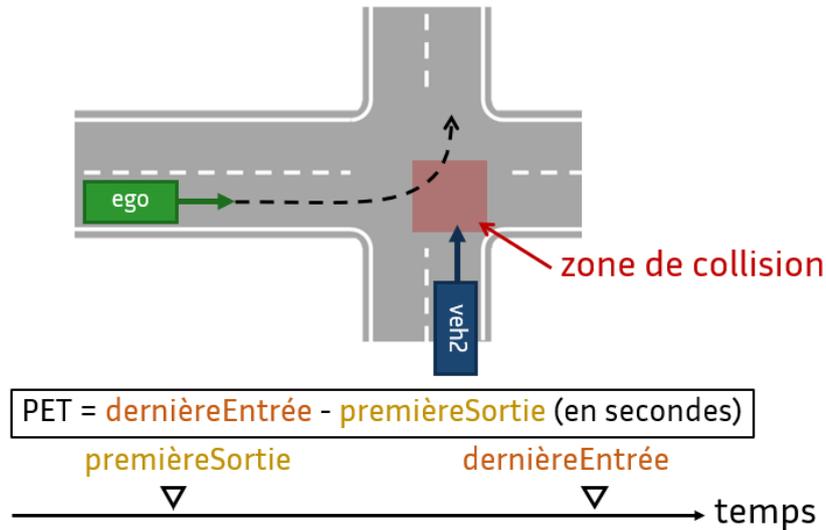


FIGURE 2.14 – Schéma explicatif du PET

2.4.3.1 Profils de vitesse de référence

Les profils de vitesse de référence sont établis pour les sept scénarios décrits en section 2.4.1.1. Pour un scénario donné, l'objectif est d'identifier les formes les plus courantes de profil de vitesse en partitionnant en plusieurs sous-groupes (ou clusters) les profils générés par les participants à la simulation de conduite. Dans chaque sous-groupe, la moyenne du cluster est ensuite utilisée comme profil de vitesse de référence. Afin de pouvoir comparer différents profils de vitesse générés par les conducteurs humains, la vitesse est décrite en fonction de l'abscisse curviligne (ou position longitudinale) le long du chemin et non en fonction du temps. Ceci évite que des profils différents ne se désynchronisent dans le temps et ne puissent être comparés qu'après une correction temporelle (time-warping). L'hypothèse que la vitesse ne dépend que de la position longitudinale est raisonnablement valide pour ces scénarios où un contrôleur latéral s'assure que le véhicule reste dans un couloir prédéfini.

Les profils générés sont divisés en trois groupes (véhicule ego en premier, autre véhicule en premier, ou situation trop risquée) en considérant le PET du scénario et l'ordre de passage des deux véhicules dans l'intersection. Par convention, le PET est positif si le véhicule ego est passé en premier, et est négatif lorsqu'il a cédé le passage. Les cas risqués sont mis de côté : ils peuvent correspondre à des valeurs de PET trop faibles (incluant les cas de collisions) ou à des cas notés comme trop risqués par les participants via le formulaire de risque subjectif (> 8 sur une échelle de 1 à 10). Ensuite, un clustering k-mean est effectué sur les 2 groupes restants, ce qui permet d'obtenir des profils moyens qui rendent compte des principales tendances du comportement des participants humains lorsqu'ils s'engagent sur l'intersection (Fig. 2.15) ou lorsqu'ils cèdent le passage (Fig. 2.16). Pour appliquer l'algorithme k-mean sur les données séquentielles, chaque profil est représenté sous la forme d'un vecteur dont les éléments sont les vitesses à chaque pas de 1m le long de la trajectoire du véhicule. Les profils enregistrés se croisent rarement le long de l'abscisse curviligne (i.e. un profil plus rapide qu'un autre en un point le reste sur toute la trajectoire), ce qui permet au k-mean d'effectuer une bonne séparation des profils en clusters. Dans certains scénarios, un premier clustering fait apparaître des clusters singletons contenant un seul profil. Dans ces cas, ces profils isolés sont retirés des données avant d'effectuer un second clustering qui est alors mieux équilibré.

Finalement les profils de référence (voir Fig. 2.17) sont constitués de :

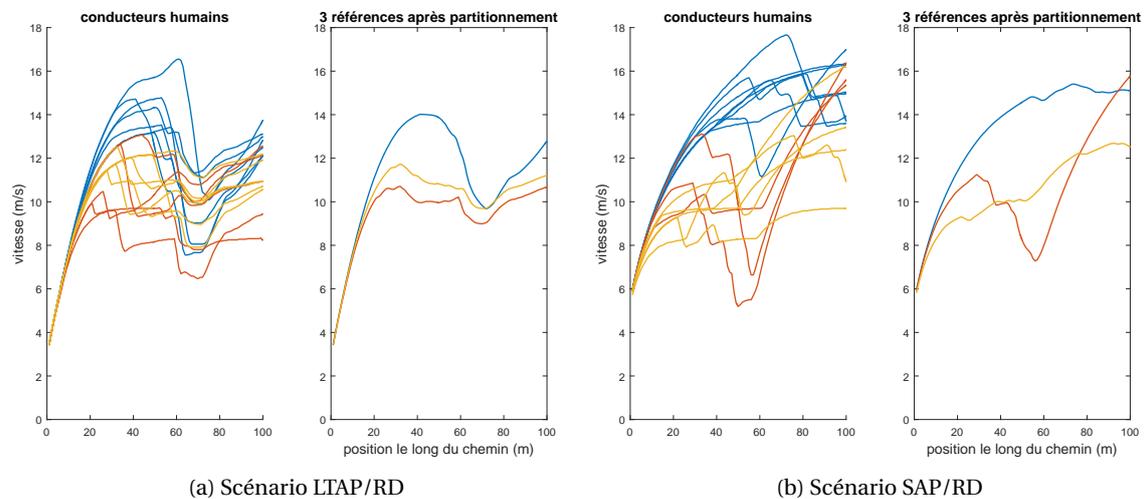


FIGURE 2.15 – Profils de vitesse d’engagement sur l’intersection exécutés par les conducteurs humains en simulation, et profils de vitesse de référence générés après séparation en 3 clusters. Ici, seuls les profils où le véhicule ego passe en premier ($PET > 0$) sont considérés.

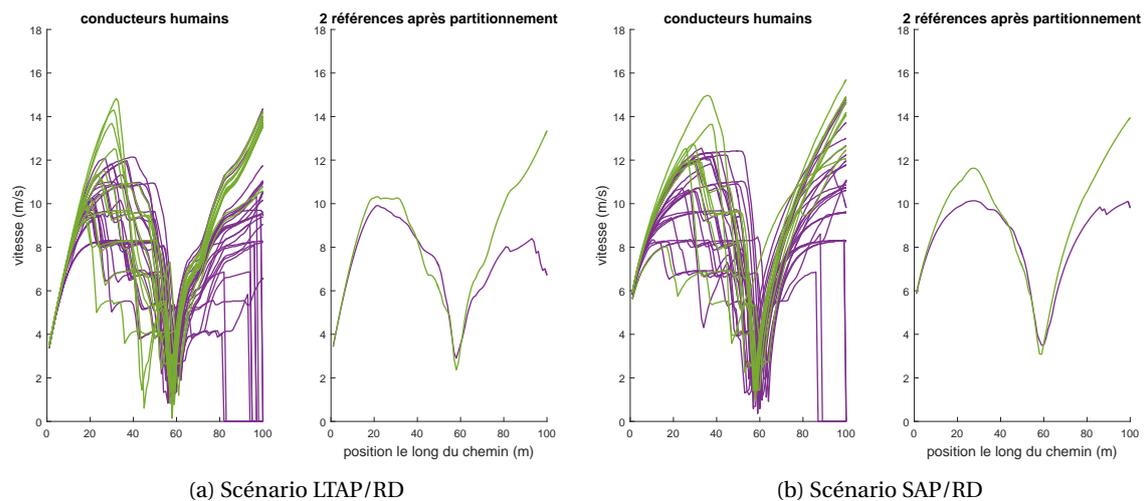


FIGURE 2.16 – Profils de vitesse cédant le passage à l’intersection exécutés par les conducteurs humains en simulation, et profils de vitesse de référence générés après séparation en 2 clusters. Ici, seuls les profils où le véhicule ego passe après l’autre véhicule ($PET < 0$) sont considérés.

1. Trois profils de vitesse pour entrer rapidement dans l'intersection : ce sont les moyennes des 3 clusters formés par les situations où le véhicule ego est passé en premier de manière sûre
2. Deux profils de vitesse pour entrer plus lentement dans l'intersection : ce sont les moyennes des 2 clusters formés par les situations où le véhicule ego a cédé le passage
3. Un profil pour s'arrêter avant l'intersection au niveau de la ligne de « cédez-le-passage » : ce profil modélise un arrêt définitif avec une vitesse nulle après l'intersection.

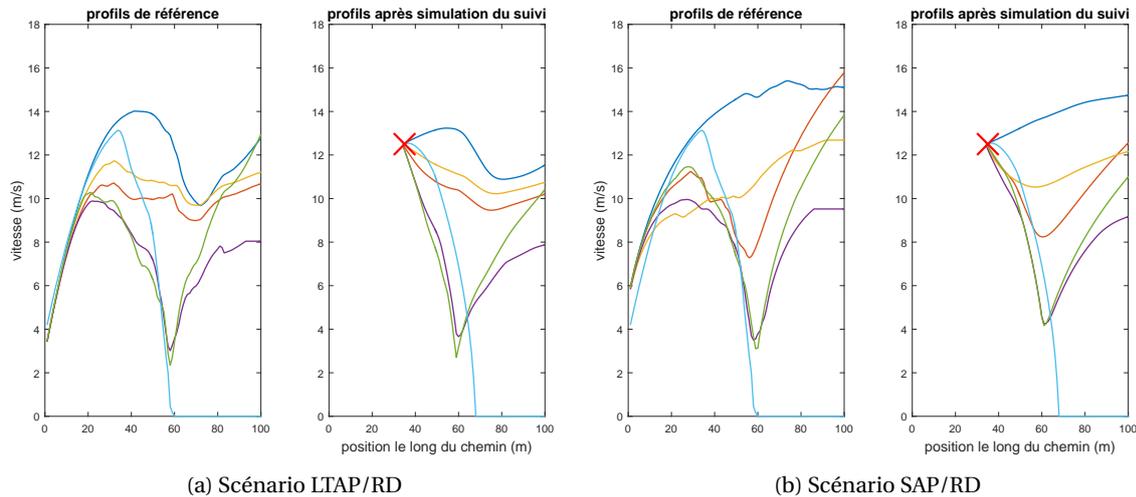


FIGURE 2.17 – Les 6 profils de référence après clustering : 3 profils d'engagement, 2 cédant le passage et 1 d'arrêt. A droite, la simulation de réponse longitudinale du véhicule suivant les références à partir d'un état initial donné (croix rouge)

2.4.3.2 Prédiction à court terme

Lorsque le VA approche une intersection alors qu'un autre véhicule veh2 se rapproche depuis une autre direction, la décision de passer avant ou après veh2 doit être prise. Pour cela, une prédiction à court terme est effectuée à chaque itération (i.e. toutes les 100ms) avec un horizon temporel de 5 secondes. Cette prédiction n'est pas nécessaire si le VA ne peut atteindre l'intersection durant l'horizon temporel. La prédiction du mouvement futur du VA utilise les profils de vitesse de référence, tandis que la manœuvre de veh2 (son chemin futur) est supposée connue et sa vitesse est supposée rester constante.

Pour pouvoir être adapté à toute situation particulière, les profils de référence calculés statiquement (voir « profils de référence » Fig. 2.17) font l'objet d'une commande prédictive (MPC) en étant utilisés comme consigne d'un module de contrôle longitudinal modélisant le contrôleur du VA. L'algorithme de contrôle s'initialise sur le vecteur d'état du VA à l'instant t et simule le futur mouvement longitudinal du véhicule en s'asservissant en vitesse sur le profil de référence. En sortie sont obtenus des profils réalistes, adaptés à la situation de l'instant t , qui peuvent effectivement être réalisés par le véhicule (voir « profils après simulation du suivi » Fig. 2.17).

A partir des prédictions de mouvement des deux véhicules, une valeur de PET est prédite pour chacun des 6 profils de vitesse adaptés et sert de métrique de risque au processus de décision. La prédiction s'effectue sur chaque profil de vitesse adapté pour le VA, afin que plusieurs réactions longitudinales soient testées sur la situation à l'instant t . La prédiction s'arrête lorsque le VA quitte la zone de collision potentielle en croisant le chemin de veh2.

2.4.3.3 Processus de décision

Le processus de décision s'appuie sur la métrique de risque objectif et sur la minimisation du temps de parcours pour le VA. Les trajectoires menant à un PET dont la valeur absolue est estimée

en dessous d'un seuil critique sont rejetées car elles sont trop risquées. Une fois que les profils adaptés les plus risqués sont filtrés, celui menant à la vitesse de consigne la plus élevée est retenu afin d'optimiser à chaque instant le temps de parcours. Dans les cas où aucun profil adapté ne peut être retenu (i.e. les cas d'urgence où toutes les options dépassent le seuil maximal de risque), le profil d'arrêt à l'intersection est automatiquement sélectionné. Cela peut permettre au VA de s'arrêter avant l'intersection, ou tout du moins réduit fortement sa vitesse. Comme mentionné précédemment, le VA passe en suivi de voie classique aussitôt qu'il se trouve sur le chemin de veh2, étant donné qu'une baisse de vitesse sur le futur chemin d'un autre véhicule est potentiellement dangereux. Finalement, le profil de vitesse retenu est envoyé au module de contrôle longitudinal.

2.4.4 Validation quantitative

Cet algorithme permet au VA de traverser les intersections de manière sûre en évitant les collisions dans les différents scénarios. La stratégie de conduite est fluide et semble répliquer le comportement humain dans la mesure de ce qui peut être perçu dans l'environnement de simulation présenté. Ce qui suit va détailler une validation objective de cet algorithme.

Pour permettre une comparaison objective, le comportement d'un VA dont la vitesse de consigne est calculée par l'approche détaillée ci-dessus est testé dans les mêmes scénarios et configurations qui ont été utilisés pour enregistrer les profils de vitesse des participants humains en 2.4.1.1. Afin de faciliter la comparaison des résultats et de mieux situer les performances de l'algorithme par rapport à celles des conducteurs humains, les dix conducteurs humains ont été répartis en deux groupes de cinq via un clustering de leurs vitesses moyennes à l'intérieur de l'intersection, sur les 56 configurations et scénarios. Par la suite, le groupe des conducteurs les plus rapides est nommé "conducteurs agressifs" et celui des plus lents "conducteurs prudents". Une modélisation de deux comportements différents VA1 et VA2 est effectuée en considérant deux valeurs différentes de PET dans le processus de décision. VA1 a une valeur de $|PET| < 1.5s$ tandis que VA2 est calculé avec $|PET| < 0.7s$.

Comme présenté plus haut, les sept scénarios ont chacun huit variantes, ce qui conduit à 56 configurations différentes. Trois métriques ont été choisies pour évaluer les performances de conduite sur les différentes configurations.

1. Le nombre de collisions (Fig. 2.18). Dans le cas des deux groupes de conducteurs humains, les valeurs du conducteur ayant eu le plus de collisions et de celui en ayant eu le moins sont affichées, ainsi que la moyenne du groupe.
2. Le nombre de situations où le véhicule ego a réussi à passer avant l'autre véhicule (Fig. 2.19). Dans le cas des deux groupes de conducteurs humains, les valeurs extrémales parmi les cinq conducteurs sont à nouveau affichées, ainsi que la moyenne du groupe.
3. La vitesse moyenne du véhicule à l'intérieur de l'intersection (Fig. 2.20). Les instants où le véhicule est à l'arrêt ne sont pas pris en compte. Pour chaque scénario, la moyenne et la déviation standard sont calculées sur les 8 configurations différentes (et sur les 5 conducteurs dans le cas des groupes).

Dans les 7 scénarios, et pour toutes leurs configurations, les VA1 et VA2 ont réussi à éviter toute collision avec l'autre véhicule. (voir Fig. 2.18). Toutefois, dans le cas des scénarios impliquant des participants humains, les véhicules de suivi ont des vitesses qui permettent de générer des situations « standardisées » au niveau des intersections. Certaines d'entre elles peuvent être dangereuses et peu habituelles pour des conducteurs humains. Le comportement de ce deuxième véhicule était donc très artificiel, ce qui explique le taux élevé de collisions pour les conducteurs humains dans les deux groupes.

Le groupe de conducteurs agressifs entre en moyenne plus souvent en premier sur l'intersection, comparé au groupe de conducteurs prudents (voir Fig. 2.19). La seule exception est le scénario de carrefour giratoire, où les valeurs moyennes sont identiques pour les deux groupes. Les deux versions de VA ont une conduite plus prudente afin d'éviter de prendre trop de risques. Alors que le VA2 qui accepte de prendre un risque un peu plus élevé passe en premier plusieurs fois dans

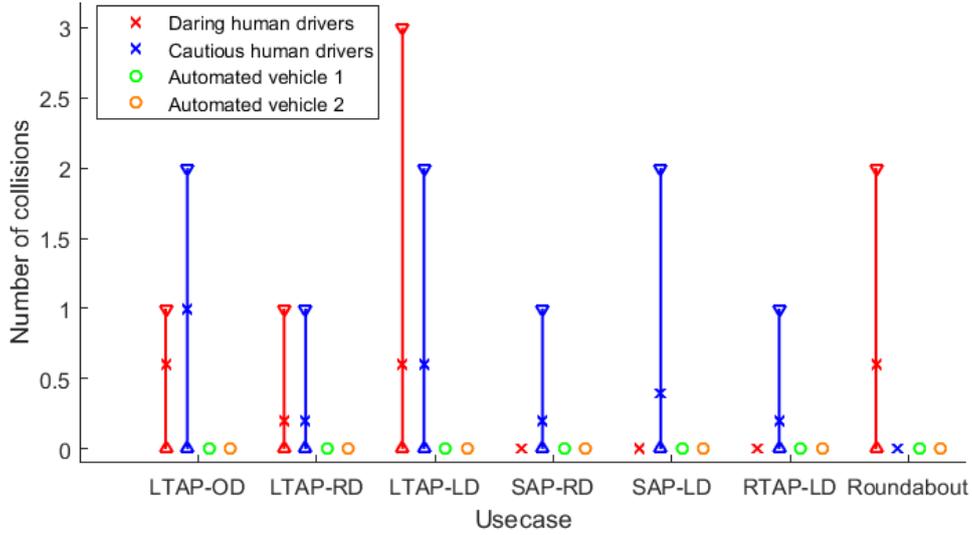


FIGURE 2.18

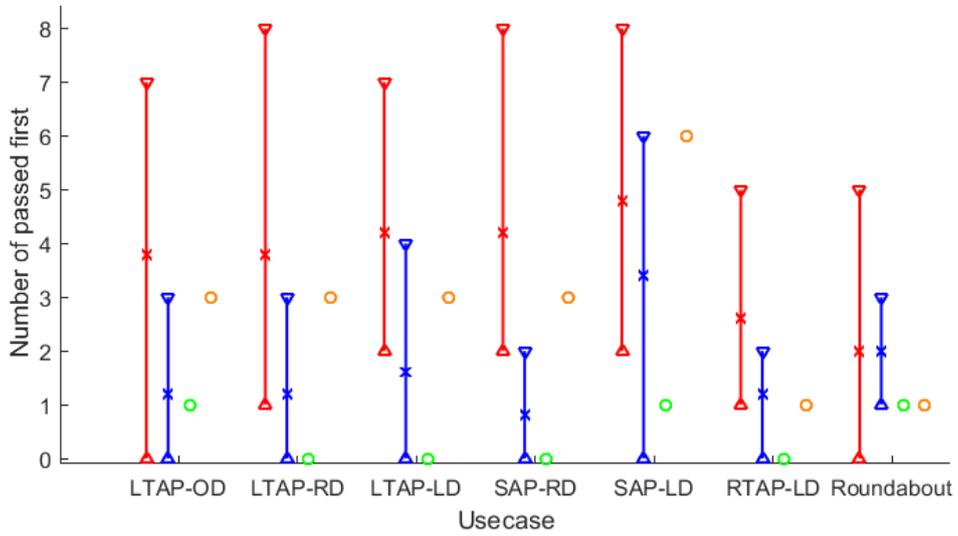


FIGURE 2.19

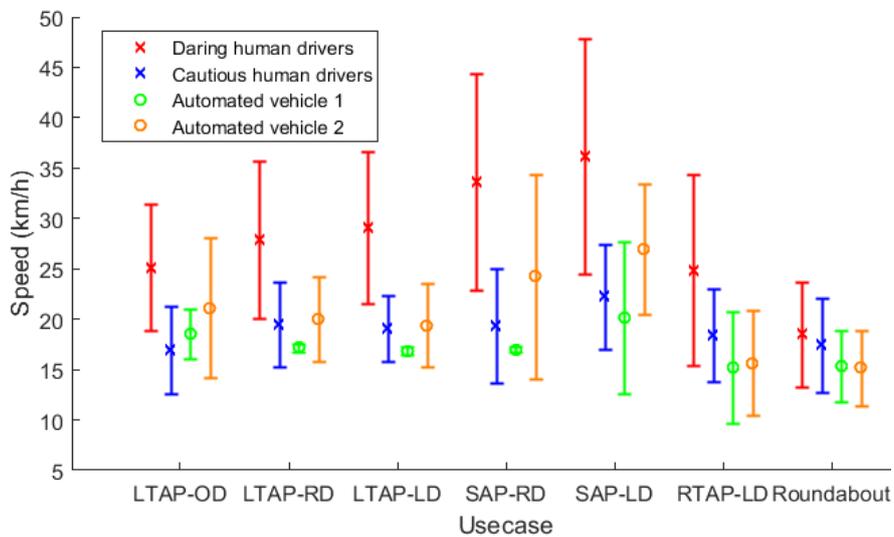


FIGURE 2.20

différents scénarios, le VA1 cède presque toujours le passage au véhicule de suivi qui arrive. Ainsi, le comportement du VA2 peut être comparé à ceux des conducteurs humains dans la mesure où il est capable de saisir une opportunité de s'insérer sur l'intersection. Ses performances à ce niveau se situent souvent entre les deux groupes de conducteurs humains. Dans le scénario SAP-LD, le VA2 arrive à s'insérer le premier encore plus souvent en moyenne que le groupe de conducteurs agressifs. Dans ce cas de figure, les conducteurs semblent laisser passer des opportunités d'insertion. Dans les situations sur carrefour giratoire, les deux versions de VA restent plus prudents et ne s'insèrent en premier qu'une seule fois pour toutes les configurations.

2.4.5 Conclusion

Un algorithme de prise de décision dédié à un véhicule automatisé circulant dans une intersection sans signalisation en présence d'un autre véhicule a été présenté. Cette approche combine un comportement longitudinal inspiré de ce que font les conducteurs humains avec un critère objectif d'évaluation du risque. Des profils de vitesse de référence sont conçus à partir de données agrégées auprès de conducteurs humains en simulation. Ces données ont permis à l'algorithme de générer un profil de vitesse qui ressemble qualitativement au comportement humain, tout en minimisant le risque encouru grâce à des critères quantitatifs.

Dans la littérature, le PET s'est avéré être un critère très utile pour estimer le risque aux intersections sans signalisation. Il a donc été utilisé comme critère de risque dans cette approche d'adaptation de vitesse, afin d'évaluer le meilleur profil de vitesse via plusieurs prédictions à court terme. Un seuil de risque acceptable peut être modulé pour modifier le comportement de l'algorithme : en acceptant peu de risque, le véhicule demeure prudent en cédant le passage dans la plupart des cas étudiés (cf. VA1, associé à une valeur de PET de 1.5s recommandée par les études sur le facteur humain), en prenant plus de risque, il peut saisir plus d'opportunités de s'insérer avant l'autre véhicule (cf. VA2, associé à une valeur de PET de 0.7s jugé nettement risqué d'après les mêmes études).

La prise de décision est faite après avoir éliminé les profils de vitesse notés comme présentant trop de risque selon le seuil de PET. Le profil le plus rapide est alors retenu, afin d'atteindre la destination au plus vite. On suppose ici que tous les profils de référence proposés sont suffisamment confortables pour pouvoir être retenus, mais il serait possible d'ajouter un critère optimisant le confort de la vitesse optimale retenue (en matière d'accélération maximale par exemple). Dans les scénarios de validation étudiés, l'approche a permis de minimiser le risque lors des interactions et a réussi à éviter toute collision, y compris dans des situations délicates pour les conducteurs humains.

2.5 Extension aux environnements semi-contraints

Considérons maintenant le cas d'un VA évoluant dans un **environnement semi-contraint** tel que défini en introduction du chapitre. Ce type d'environnement correspond à celui de certaines larges intersections parisiennes (place de la Bastille, place de l'Étoile...), des entrées des péages d'autoroutes, ou peut se retrouver sur certains parkings où le marquage au sol est absent. Bien que le code de la route s'applique toujours dans ces cas, il est bien plus difficile pour un véhicule automatisé d'identifier un ensemble discret de manœuvres, ou de planifier un chemin sans pouvoir s'appuyer sur la topologie de l'environnement.

La stratégie de décomposition chemin-vitesse introduite précédemment est généralisée à une adaptation dynamique de la vitesse dans une topologie de route semi-contrainte. Comme nous l'avons vu précédemment, un chemin est planifié a priori pour le VA, en omettant la présence des obstacles mobiles. Au lieu de suivre la voie de circulation assignée au VA comme dans le cas d'un carrefour, ce chemin est désormais tracé en considérant que l'environnement est vide de tout obstacle mobile, et en optimisant la trajectoire du VA lorsqu'il circule à sa vitesse maximale.

De nombreux aspects de l'approche d'adaptation de vitesse présentée dans la section précédente ont été modifiés afin de pouvoir fonctionner dans un environnement semi-contraint :

- La zone de collision potentielle entre le VA et un autre véhicule n'est plus définie statiquement sur la topologie de la route, mais doit être recalculée continuellement.
- La topologie de l'environnement ne permettant plus de fixer des profils de vitesse de référence en fonction de la position du véhicule par rapport à une intersection, ces références sont redéfinies continuellement en fonction de la configuration de la situation.
- Certaines limitations sont inhérentes à la métrique de risque PET lorsqu'elle est utilisée dans ce type d'environnement, l'utilisation d'une autre métrique est donc préférée.
- La validation ne peut plus s'effectuer uniquement sur des variantes longitudinales de scénarios de croisement à une intersection, il est nécessaire de définir de nouveaux scénarios de conduite permettant de valider systématiquement le comportement de l'approche.

2.5.1 Un Environnement semi-contraint

2.5.1.1 Topologie de route

Comme nous l'avons vu précédemment, certaines infrastructures de transport routier présentes au sein des grandes agglomérations diffèrent des simples intersections signalisées comme les carrefours avec des *cédez-le-passage* ou les petits carrefours giratoires, par exemple. Ces environnements comportent un vaste espace de circulation pour les véhicules, espace où le marquage au sol est quasi inexistant. Dans de tels cas de figure, l'absence de voies indiquées sur le sol réduit les restrictions imposées aux chemins suivis par les véhicules qui traversent l'infrastructure.

Le modèle d'environnement semi-contraint choisi dans ce travail est un large **rond-point** qui ressemble aux topologies de route que l'on peut retrouver sur les grandes places parisiennes, comme la place d'Italie ou la place de l'Étoile, par exemple. L'espace navigable est alors constitué d'une couronne très large sans aucun marquage au sol et reliée à différentes entrées et sorties, chacune d'entre elles pouvant posséder plusieurs voies. La largeur de la couronne est telle que plus d'une dizaine de véhicules peuvent circuler en parallèle. À l'intérieur de la couronne, l'absence de marquage au sol induit des interactions qui ne peuvent être restreintes à un petit ensemble de manœuvres précises du type « se replier sur la file de droite », « serrer à gauche », etc. comme dans les petits ronds-points.

2.5.1.2 Scénarios

L'étude et la modélisation des interactions dans un environnement semi-contraint requiert de définir des scénarios de conduite propres à ce type de topologie de route. Ces scénarios diffèrent des manœuvres classiques de changement de voie réalisées sur un carrefour en X ou sur un carrefour giratoire.

À partir d'observations effectuées sur ce type d'environnement à Paris, trois principaux types d'interactions ont été identifiés :

1. le scénario « cut-in » : un véhicule coupe la route d'un autre à l'intérieur de l'intersection.
C'est le scénario générique d'interaction entre des véhicules circulant à l'intérieur de l'environnement semi-contraint. Lorsque le VA circule à l'intérieur de l'environnement semi-contraint, sa route peut être coupée par le chemin emprunté par d'autres véhicules provenant d'un autre endroit ou se dirigeant vers une autre sortie.
2. le scénario « yield » : un véhicule doit céder le passage à d'autres.
Ce scénario se rencontre dans les intersections où la priorité n'est pas renseignée par des panneaux de signalisation. Les véhicules doivent donc respecter la priorité à droite, ce qui revient, pour ceux qui sont à l'intérieur de l'intersection, à céder le passage aux véhicules qui y entrent.

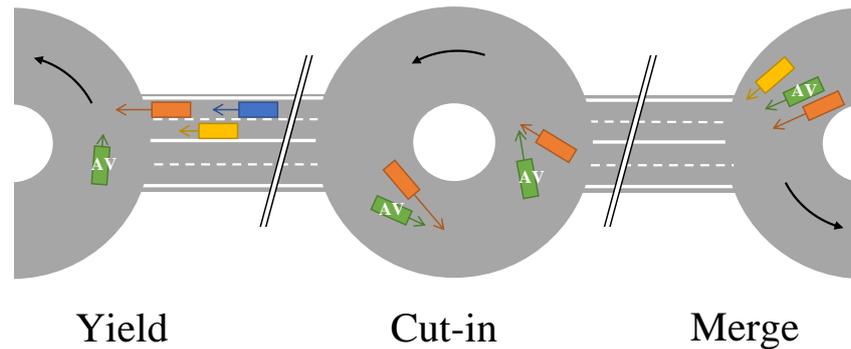


FIGURE 2.21 – Les trois scénarios considérés pour les interactions possibles entre le VA (en vert) et d’autres véhicules dans un environnement semi-contraint.

3. le scénario « merge » : plusieurs véhicules se dirigent vers la même sortie de l’intersection.

Le troisième scénario cible la transition entre l’environnement semi-contraint et un environnement contraint qui correspond aux voies de sorties sur lesquelles un marquage au sol est de nouveau présent. Lorsque le trafic est dense, le flot de véhicules tentant de s’insérer sur une même sortie à une ou deux voies peut engendrer un embouteillage. La tâche consiste, pour chaque véhicule, à choisir la voie sur laquelle la sortie s’effectuera et à s’y insérer en évitant les situations conflictuelles avec les autres véhicules.

Nous considérons dans la suite les scénarios où le VA se doit de prendre en compte le mouvement d’autres véhicules lors du calcul de son mouvement. Nous supposons que les autres véhicules respectent le code de la route. Par conséquent, les scénarios où le VA est prioritaire (par exemple, lorsqu’il entre sur l’intersection) ne sont pas traités, car c’est alors aux autres véhicules de s’adapter à la situation. La Figure 2.21 présente les trois scénarios *cut-in*, *yield* et *merge* sur une intersection de type rond-point.

2.5.2 Algorithme

L’algorithme de calcul du mouvement du véhicule VA est le même pour les trois scénarios *cut-in*, *yield* et *merge* présentés dans la section précédente. La planification dynamique de chemin n’est pas effectuée dans cet algorithme : on suppose que le chemin du véhicule VA a déjà été calculé de façon statique en ne considérant que l’espace navigable et les objets statiques et la vitesse de VA le long de ce chemin est calculée en tenant compte des interactions potentielles avec les autres véhicules. Le calcul de la planification de chemin statique est détaillé dans la section 2.5.2.1. A chaque instant t , le chemin de VA débute à la position actuelle du véhicule, respecte les contraintes cinématiques liées à son mouvement, et se termine à sa destination locale (ici, peu après la sortie de l’intersection considérée). On suppose également qu’une détection et un suivi appropriés des véhicules environnants VA, renseignant leurs positions et vitesses en temps réel, est disponible.

Étant donné un horizon temporel T_H , pour chaque pas de temps t , l’algorithme calcule le nouveau profil de vitesse pour le véhicule VA en quatre étapes :

1. S’il existe des véhicules dans le voisinage de VA, un modèle prédictif est utilisé pour calculer leurs positions futures et leurs potentielles intersections avec le chemin statique de VA durant la période $[t, t + T_H]$. Sinon, le profil de vitesse le plus rapide est utilisé pour VA et l’on passe à l’étape 4.
2. Si d’autres véhicules intersectent le chemin de VA durant la période $[t, t + T_H]$, l’ensemble des profils de vitesse de référence pour VA (présentés en section 2.5.2.4) est calculé en tenant compte des intersections du chemin de VA et des autres véhicules concernés. Dans le cas contraire, le profil de vitesse le plus rapide est utilisé pour VA et l’on passe à l’étape 4.

3. Un profil de vitesse pour VA est sélectionné parmi ceux présentant le moins de risque pour la situation donnée. Cette étape est décrite dans la section 2.5.2.6.
4. Ce nouveau profil de vitesse est utilisé par le module de contrôle longitudinal pour adapter l'accélération de VA. Ce processus est détaillé en section 2.5.2.7.

2.5.2.1 Planification de chemin

La planification de chemin est effectuée hors-ligne en fonction de l'espace navigable, de la topologie de la route, et d'une destination locale, sans tenir compte des obstacles mobiles présents dans le voisinage. Le chemin calculé est formé d'un ensemble de courbes de Bézier jointes une à une continûment (par [continuité G1](#)) depuis une position initiale jusqu'à la destination. Ce chemin est discrétisé sur un ensemble de points distribués régulièrement auxquels sont associées les valeurs de vitesse maximale en chaque point. La vitesse maximale est calculée à partir de la donnée de la vitesse limite autorisée sur l'intersection et de la courbure locale du chemin, en suivant l'équation 2.5 :

$$v_{Max}(l) = \min \left(vitesseLimite, \sqrt{\frac{acc_{max}}{\kappa(l)}} \right) \quad (2.5)$$

Où l est la position longitudinale en mètres définie le long du chemin, $acc_{max} = 3ms^{-2}$ est l'accélération maximale retenue comme critère de confort, $\kappa(l)$ est la valeur absolue de la courbure du chemin à la position l . A l'intérieur de l'intersection la vitesse limite autorisée est égale à $13ms^{-1}$ ($vitesseLimite = 13ms^{-1}$). Dans le cas d'une courbure de chemin nulle $\kappa(l) = 0$, le véhicule peut atteindre la vitesse maximale autorisée : $v_{Max}(l) = vitesseLimite$.

2.5.2.2 Modèle de prédiction

Le risque de collision est évalué en utilisant une extrapolation de la position du véhicule VA et des véhicules de son voisinage durant l'horizon de prédiction. La prédiction du mouvement des véhicules voisins de VA est donc un élément clé du processus de prise de décision. Comme nous l'avons mentionné précédemment, l'utilisation de modèles comportementaux d'interaction fondés sur des manœuvres est adaptée aux situations où les véhicules évoluent dans un environnement routier contraint (voies délimitées par des marquages au sol, etc.), ce qui permet de restreindre les degrés de liberté des véhicules. L'environnement semi-contraint n'étant pas propice à l'établissement de tels modèles, une prédiction cinématique simple a été utilisée. Les véhicules sont supposés se déplacer avec une vitesse longitudinale v et une vitesse de rotation de lacet ω constantes (Constant Turn Rate and Velocity, cf. [SCHUBERT et collab. \[2008\]](#)) durant l'horizon temporel. Dans cet environnement avec un large espace ouvert, ce modèle à rotation constante est préférable à une prédiction de trajectoire rectiligne utilisée dans la section 2.4, car les véhicules y ont plus de degrés de liberté. L'hypothèse de vitesse constante est préférée à un modèle d'accélération constante, car ce dernier est considéré trop optimiste lors du freinage. En effet, lorsqu'un véhicule s'approche d'une intersection il réduit sa vitesse, sans pour autant finir nécessairement par s'arrêter. Le modèle de vitesse constante permet d'être plus conservatif et de ne pas supposer que l'autre véhicule réagira à l'approche du VA. Le vecteur d'état de chaque véhicule à l'instant t est défini comme suit :

$$\vec{s}(t) = (x \ y \ \theta \ v \ \omega)^T \quad (2.6)$$

Les états futurs sont déduits en appliquant la fonction de transition suivante :

$$\vec{s}(t + \Delta t) = \begin{pmatrix} \frac{v}{\omega} \sin(\omega \Delta t + \theta) - \frac{v}{\omega} \sin(\theta) + x(t) \\ -\frac{v}{\omega} \cos(\omega \Delta t + \theta) + \frac{v}{\omega} \cos(\theta) + y(t) \\ \omega \Delta t + \theta \\ v \\ \omega \end{pmatrix} \quad (2.7)$$

Ce modèle de prédiction de mouvement est étendu par la suite pour inclure certaines manœuvres réactives d'évitement de collision dans des cas simples. L'hypothèse faite ici est que

chaque véhicule réagit *a minima* aux obstacles situés sur son chemin et tente d'éviter les collisions en réduisant sa vitesse de façon appropriée. Un terme longitudinal est donc ajouté au modèle de prédiction afin que le véhicule ajuste sa vitesse si un autre véhicule se trouve sur la trajectoire prédite. Ce comportement longitudinal est modélisé en utilisant le modèle de conduite « Intelligent Driver Model », (IDM) décrit dans TREIBER et collab. [2000], qui fournit une accélération continue en fonction de la vitesse du véhicule, de la distance le séparant de l'obstacle sur son chemin, et de la différence de vitesses entre les deux. Le modèle IDM est utilisé comme modèle de suivi de véhicule cible (« car following ») et n'effectue aucune anticipation de mouvement : il ne considère que ce qui est dans son chemin futur à l'instant t . La fonction d'accélération est composée d'un terme d'accélération libre $\dot{v}_{acc}(v)$ et d'un terme de freinage $\dot{v}_{brake}(g, v, \Delta v)$ qui réagit à un éventuel obstacle se situant devant :

$$\dot{v}(g, v, \Delta v) = \dot{v}_{acc}(v) + \dot{v}_{brake}(g, v, \Delta v) \quad (2.8)$$

$$\dot{v}_{acc}(v) = a \left[1 - \left(\frac{v}{v_d} \right)^\delta \right] \quad (2.9)$$

$$\dot{v}_{brake}(g, v, \Delta v) = -a \left(\frac{g^*(v, \Delta v)}{g} \right)^2 \quad (2.10)$$

où v_d est la vitesse de consigne, a l'accélération maximale que peut fournir le véhicule, δ est un paramètre contrôlant la rapidité de convergence vers la vitesse de consigne, g est la distance séparant le véhicule de celui qui le précède et $g^*(v, \Delta v)$ est la distance préférée, définie comme suit :

$$g^*(v, \Delta v) = g_{min} + v \cdot t_d + \frac{v \cdot \Delta v}{2\sqrt{ab}} \quad (2.11)$$

où g_{min} est la distance minimale désirée lorsque les deux véhicules sont à l'arrêt, $v \cdot t_d$ un délai de sécurité exprimé en distance parcourue, valable lorsque la vitesse varie peu, et $\frac{v \cdot \Delta v}{2\sqrt{ab}}$ une distance d'approche « intelligente » lorsque l'écart des vitesses est important. Les valeurs de ces paramètres sont données en Table 2.5.

| paramètre IDM | valeur |
|----------------------------------|----------------------|
| vitesse de consigne v_d | 11.1 m/s |
| délai de sécurité t_d | 1.5 s |
| distance minimale g_{min} | 2 m |
| accélération maximale a | 2.5 m/s ² |
| décélération de confort b | 4.0 m/s ² |
| exposant de convergence δ | 3 |

TABLEAU 2.5 – Paramètres du modèle IDM

L'accélération qui en résulte sert d'entrée au module de contrôle longitudinal qui met alors à jour l'état du véhicule pour la prochaine itération. La prédiction est effectuée pour chaque véhicule situé dans le voisinage du VA. Elle est recalculée toutes les 100ms en utilisant un horizon temporel T_H de 5 secondes.

2.5.2.3 Estimation du risque

Le module d'estimation de risque évalue si une situation d'interaction entre le véhicule automatisé (VA) et un véhicule obstacle (VO) est risquée. L'évaluation est quantitative et s'effectue à chaque pas de temps durant l'horizon temporel $[t, t + T_H]$.

Le PET précédemment utilisé sur des scénarios d'intersections n'est a priori pas le critère de risque le plus pertinent dans un environnement semi-contraint, car c'est une métrique spécifiquement dédiée aux intersections simples. En étudiant précisément certains cas de figure de croisement, on peut remarquer que le PET et le TG sont des mesures tout à fait équivalentes sur une

intersection à 90° (cf. Fig. 2.22a), mais que le PET pose certains problèmes dans d'autres configurations. Cette équivalence entre le PET et le TG est valable lorsque la zone de collision potentielle utilisée pour le calcul du PET est strictement égale à l'intersection des deux couloirs de passage des véhicules. C'est à partir de cette équivalence dans le cas d'une intersection à angle droit que nous avons choisi d'utiliser les mêmes valeurs de seuil de risque limite pour le TG, en suivant les conclusions de BÄRGMAN et collab. [2015] concernant le PET.

Un défaut de la métrique PET peut être observé lorsque la configuration du croisement présente un angle aigu entre les chemins des véhicules, comme on peut le voir sur la Fig. 2.22b. En effet, étant donné que le calcul du PET tient compte de la forme de la zone de collision potentielle (en rouge sur les figures), un PET de 0s peut être observé dans une situation pourtant peu risquée où les véhicules sont espacés d'une distance arbitrairement grande, en fonction de l'angle de croisement entre les chemins des deux véhicules. Au contraire, le calcul du TG tient compte implicitement de cette distance, car il mesure le laps de temps qui sépare les trajectoires des deux véhicules.

Un autre défaut du PET est illustré dans la Fig. 2.22c, où les chemins suivis par les véhicules se rejoignent en une seule direction. Dans ce cas de figure, la zone de collision potentielle peut être indéfiniment longue, de sorte que les instants t_{sort} où les véhicules sortent de la zone ne sont pas définis. Le TG permet de résoudre également cette situation car il est toujours défini lorsque deux trajectoires s'intersectent, et une valeur de TG de 0s correspond toujours à une situation de collision.

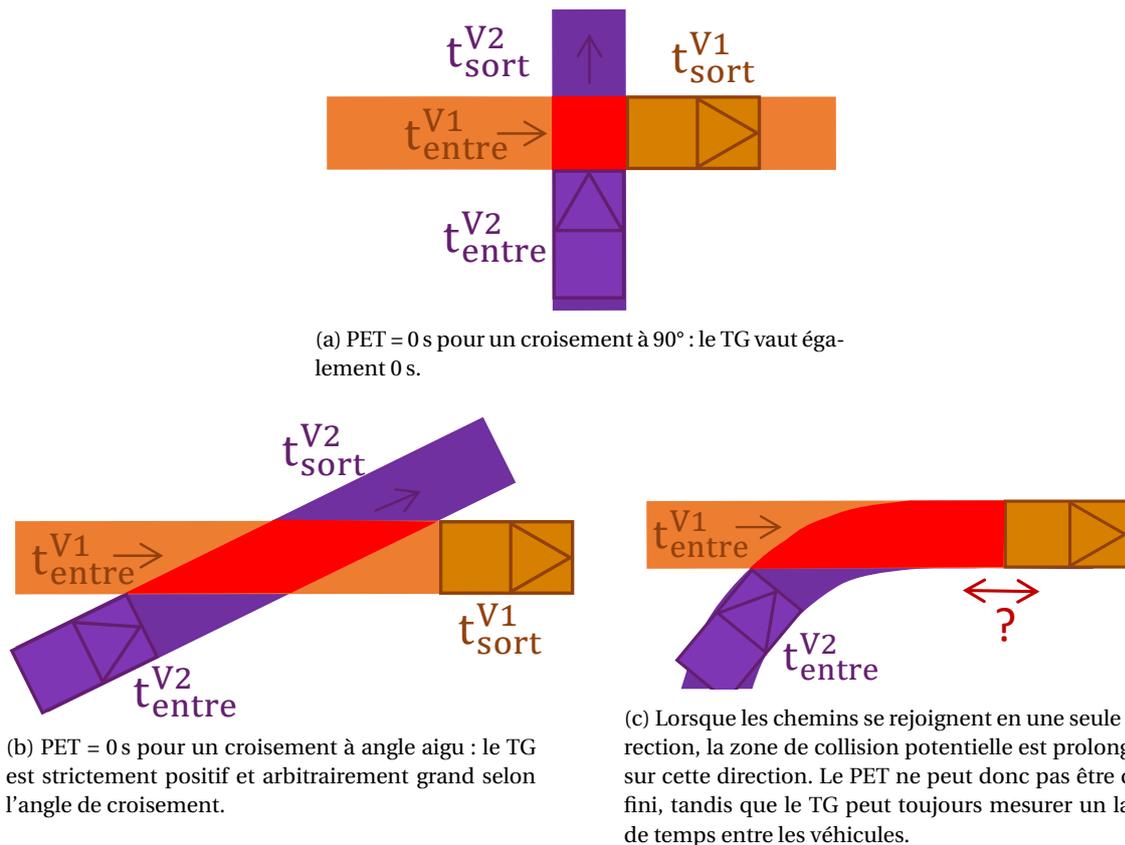


FIGURE 2.22 – Différentes situations de croisement entre deux véhicules V1 (orange) et V2 (violet). La zone de collision potentielle (rouge) est l'intersection des futurs chemins des deux véhicules.

Afin de pouvoir traiter ces différentes situations, courantes dans un environnement semi-contraint, nous avons opté pour le TG comme métrique de risque. A chaque fois que le chemin du VA intersecte une trajectoire prédite d'un VO, une zone de collision potentielle est générée. Elle correspond au polygone résultant de l'intersection des deux polygones de « couloir » créés par les chemins des véhicules élargis de leur largeur. Lorsqu'une des futures positions prédites de l'un des

véhicules entre dans cette zone alors que l'autre véhicule se rapproche également de la zone, le TG entre les deux trajectoires est calculé selon l'algorithme 1. Cet algorithme ne fait pas l'hypothèse de vitesse constante pour le véhicule approchant le croisement, mais utilise la prédiction de son profil de vitesse le long de son chemin.

Data : positionInitialeV1, positionInitialeV2, profilVitesseV2, t, horizonTemps

Result : TG(V1, V2)

positionV2 = positionInitialeV2;

for dt=0; dt < horizonTemps; dt++ **do**

if collision(positionV2, positionInitialeV1) **then**

 return dt;

else

 positionV2 += profilVitesseV2[t + dt];

end

end

return horizonTemps;

Algorithme 1 : Calcul du TG à l'instant t

Par la suite, le TG entre le VA et le VO sera dénoté $TG_{obstacle}$ et sera, par convention de signe, négatif si le VA est derrière l'autre véhicule ($TG_{obstacle} = -TG(VO, VA)$) et positif dans le cas contraire ($TG_{obstacle} = TG(VA, VO)$). Finalement, la métrique de risque quantifiant le danger de collision entre le VA et le VO à chaque instant t_0 est le TG avec la plus faible valeur absolue observée sur toute la période $[t_0, t_0 + T_H]$ (cf. équation 2.12). Cette estimation de risque est effectuée pour chacun des profils de vitesse candidats du VA qui sont présentés dans la section suivante, et pour la trajectoire prédite de chacun des autres véhicules.

$$|minTG(V1, V2)| = \min_{t \in [t_0, t_0 + T_H]} |TG(V1(t), V2(t))| \quad (2.12)$$

2.5.2.4 Profils de vitesse candidats

Dans ces travaux, un ensemble de trois profils de vitesse candidats est retenu afin de simplifier les options longitudinales possibles du VA. Ces profils sont mis à jour à chaque itération, en fonction, notamment, de la zone de collision potentielle la plus proche :

1. Le profil « de croisière » : le véhicule accélère jusqu'à atteindre la vitesse maximale indiquée par le processus de planification pour la portion de chemin concernée.
2. Le profil « constant » : le véhicule maintient indéfiniment sa vitesse à sa vitesse instantanée.
3. Le profil « d'arrêt » : le véhicule s'approche de la prochaine zone de collision potentielle et effectue un freinage confortable pour s'arrêter juste avant celle-ci. Il est à noter que ce profil n'induit pas forcément un freinage immédiat du véhicule, notamment lorsque la prochaine zone de collision potentielle est éloignée.

Le profil « de croisière » prend en chaque point du chemin la valeur $v_{Max}(l)$ définie par l'équation 2.5, sauf lorsqu'il est nécessaire d'effectuer une régulation de distance (ou Adaptive Cruise Control en anglais : ACC) afin d'adapter la vitesse à celle d'un autre véhicule plus lent se trouvant sur le futur chemin du VA. Dans ce cas particulier, le profil « de croisière » est ajusté comme suit :

$$\left\{ \begin{array}{l} \text{if } dist > 2 \cdot dist_{cible}, \\ \quad v_{croisière}(l) = v_{Max}(l) \\ \text{if } dist_{cible} \leq dist \leq 2 \cdot dist_{cible}, \\ \quad v_{croisière}(l) = v_{cible} + \frac{dist - dist_{cible}}{dist_{cible}} \cdot (v_{Max}(l) - v_{cible}) \\ \text{if } dist < dist_{cible}, \\ \quad v_{croisière}(l) = v_{cible} \cdot \frac{dist}{dist_{cible}} \end{array} \right. \quad (2.13)$$

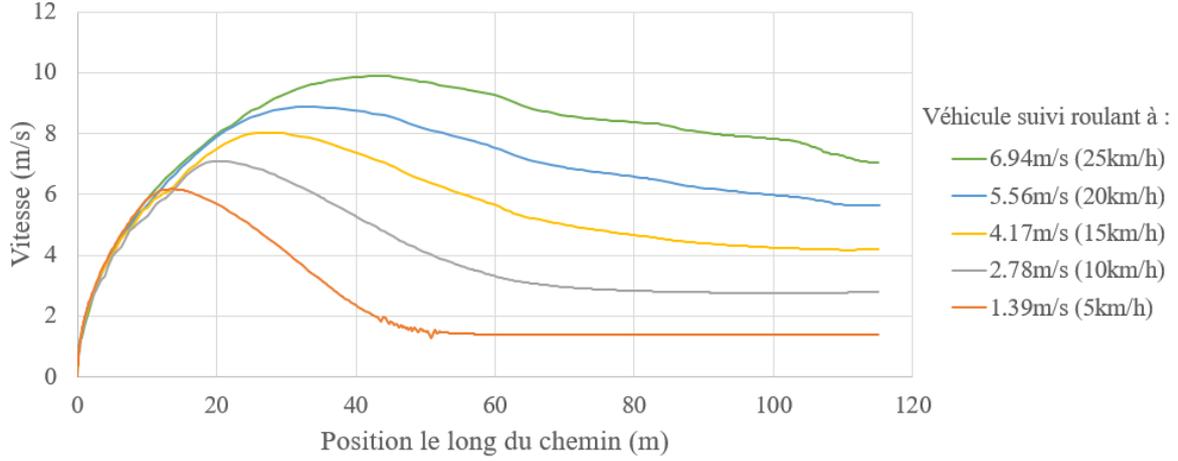


FIGURE 2.23 – Comportement de la régulation automatique de distance (ACC) pour le profil de vitesse « de croisière » du VA derrière un véhicule cible qui avance à différentes valeurs de vitesse constante.

Où $dist$ est la distance entre les deux véhicules, v_{cible} est la vitesse du véhicule ciblé par l'ajustement de vitesse et $dist_{cible} = 2 + 0.5 \cdot v_{VA}$ est la distance consigne à respecter. Des exemples de profils de vitesse qui résultent de cette régulation de distance effectuée par le VA en simulation sont montrés sur la Fig. 2.23.

Le profil « constant » est supposé conserver indéfiniment la vitesse instantanée v_{VA} du VA, à ceci près que cette vitesse ne peut excéder $v_{croisière}$. Ainsi, pour chaque position longitudinale l , $v_{constant}(l) = \min(v_{constant}(0), v_{croisière}(l))$.

Le profil « d'arrêt » doit permettre un arrêt contrôlé avant la prochaine zone de collision potentielle. Il impose une décélération constante $Acc_{Freinage} = -4.5 m s^{-2}$. Ce profil de vitesse est défini par les équations 2.14 et 2.15, où $dist_{zone}$ est la distance restante entre le VA et la zone moins une marge fixée à 1 m.

$$dist_{Arrêt} = \frac{v_{VA}^2}{2 \cdot |Acc_{Freinage}|} \quad (2.14)$$

$$\begin{cases} \text{if } dist_{zone} > dist_{Arrêt}, & v_{arrêt}(l) = v_{VA} \\ \text{if } 0 \leq dist_{zone} \leq dist_{Arrêt}, & v_{arrêt}(l) = v_{VA} \cdot \frac{dist_{zone}}{dist_{Arrêt}} \\ \text{if } dist_{zone} < 0, & v_{arrêt}(l) = 0 \end{cases} \quad (2.15)$$

Les trois profils présentés précédemment ne sont pas directement utilisés dans le processus d'évaluation du risque détaillé dans la section 2.5.2.3 mais sont chacun donnés comme consigne à un module interne MPC simulant le contrôle longitudinal du véhicule (voir section 2.5.2.7). De nouveaux profils dits « candidats » sont alors générés : ils sont initialisés à la vitesse instantanée du VA et suivent ensuite le profil de consigne en respectant le modèle de véhicule simulé au sein du module de contrôle. On obtient ainsi des profils de vitesse réalistes qui rendent compte au mieux de la réponse longitudinale du véhicule (notamment par rapport à ses limitations d'accélération et d'à-coup ou de jerk). Ces profils « candidats » sont alors utilisés dans le calcul du TG.

2.5.2.5 Interactions avec plusieurs véhicules

L'estimation de risque introduite ci-dessus n'est valable que pour une interaction entre deux véhicules, le VA et un autre. Elle peut être généralisée en une approche plus complète qui tient compte d'un nombre indéterminé de véhicules. Les véhicules dans le voisinage du VA dont les trajectoires prédites n'intersectent pas le chemin du VA en une zone de collision potentielle ne sont pas considérés par la suite. L'hypothèse est faite qu'un profil complet peut être construit en combinant autant de profils de référence qu'il y a de véhicules considérés, et en supposant comme précédemment qu'un seul profil de référence est suffisant pour interagir convenablement avec

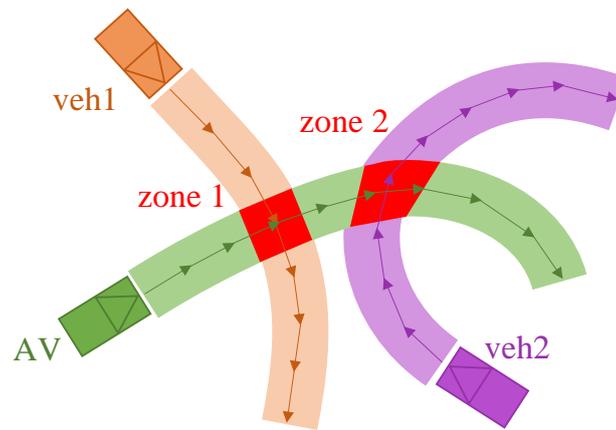


FIGURE 2.24 – Schéma d'une situation avec deux interactions successives : les trajectoires prédites des deux autres véhicules intersectent le chemin du VA durant la période $[t, t + T_H]$ en formant deux zones de collision potentielle.

chaque véhicule. Afin d'éviter que la complexité de l'algorithme n'explose avec le nombre de véhicules, ceux-ci sont ordonnés selon la proximité de leur zone de collision potentielle après projection sur le chemin du VA. Les interactions entre le VA et les autres véhicules sont ainsi considérées séquentiellement une à une.

Comme nous l'avons vu dans la section 2.5.2.4, à chaque interaction plusieurs profils de référence sont considérés pour le VA. Lorsque le VA doit interagir avec plusieurs véhicules durant l'horizon temporel $[t, t + T_H]$, un arbre est construit pour calculer les différentes combinaisons de profils de référence pour le VA. L'arbre a une hauteur N égale au nombre de zones de collision potentielle présentes sur le futur chemin du VA. Les nœuds de cet arbre représentent les vecteurs d'état des véhicules en présence (position, vitesse, accélération) et chaque arête correspond à une hypothèse de profil de référence du VA durant une interaction avec un autre véhicule. La hauteur d'un nœud dans l'arbre correspond ainsi au numéro de la zone de collision associée à l'interaction à l'origine du nœud. L'arbre a au plus C^N feuilles, où C est le nombre de profils de référence. L'ensemble des chemins possibles dans l'arbre reliant la racine à une feuille correspond à l'ensemble des profils complets possibles, qui combinent différents profils de référence du VA. Pour chaque interaction du VA avec un autre véhicule et pour chaque profil de référence testé, une prédiction partielle est effectuée jusqu'à la prochaine zone de collision potentielle afin de décrire le nœud enfant. Avant de calculer la prédiction partielle représentée par une arête de l'arbre, les profils de référence doivent être adaptés (cf section 2.5.2.4) à la situation intermédiaire représentée par le nœud parent. Cette situation comprend les vecteurs d'état des véhicules et la position de la zone de collision potentielle. La transition d'une référence de profil de vitesse à une autre s'effectue juste après avoir passé la première zone de collision potentielle et les continuités de la vitesse et de l'accélération sont garanties par le module simulant la réponse du contrôleur longitudinal (cf section 2.5.2.7). Un TG est ensuite calculé (cf section 2.5.2.3) pour cette prédiction partielle et sauvegardé au niveau du nœud enfant. L'arbre est développé jusqu'à la dernière zone de collision potentielle détectée sur l'horizon temporel.

Pour illustrer cette approche, considérons l'exemple de la Fig. 2.24 qui schématise une situation avec deux véhicules (*veh1* et *veh2*) qui croisent le futur chemin du VA. Les polygones *zone1* et *zone2* sont les zones de collision potentielle de ces deux interactions. En considérant les trois profils de vitesse de référence pour le VA (« croisière », « constant » et « arrêt »), l'arbre affiché sur la Fig. 2.25 représente les différentes combinaisons possibles.

Dans cet arbre, le nœud racine n_0 représente la situation initiale au temps t . Les nœuds intermédiaires n_C , n_K et n_A correspondent aux situations obtenues après les prédictions partielles

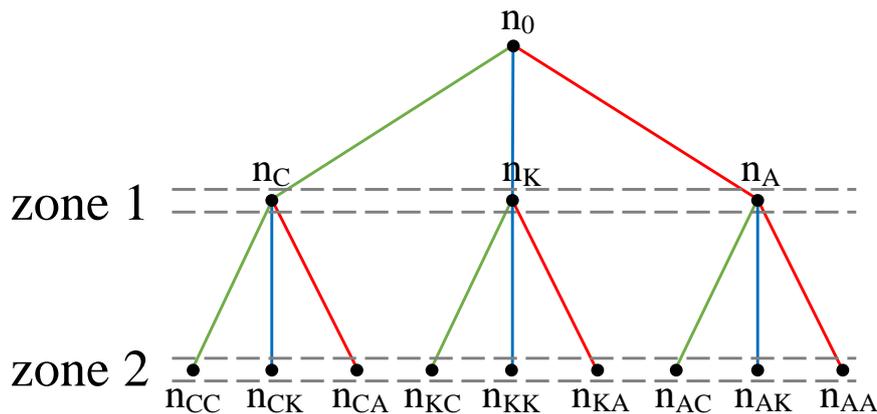


FIGURE 2.25 – L'arbre des combinaisons possibles entre les différents profils de références, dans une situation avec deux zones de collision potentielle. Les arêtes vertes, bleues et rouges correspondent respectivement aux prédictions partielles associées aux profils de vitesse « croisière », « constant » et « arrêt ».

pendant lesquelles le VA a suivi les profils de référence « croisière », « constant » et « arrêt », respectivement. Dans cette situation intermédiaire, l'interaction avec le premier véhicule est terminée car l'un des deux véhicules (VA ou *veh1*) a dépassé la zone de collision potentielle. Les neuf nœuds feuilles correspondent aux états possibles après les deux interactions successives. Le nœud n_{CA} , par exemple, caractérise la situation dans laquelle le VA se trouve lorsqu'il a maintenu sa vitesse en suivant le profil « de croisière » le temps de passer la *zone1* (on peut ainsi imaginer qu'il est probablement passé avant *veh1* mais ce n'est pas certain), puis qu'il a freiné selon le profil « d'arrêt » afin de céder le passage à *veh2*. La prédiction des positions des deux véhicules à chaque pas de temps permettra d'évaluer le risque de ce comportement, notamment si le VA a le temps de freiner confortablement entre les deux zones. Dans l'arbre de la Fig. 2.25, douze arêtes représentant des prédictions partielles sont affichées, aboutissant à neuf profils de vitesse combinant chacun deux références. La Fig. 2.26 montre les différentes prédictions de profils de vitesse pour le VA et les positions longitudinales des nœuds feuilles. Les feuilles n_{AK} et n_{AA} sont positionnées avant la *zone1* : elles représentent deux combinaisons de profils où le VA s'arrête avant la première zone et reste à l'arrêt jusqu'à l'horizon temporel. Les trois feuilles n_{CC} , n_{CK} et n_{KC} partagent la même vitesse à la sortie de la *zone2*, mais il est important de noter que les trois profils ne parviennent pas forcément à cette position au même instant. La feuille n_{AC} représente la combinaison du profil « d'arrêt » avant la *zone1* jusqu'à laisser passer *veh1* qui continue en profil « de croisière » pour réaccélérer ensuite. Lorsqu'il quitte la *zone2*, ce profil global est moins rapide que celui menant en n_{CC} . Le profil associé à la feuille n_{CA} avait pour objectif d'effectuer un arrêt avant la *zone2*, mais le profil de référence indique que la décélération de freinage, qui ne commence qu'après avoir passé la *zone1*, ne sera pas suffisante pour que le VA s'arrête avant d'être sur le chemin de *veh2*. Cette situation est donc potentiellement dangereuse en théorie, mais cela dépend toujours de l'état de *veh2* lorsque le VA entre sur la zone. Dans tous les cas, seule la mesure du TG sur la simulation temporelle des trajectoires des deux véhicules pourra déterminer le niveau de risque d'un profil de vitesse.

2.5.2.6 Prise de décision

L'objectif de l'étape de décision est de choisir le profil de vitesse de référence (« croisière », « constant » ou « arrêt ») à appliquer immédiatement, en réaction à la première zone de collision potentielle sur le chemin du VA. Pour autant, cette décision tient compte de l'intégralité de l'arbre décrit ci-dessus pour évaluer le risque des trois premiers profils de référence. Ainsi, le risque associé à chacun des trois premiers profils de référence dépend des interactions prédites sur les autres zones de collision potentielle. Afin de calculer le TG associé à chaque profil de vitesse de réfé-

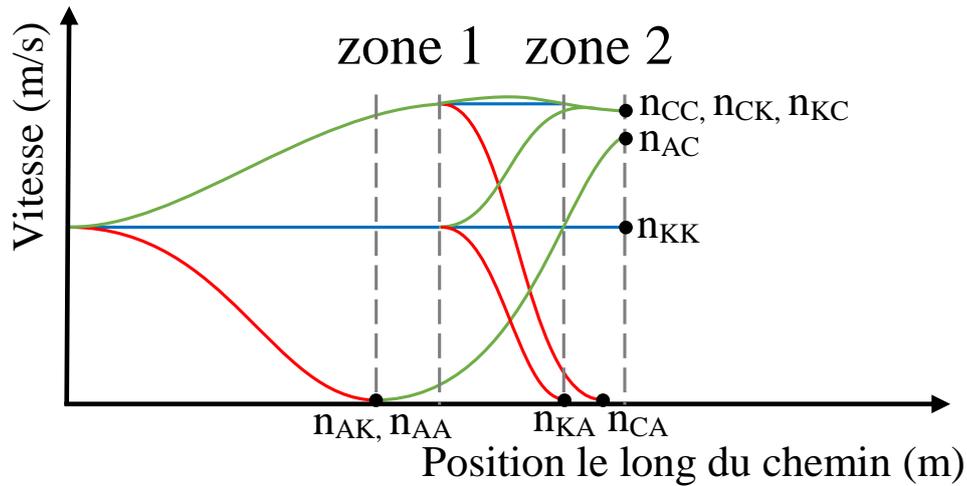


FIGURE 2.26 – Les combinaisons possibles de profils de vitesse pour deux zones de collision potentielle avec les trois profils de référence (vert pour « croisière », bleu pour « constant » et rouge pour « arrêt »).

rence, le TG minimum est propagé depuis les feuilles vers la racine via un parcours en profondeur (de type postfixe) utilisant cette règle :

$$TG(n) = \min \left(TG(n), \max_{n_e \in \text{enfants}(n)} (TG(n_e)) \right) \quad (2.16)$$

Au niveau d'un nœud n , l'opérateur maximum est utilisé pour sélectionner parmi ses enfants le profil partiel le moins risqué pour le VA (avec un TG plus élevé en valeur absolue). Le TG du nœud n est mis à jour via l'opérateur minimum afin de considérer le pire cas entre ce qui se passe durant et après l'interaction correspondant au nœud n . A la fin du parcours, chacun des trois profils de référence de la première interaction est associé à une valeur de TG correspondant à l'instant le plus critique de la branche la moins risquée qui part de cette première référence. Cette valeur tient donc compte de toutes les prédictions effectuées pour les futures interactions du VA avec les autres véhicules une fois cette première référence suivie.

Les références trop risquées, c'est à dire associées à un TG tel que $\|tg\| < \text{minTG}$ sont écartées et le processus de décision sélectionne ensuite le profil optimal parmi les options restantes. La valeur minTG est le paramètre principal du comportement de l'algorithme d'adaptation de vitesse et permet d'ajuster le seuil de risque maximal pris par le véhicule. L'algorithme est testé avec deux valeurs : 1.5s pour imiter un conducteur relativement prudent dans un contexte où il n'a pas la priorité et 0.7s pour un comportement qui va plus souvent tenter de s'insérer au risque de se rapprocher de la trajectoire d'un autre véhicule. Ce dernier comportement est approprié dans le cas où le VA a la priorité, mais peut être perçu comme trop agressif si le VA est censé céder le passage devant un autre véhicule, comme c'est souvent le cas dans les scénarios testés ici. Pour ces travaux, seuls la vitesse et le risque sont considérés, c'est pourquoi le profil retenu est toujours le plus rapide parmi ces options. D'autres critères de performance tels que l'économie d'énergie ou le confort de conduite auraient également pu être utilisés. Le processus de décision fournit en sortie un des trois profils de vitesse de référence pour la première interaction.

2.5.2.7 Module de contrôle

Le profil de vitesse de référence est donné en entrée au module de contrôle longitudinal qui pilote la vitesse du véhicule. L'accélération de consigne est calculée selon l'équation 2.17.

$$Acc_{target} = a \left(1 - \left(\frac{v_{VA}}{v_{profil}(0)} \right)^3 \right) \quad (2.17)$$

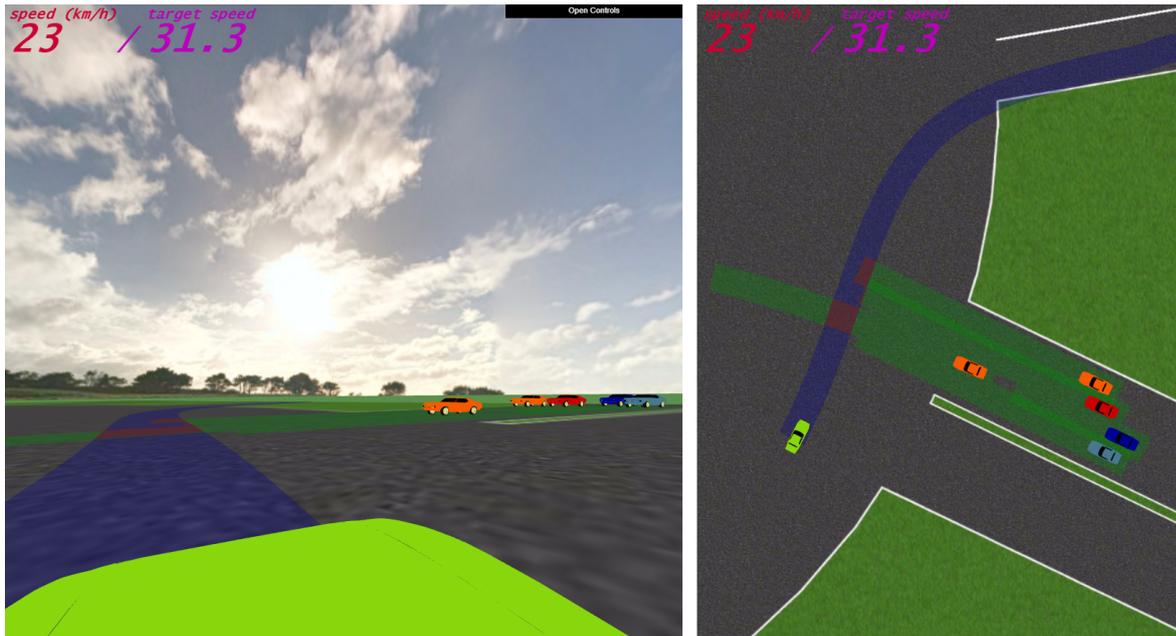


FIGURE 2.27 – Aperçu de la topologie de route semi-contrainte dans le simulateur. A gauche la vue depuis le VA, à droite une vue d'en haut. Le VA est vert, son chemin est en bleu transparent. Les trajectoires prédites des autres véhicules sont en vert transparent et les zones de collision potentielle en rouge.

Où v_{VA} est la vitesse instantanée du VA et $v_{profil}(0)$ est la première valeur de vitesse du profil, à la position instantanée du VA. Par ailleurs, un contrôleur latéral ajuste l'angle de lacet du véhicule afin de suivre le chemin statique pré-calculé en minimisant l'accélération latérale et l'écart en distance au chemin de référence.

2.5.3 Validation

Plus de 3000 situations différentes de conduite dans un environnement semi-contraint ont été simulées afin de valider les décisions prises par l'algorithme. Chacun des trois scénarios présentés section 2.5.1.2 ont été déclinés en larges gammes de configurations différentes afin d'évaluer les performances et les limitations de l'approche en matière de sécurité.

La Fig. 2.27 montre une vue du simulateur utilisé lors des tests avec la topologie de route d'environnement semi-contraint utilisée pour les trois scénarios (« cut-in », « yield », et « merge »).

Deux valeurs différentes ont été utilisées pour le seuil de risque maximum acceptable afin de varier le comportement du VA : une configuration avec un TG de 0.7s pour un comportement plus agressif et une autre avec un TG de 1.5s pour un comportement plus prudent.

2.5.3.1 Cut-in

Dans ces scénarios « cut-in », le chemin du VA va croiser celui d'un autre véhicule avec un angle aigu. Ces situations correspondent aux cas où les deux véhicules au sein de l'intersection non supervisée ont des destinations différentes et où leurs chemins se croisent à une distance plus ou moins critique. Dans certains cas, ce sont des situations de « queue de poisson » où le VA doit être très réactif pour éviter d'entrer en collision avec un véhicule particulièrement agressif. Afin de tester le processus de décision, le VA est placé dans l'environnement de simulation au milieu d'un espace navigable libre sans marquage au sol, en présence d'un véhicule obstacle (VO). Aux deux véhicules sont attribués des chemins qui se croisent en une zone de collision potentielle. Il existe 12 configurations possibles pour le chemin du VO : 2 types de virage (chemin rectiligne ou incurvé) et 6 angles de croisement entre les 2 chemins. Par symétrie du scénario, seuls les cas où le VO arrive sur la droite du VA sont considérés. Ces 12 variantes de chemin pour le VO ainsi que le chemin du VA sont illustrés Fig. 2.28.

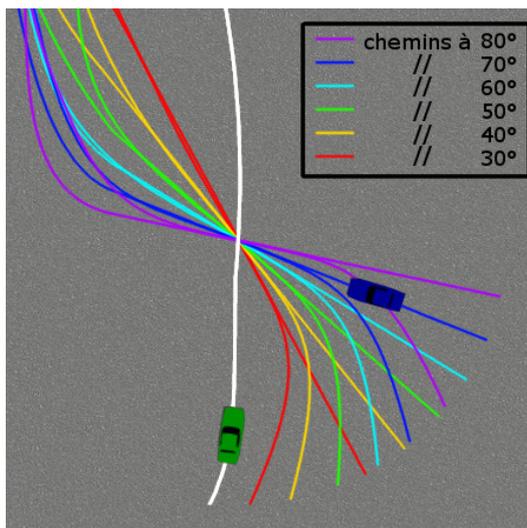


FIGURE 2.28 – Les 12 variantes de chemins pour le VO dans les scénarios « cut-in ». Le chemin du VA est en blanc.

Étant donné le modèle de prédiction de trajectoire utilisé, on s'attend à ce que les interactions en présence d'un chemin incurvé soient les plus critiques. De même, la situation est plus délicate à gérer lorsque l'angle entre les deux chemins est aigu. Afin de créer des situations plus ou moins critiques, les positions et vitesses de départ des deux véhicules varient. La position de départ du VA est choisie en fonction de sa vitesse initiale : il est éloigné de la zone de collision potentielle d'une distance égale à sa distance de freinage augmentée d' 1 m . De cette façon, il est physiquement possible d'éviter d'entrer en collision à partir de l'état initial dans toutes les configurations. Pour autant, cet arrêt peut ne plus être possible si l'algorithme tarde à freiner suite à une mauvaise décision : quelques centaines de millisecondes suffisent à rendre certaines collisions inévitables. Cinquante configurations initiales sont créées dans l'intervalle de vitesse $[5.0\text{ m/s}, 11.9\text{ m/s}]$ avec les positions de départ correspondantes. Au total, 1200 simulations sont effectuées (2 comportements du VA, 2 types de virage pour le VO, 6 angles d'incidence pour le VO, 50 configurations initiales).

2.5.3.2 Yield

Dans le scénario « yield » on considère une flotte de véhicules qui entrent sur l'environnement semi-contraint alors que le VA circule à l'intérieur du **rond-point**. Il est donc censé leur céder le passage car la priorité à droite s'applique. Dans ce type d'environnement, déterminer précisément les interactions conflictuelles et adapter sa vitesse en fonction des priorités sont des tâches particulièrement difficiles du fait de l'absence de tout marquage au sol. La flotte de véhicules obstacles est constituée de quatre véhicules A, B, C et D arrivant en parallèle sur les quatre voies de la même entrée du rond-point : chacun se voit donc attribuer un chemin statique différent. Deux chemins statiques différents sont testés pour le VA. Sur chacun, quatre configurations de départ (position et vitesse) sont utilisées. Comme pour le scénario précédent, les positions de départ sont choisies de sorte qu'il soit encore possible pour le VA de s'arrêter avant la première zone de collision potentielle détectée. Les configurations initiales des véhicules obstacles (VO) sur leurs chemins respectifs sont aussi variées : 63 combinaisons différentes des 4 positions et 4 vitesses initiales sont tirées au hasard. La Fig. 2.29 montre les 2 variantes de chemins possibles pour le VA ainsi que les 4 chemins des VO. Au total, 1008 simulations sont effectuées (2 comportements du VA, 2 chemins du VA, 4 configurations initiales du VA, 63 configurations initiales des 4 VO).

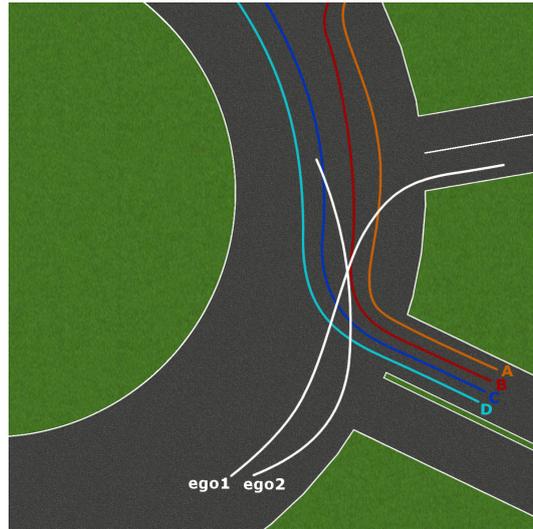


FIGURE 2.29 – Les quatre chemins pour la flotte de véhicules obstacles (A-D) entrant sur l’intersection dans les scénarios « yield » et les 2 chemins disponibles pour le VA qui doit leur céder le passage (en blanc).

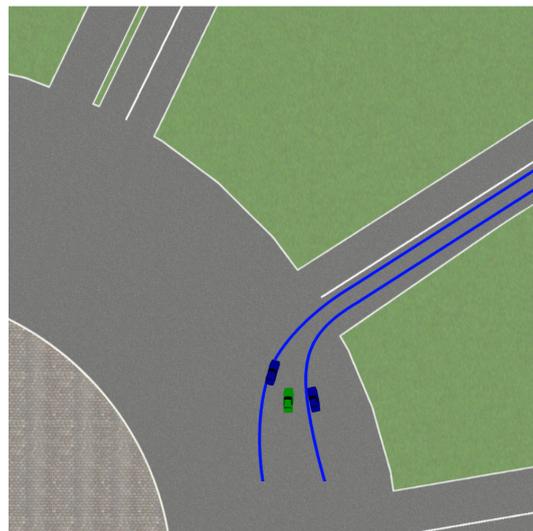


FIGURE 2.30 – Les chemins des deux véhicules obstacles qui sortent du rond-point (en bleu). Le VA (en vert) doit s’insérer en choisissant une des deux voies de sortie.

2.5.3.3 Merge

Le scénario « merge » traite le cas de la transition entre l’espace semi-contraint qu’est le rond-point et une de ses sorties, où les voies sont clairement délimitées par un marquage au sol. Le VA sort du rond-point par une rue à deux fois deux voies, au moment où deux véhicules obstacles situés de part et d’autre du VA se dirigent vers la même sortie. Un des VO sort en empruntant la voie de gauche et l’autre utilise celle de droite. Cette situation avec trois véhicules s’insérant sur deux voies force le VA à choisir la voie la plus appropriée à sa manœuvre, ainsi qu’à adapter sa vitesse afin d’éviter toute collision avec les véhicules obstacles. Une vue de haut montre les chemins utilisés dans le scénario dans la Fig. 2.30.

Dans ce scénario, l’algorithme de planification de mouvement ne doit pas seulement effectuer une adaptation de vitesse sur un chemin donné, mais doit également choisir la voie optimale sur laquelle s’insérer. La planification de trajectoire est modélisée de manière simple en utilisant 2 chemins pré-calculés statiquement sur lesquels l’approche d’adaptation de vitesse est appliquée. Parmi ces deux chemins, le meilleur est choisi à chaque itération et sert comme consigne pour le contrôleur latéral. Une hystérésis est utilisée lors du choix du chemin, afin d’éviter les oscilla-

tions durant la transition entre les deux voies. Comme précédemment, les configurations de départ des trois véhicules varient afin de créer des situations ayant différents niveaux de dangerosité. On considère 100 configurations de départ pour le couple de VO, avec des vitesses tirées aléatoirement dans l'intervalle $[8.3m/s, 13.9m/s]$ et 5 configurations de départ pour le VA, avec les vitesses 8.3, 9.7, 11.1, 12.5 et $13.9m/s$. Pour chaque vitesse, la position de départ est calculée comme précédemment afin de permettre au VA de s'arrêter avant de croiser le chemin d'un autre véhicule. Au total, 1000 simulations sont effectuées (2 comportements du VA, 5 configurations initiales du VA, 100 configurations initiales pour les VO).

2.5.4 Résultats

La validation de l'approche est uniquement effectuée au niveau de la sécurité de la conduite. Le TG introduit précédemment est utilisé comme critère quantifiant le risque. Les événements critiques tels que les collisions ou les conflits entre véhicules ayant évité de justesse une collision sont reportés. Ceci permet d'évaluer la performance de l'algorithme en matière de sécurité dans des scénarios critiques et d'envisager les améliorations possibles.

Il est important de noter que l'on distingue par la suite le calcul des collisions, effectué en temps réel par le simulateur lorsque les maillages 3D de deux véhicules se touchent, et le calcul du TG minimal entre deux véhicules, effectué a posteriori à partir des trajectoires enregistrées et un modèle 2D simplifié des véhicules (rectangle orienté, ou boîte englobante). Si les rectangles orientés en 2D de deux véhicules s'intersectent au cours d'un scénario enregistré, le TG indique le laps de temps minimum entre ces deux rectangles. Cette précision a son importance pour expliquer certains résultats : un TG égal à 0 sur un scénario n'implique pas forcément qu'il y a eu une collision (même si la réciproque est vraie). Un TG de 0 sans collision peut être dû à la discrétisation temporelle des enregistrements à une fréquence de 10Hz (les TG inférieurs à $100ms$ sont ainsi approximés à 0s), ou au fait que les rectangles orientés ne sont qu'une approximation en 2D des maillages des véhicules, et donc peuvent s'intersecter lorsque les maillages sont proches mais ne se touchent pas.

Certains enregistrements de scénarios de validation sont présentés dans une vidéo qui a été mise en ligne : <https://youtu.be/70TKJovhpXo>.

2.5.4.1 Cut-in

Les valeurs de TG pour les 1200 configurations du scénario « cut-in » sont reportées sur la Fig. 2.31. Les valeurs négatives indiquent que le VO a passé la zone de collision en premier (le VA lui a cédé le passage), et les valeurs positives indiquent que le VA est passé en premier (cf. section 2.5.2.3). L'absence d'occurrences autour de la valeur 0 signifie qu'il n'y a eu ni collision ni situation vraiment dangereuse. En effet, la valeur de TG la plus faible de toutes les configurations est $0.5s$ (12 cas où le VA cède le passage et 1 cas où il passe en premier). Cette valeur de $0.5s$ est légèrement inférieure à la consigne donnée pour le comportement du VA « agressif ».

En comparant les TG des deux types de comportement du VA, les valeurs négatives (le VA cède le passage) montrent deux distributions centrées sur les valeurs de consigne de $0.7s$ pour le comportement « agressif » et $1.5s$ pour le comportement « prudent ». Dans les configurations de type « agressif » le VA peut passer en premier dans 207 cas, alors que quand il est « prudent » il passe en premier dans 155 cas. Les valeurs de TG positives semblent dispersées sur l'intervalle $[0.5s, 3.5s]$ sans qu'il soit pour autant possible de déceler clairement un motif de distribution. Cela s'explique par le fait qu'à partir du moment où le VA commence à prendre un profil de vitesse correspondant à la décision de céder le passage au VO, le profil de vitesse du VO devient prépondérant dans la valeur finale du TG. Les cas de figures les plus notables correspondent aux situations où le VA doit s'arrêter avant l'intersection, et ne peut redémarrer qu'une fois l'autre véhicule passé. Dans ce cas, la valeur du TG dépend exclusivement du temps mis par le VO à quitter l'intersection (plus le temps que le VA redémarre et traverse l'intersection en l'absence de toute interaction, temps qui ne varie pas d'un scénario à l'autre).

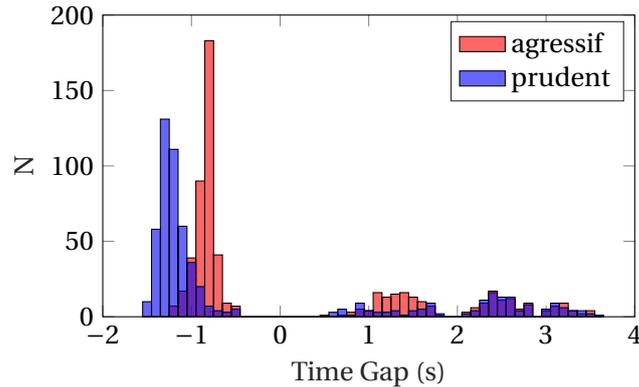


FIGURE 2.31 – Comparaison des deux histogrammes des valeurs de TG entre les deux comportements du VA pour le scénario « cut-in »

Puisque les valeurs faibles de TG indiquent un risque plus élevé, elles sont examinées plus en détails. Il y a au total 36 configurations avec un TG dont la valeur absolue est égale ou inférieure à 1.0s. La majorité de ces situations se produisent dans des configurations de VA avec le comportement « prudent » et lorsque le chemin du VO est courbé (64%). Dans ces configurations à risque élevé le comportement « agressif » est présent dans 13 cas (26%) avec un TG qui vaut systématiquement 1.0s : 9 situations lorsque le chemin du VO est rectiligne et 4 lorsqu'il est courbé. Les différents angles de croisement montrent la même prévalence dans ces configurations à risque élevé (18% pour chaque angle), sauf pour l'angle de 30° qui n'apparaît seulement que dans 3 cas. Dans ces configurations à risque élevé, on observe également une forte corrélation entre le type de chemin du VO et les conditions initiales des véhicules (position et vitesse). Les configurations avec un chemin courbé pour le VO sont critiques le plus souvent lorsque les vitesses des deux véhicules sont élevées en début de scénario. Les configurations avec un chemin rectiligne pour le VO sont critiques le plus souvent lorsque la vitesse du VA est nettement plus élevée que celle du VO.

Pour les configurations menant à une valeur négative du TG (le VA cède le passage au VO), une faible valeur absolue de TG indique également un risque élevé. Il y a 24 cas critiques avec une valeur absolue inférieure à 1.0s : 12 cas avec $-0.5s$ et 12 cas avec $-0.6s$. Le comportement « agressif » n'est que légèrement majoritaire dans ces situations critiques (58% et 75% respectivement). Les configurations avec un chemin courbé et un angle d'incidence de 30° sont de loin les plus critiques. Ils correspondent à 10 cas pour le TG de $-0.5s$ et 9 cas pour le TG de $-0.6s$. Dans les rares situations critiques où le chemin du VO est rectiligne, on constate des angles d'incidence plus élevés et des conditions initiales similaires à celles où le VA passe en premier, c'est à dire des vitesses initiales élevées pour les deux véhicules (en moyenne $11.1 m/s$).

2.5.4.2 Yield

Dans ce scénario « yield » le VA cède le passage presque toujours à tous les véhicules qui entrent sur l'intersection, mais dans certains cas il décide de s'insérer avant certains véhicules de la flotte. Dans la majorité des configurations le VA doit s'arrêter complètement le temps de laisser passer les autres véhicules. Les positions où le VA s'arrête le plus souvent sont affichées en rouge Fig. 2.32, avec les chemins effectivement parcourus par les véhicules. Un rectangle orienté semi-transparent qui schématise les dimensions du véhicule a été placé sur chaque position où le VA est à l'arrêt, et toutes les configurations sont ainsi superposées. Il apparaît que la majorité des arrêts a lieu avant le couloir du véhicule D (celui circulant dans la voie la plus proche du VA). Cependant, d'autres zones concentrent les arrêts du VA, notamment entre les couloirs des véhicules B et C : l'algorithme a pu identifier cette zone comme étant sûre et s'y arrêter après avoir traversé le couloir du véhicule C le temps de laisser passer le VO B. Bien que la zone I-C soit située sur le couloir du VO D, elle ne correspond pas à un cas de collision car le véhicule D est déjà passé lors de l'arrêt du VA qui cède alors le passage au VO C. Cette zone d'arrêt n'est donc pas risquée en soi.

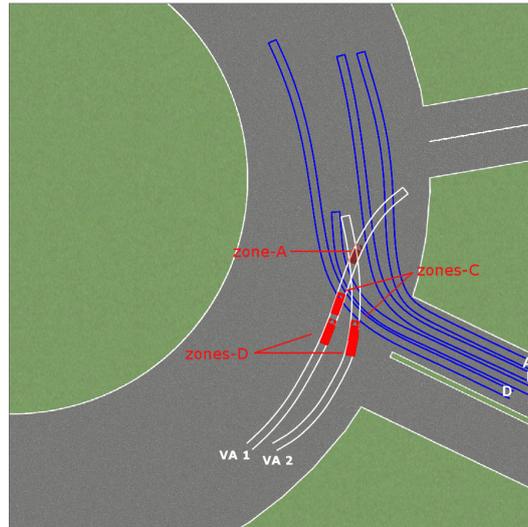


FIGURE 2.32 – Les positions d’arrêt du VA (en rouge) sur le scénario « yield », l’opacité du rouge indiquant la redondance. Sont également affichés les chemins parcourus par les quatre véhicules obstacles (en bleu) et les deux chemins disponibles pour le VA (en blanc). Ces couloirs sont retracés à partir des positions effectives dans les enregistrements.

Le tableau 2.6 donne une vue d’ensemble des différentes issues possibles dans le scénario « yield » : le VA s’insère avant certains VO de la flotte et cède le passage aux autres. Sur toutes les simulations effectuées, un seul cas d’accident a été observé. Dans cette configuration, le VA suit le comportement « agressif » et passe avec succès avant les véhicules D et C avant de freiner en urgence à l’arrivée du véhicule A sans réussir à éviter la collision. La collision a lieu entre le flan gauche du VO A et l’avant du VA, à une vitesse d’impact de $1.2m/s$. Dans la grande majorité des cas, le VA cède le passage en toute sécurité aux véhicules entrants qui ont la priorité. Les cas où le VA passe avant au moins un autre véhicule sont surtout induits par le comportement « agressif ». Le nombre de cas d’insertion du VA avant un autre VO avec le comportement « prudent » est assez faible (23 occurrences), et a systématiquement lieu lorsque le véhicule en question est le VO D (id 3 sur le tableau). Cela s’explique par la position initiale de celui-ci, souvent largement éloignée de celle du VA. Le TG dans ces situations est de l’ordre de $1.7s$ en moyenne ($std = 0.1s$). Les TG des cas 1 à 5 du tableau sont présentés Fig. 2.33. Il y a deux situations critiques avec un TG de valeur absolue inférieure à $0.4s$ dans le cas d’identifiant 5, l’une des deux situations étant le cas de collision vu précédemment. Dans les deux cas, le comportement « agressif » du VA le pousse à s’engager rapidement pour passer successivement sur les couloirs des véhicules D, C et A. La situation semble a priori sans danger mais la succession des trois zones de collision potentielle empêche par la suite le VA de s’arrêter avant d’avoir franchi complètement le couloir du VO C. Dans ces deux situations, le VO A a accéléré entre-temps pour s’élancer sur l’intersection et son contrôleur longitudinal ne tient pas compte de l’arrivée du VA sur son flan gauche. La situation devient alors critique par l’enchaînement de ces événements. Les autres situations où le VA décide de s’insérer devant un ou plusieurs autres véhicules sont gérées sans se terminer de façon critique.

Il est important de noter que les situations dangereuses n’ont lieu qu’en présence d’un VA conduisant avec le comportement « agressif ». Comme mentionné dans la littérature en « facteurs humains », un TG de $0.7s$ est perçu comme très agressif par un conducteur ou un passager humain. Ce type de comportement n’est clairement pas approprié dans une situation où le code de la route spécifie explicitement que le véhicule doit céder le passage au trafic entrant sur la droite. Une piste d’amélioration dans ce type de scénario consisterait à prendre en compte localement les règles du code de la route et à les modéliser en modifiant certains paramètres tels que le TG minimum autorisé.

| identifiant | description du résultat | nombre d'occurrences |
|-------------|-------------------------------------|----------------------|
| -1 | collision | 1 |
| 0 | cède le passage à tous | 911 |
| 1 | s'insère avant le véhicule A | 8 |
| 2 | s'insère avant le véhicule C | 8 |
| 3 | s'insère avant le véhicule D | 74 |
| 4 | s'insère avant les véhicules D et A | 1 |
| 5 | s'insère avant les véhicules D et C | 5 |

TABLEAU 2.6 – Nombre d'occurrences des différentes situations finales du scénario « yield »

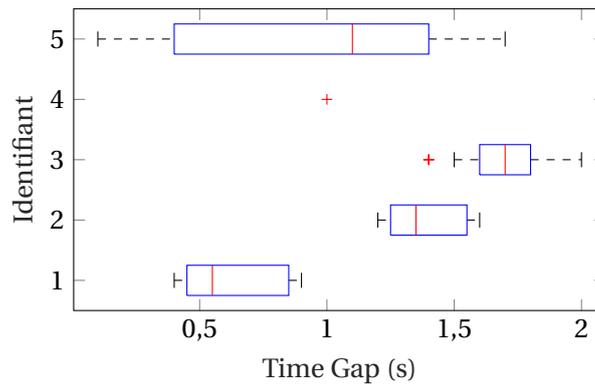


FIGURE 2.33 – TG des différents cas de figure du scénario « yield » (cf les identifiants de la Table 2.6)

2.5.4.3 Merge

Un histogramme du TG de toutes les configurations du scénario « merge » est présenté sur la Fig. 2.34. Le TG final retenu pour une configuration impliquant deux autres véhicules est le plus critique des deux (i.e. le TG ayant la plus petite valeur absolue). La majorité des valeurs négatives (le VA cède le passage) pour le comportement « agressif » se concentre autour de $-1.0s$, tandis que pour le comportement « prudent » le pic se situe entre $-1.5s$ et $-2.0s$. Cependant la distribution observée semble moins claire que pour le scénario « cut-in », ce qui s'explique par la présence ici de deux autres véhicules : le VO responsable du TG affiché n'est pas forcément le premier véhicule avec lequel le VA a interagi. Comme précédemment, les valeurs positives sont difficiles à interpréter car elles dépendent essentiellement du comportement du VO après le passage du VA. Le VA réussit à s'insérer 129 fois avant les deux autres véhicules, 19 fois avant l'un d'entre eux en cédant le passage à l'autre, et cède le passage aux deux dans 852 configurations.

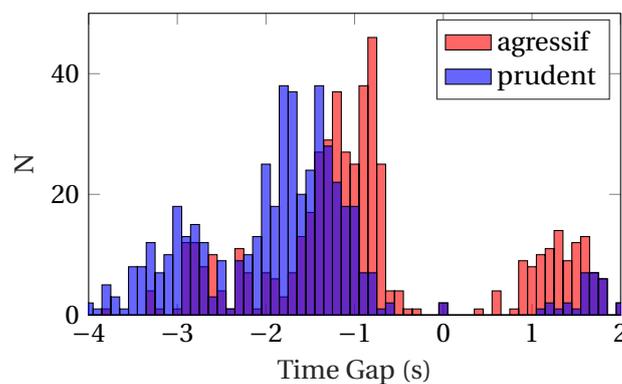


FIGURE 2.34 – Histogramme du Time Gap sur toutes les configurations du scénario « merge », pour les deux comportements du VA

Les 1000 configurations simulées se sont déroulées sans aucune collision. Cependant, la valeur du TG indique 0s dans 4 cas. Comme nous l'avons remarqué précédemment, il y a une différence entre les rectangles orientés en 2D et la forme exacte des véhicules dans le simulateur 3D. D'une façon générale il y a plus de cas critiques, c'est à dire de cas ayant une valeur de TG inférieure à 0.5s en valeur absolue par rapport aux deux autres scénarios. Après une investigation plus poussée des enregistrements, il s'avère que ces cas au TG très faible peuvent être subdivisés en deux groupes : un « groupe A » de 5 situations non critiques dans lesquelles les réactions des différents véhicules semblent correctes et un « groupe B » de 2 situations véritablement dangereuses dans lesquelles un des conducteurs a manqué d'anticipation.

Dans le « groupe A », il y a deux cas de figures avec $TG = 0s$ sans qu'aucune collision n'ait été observée. L'une de ces situations est représentée par la séquence de rendus Fig. 2.35. Le VA (en vert) est plus lent en roulant en moyenne à $5.4m/s$ que le VO (en bleu) qui circule à $10.5m/s$. Celui-ci dépasse le VA par la gauche et sort du rond-point en utilisant la voie de gauche du bras de sortie. En rejouant l'enregistrement et en l'observant visuellement sans tenir compte des différentes métriques, la situation ne semble pas particulièrement critique. Le fait que $TG = 0s$ est dû à la superposition des rectangles (boîtes englobantes) du VA et du VO qui sont élargies d'une marge par rapport aux véritables dimensions des véhicules. Cette marge permet d'améliorer la métrique de TG en compensant d'éventuelles erreurs de mesure dues à la discrétisation temporelle induite par le simulateur. Il se peut que cette marge ne soit pas adaptée aux situations comme celle-ci où les véhicules circulent côte à côte avec une faible inter-distance latérale. Sur la Fig. 2.35, la superposition des deux boîtes englobantes a lieu entre les 3^e et 4^e images, alors que l'avant gauche du VA est très proche de l'arrière droit du VO. Les autres cas de figure du « groupe A » décrivent globalement la même situation, notamment causée par une définition non adaptée de la métrique de risque.

Dans le « groupe B », le TG semble mesurer correctement le niveau de risque des deux situations. Quand bien même il n'y a pas de collision, les véhicules sont trop proches et la situation est dangereuse, comme on peut l'observer sur la Fig. 2.36. Dans cette situation le VA compte sortir du rond-point par la voie la plus à droite en s'insérant devant le VO. Au début de la séquence, le VA circule avec une vitesse de $6.8m/s$ et le VO à $5.0m/s$, ce dernier étant situé à l'arrière du VA sur sa droite. La situation devient peu à peu critique car le modèle de prédiction de comportement utilisé (cf section 2.5.2.2) n'anticipe pas les accélérations des autres véhicules. En l'occurrence, le VO accélère en suivant son profil de vitesse prédéfini. Lorsque le VA réalise que la vitesse du VO rend la situation dangereuse pour continuer le profil « de croisière », il est déjà trop tard pour que le profil « d'arrêt » puisse freiner à temps. Par ailleurs, la dangerosité de la situation est amplifiée par deux facteurs : d'une part l'absence de toute anticipation de la part du VO (il ne réagit qu'aux véhicules situés sur son chemin, et ne freine donc que tardivement) et d'autre part le peu de marge de manœuvre que permet le comportement « agressif ». En effet, lorsque le profil de vitesse est estimé être trop risqué, l'un des points de la trajectoire du VA est à moins de 0.7s d'une collision. De manière similaire, le second cas de figure du « groupe B » montre aussi les limitations du modèle de prédiction utilisé. Dans cette autre situation le VA cède cette fois le passage au VO qui sort du rond-point. Mais le VA n'anticipe pas le virage soudain du VO et sa décélération s'avère insuffisante pour maintenir une inter-distance de sécurité raisonnable. Là encore, on peut affirmer qu'une conduite plus adaptée de la part du VO lui aurait fait adopter un virage plus large pour éviter de trop se rapprocher d'un autre véhicule. Cependant, ce type d'interaction doit pouvoir être géré par l'algorithme du VA. Une piste d'amélioration du modèle de prédiction serait de tenir compte de la topologie de la route pour calculer des trajectoires les plus probables pour les différents véhicules en présence. Cela est particulièrement valable pour le scénario « merge » où le rayon de courbure d'un véhicule peut changer soudainement afin de suivre une voie de sortie, ce qui semble pouvoir s'anticiper relativement facilement.

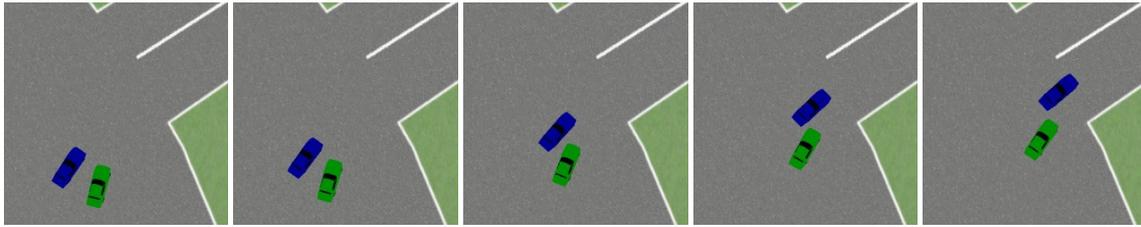


FIGURE 2.35 – Séquence de rendus du simulateur pour un cas de figure du « groupe A » en présence du VA (vert) et du VO le plus proche (bleu) sur le scénario « merge ». La séquence dure 1.2s.

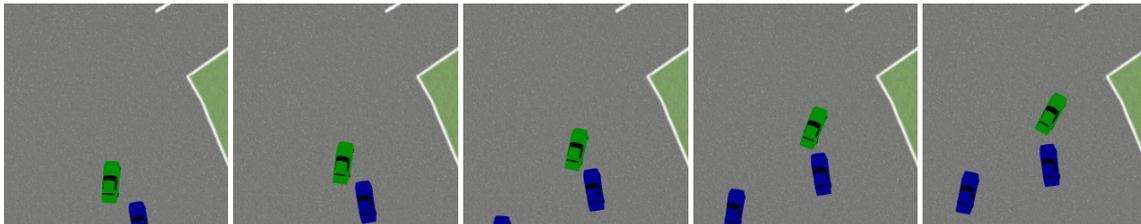


FIGURE 2.36 – Séquence de rendus du simulateur pour un cas de figure du « groupe B » en présence du VA (vert) et du VO le plus proche (bleu) sur le scénario « merge ». La séquence dure 1.2s.

2.5.4.4 Interprétation générale

D'une manière générale, ces résultats attestent la pertinence de notre approche fondée sur le TG qui permet d'interagir de manière fluide et sûre avec les autres véhicules sur un environnement de conduite semi-contraint. Les scénarios mis en place ont pour fonction de confronter le VA à des situations typiques que l'on peut trouver sur ce type d'intersection sans marquage au sol. Une validation « systématique » a été effectuée en itérant sur des centaines de cas de figure particuliers pour chaque scénario. Pour traiter toutes ces configurations, le même algorithme de planification de vitesse été employé.

Bien que doté d'une prédiction simple pour anticiper le mouvement des autres véhicules (extrapolation physique), le VA évite les collisions et les situations dangereuses dans la grande majorité des cas. Souvent la dangerosité de la situation est renforcée par le fait que les véhicules obstacles ne réagissent pas à la présence du VA. On peut néanmoins remarquer que le modèle de prédiction est la principale cause d'échec de l'algorithme. Celui-ci pourrait notamment tirer parti du contexte pour améliorer les trajectoires prédites pour les autres véhicules. En effet, la présence d'une sortie du rond-point près d'un autre véhicule devrait rendre plus plausible les trajectoires s'écartant rapidement du rond-point afin d'en sortir. De même, la proximité d'une voie d'entrée change la priorité du VA par rapport aux véhicules sur sa droite, son comportement devrait donc s'ajuster en ce sens.

Enfin, nous avons constaté l'influence majeure de la valeur minimale autorisée pour le TG sur le comportement du VA. Les performances des deux comportements (« agressif » vs « prudent ») l'attestent : lorsque le $minTG$ est fixé à $0.7s$ le VA s'insère beaucoup plus souvent devant d'autres véhicules qu'avec un $minTG$ de $1.5s$. Le comportement « agressif » est également impliqué dans la majorité des situations conflictuelles, et dans toutes celles vraiment dangereuses. On peut en conclure que la valeur $0.7s$ pour le TG minimum est trop risquée et ne devrait pas être utilisée en pratique, du moins lorsque le VA n'a pas la priorité. Ce résultat est en accord avec notre hypothèse de départ, et les conclusions des études en facteur humain (cf. BÄRGMAN et collab. [2015]), prônant plutôt une valeur située autour de $2.0s$ dans ce type d'interactions.

2.5.5 Remarques sur les limites de la validation

Dans cette section, nous discutons de certaines limites inhérentes à la notion de validation sur des situations de conduite qui impliquent d'autres véhicules. Dans la littérature sur les véhicules automatisés, il est difficile de trouver beaucoup de références à une validation systématique

d'un algorithme de planification de trajectoire sur des scènes dynamiques dans un environnement urbain. De plus, l'environnement semi-contraint qui est au centre des présents travaux rend la planification de trajectoire d'autant plus complexe tant à concevoir qu'à évaluer.

D'une manière générale en robotique, les tests en situation réelle sont préférés aux simulations pour des raisons évidentes de réalisme. Pour autant, seule la simulation permet de générer de larges quantités de scénarios en itérant sur les valeurs de chaque paramètre. Ceci a pour principal avantage d'augmenter les chances de tester les rares cas de figure qui sont vraiment problématiques : les zones grises où l'algorithme oscille entre deux décisions cruciales. De plus, certaines situations peuvent être trop dangereuses pour être testées en conditions réelles. C'est notamment le cas de situations délibérément conçues pour tester les cas critiques : parfois, la configuration de départ peut mener les véhicules à entrer en collision de façon quasi certaine. A certains égards, l'approche en simulation peut donc être plus pertinente pour tester efficacement et de façon sûre un algorithme de planification dans des situations critiques, sans que cela ne vienne supplanter la nécessité de tests en conditions réelles pour les scénarios de conduite plus courants.

Nous pensons qu'il est indispensable de tester la robustesse d'un algorithme de décision d'un véhicule automatisé en générant des scénarios qui varient la topologie de la route, le nombre d'obstacles mobiles (avec éventuellement différents types de véhicules), leurs vitesses, leurs distances au VA, leurs « comportements », leurs destinations, etc. Cela permet d'itérer sur autant de configurations que possible, y compris les plus dangereuses même si elles peuvent être rares. Cette approche évalue de manière systématique ce dont l'algorithme de décision est capable sur un large panel de situations de conduite, et permet d'en dresser un bilan quantitatif.

Ce type de validation, qui confronte l'algorithme à un large panel de variantes de scénarios prédéfinies, nous semble plus pertinent que certaines métriques plus répandues, comme le délai parcouru sans conflit ou le nombre de kilomètres entre deux désengagements du système autonome (*miles between disengagements*), qui sont couramment utilisées dans l'industrie de la voiture autonome. Ces métriques tendent notamment à « diluer » le danger que constitue une réponse inappropriée de l'algorithme à une situation donnée lorsque la probabilité d'occurrence de cette situation est faible. Cela revient à confondre le risque que l'algorithme prenne une mauvaise décision lors d'une situation donnée avec le risque que cette situation critique ne se produise. En particulier, les spécificités de l'environnement dans lequel le véhicule évolue ne sont pas prises en compte. Un kilomètre parcouru sans conflit particulier sur autoroute ne peut pas être comparé à un kilomètre en milieu urbain. De même, le passage d'une intersection régulée par des feux de signalisation, même en présence de plusieurs voies parallèles et de passages piétons, ne peut pas être comparé au fait de traverser une intersection sans signalisation. A fortiori, l'environnement semi-contraint traité ici génère encore plus d'interactions diverses avec les autres véhicules, chacune pouvant s'avérer plus ou moins critique.

Par ailleurs la validation d'un algorithme réagissant au comportement des autres véhicules suppose que ces autres véhicules réagissent à leur environnement de façon appropriée. En effet, l'activité de conduite est dans une certaine mesure fondée sur une compréhension mutuelle entre les usagers de la route, sur la supposition que les autres respectent les règles de conduite, et qu'ils soient capables d'anticiper raisonnablement nos actions. Dans cette section, nous avons choisi d'utiliser le modèle IDM (*Intelligent Driver Model*) pour animer le comportement des autres véhicules. De cette façon, les véhicules interagissant avec le VA affichent une réaction minimale et ne respectent pas aveuglément le profil de vitesse qui leur a été désigné. Cependant, ils n'anticipent pas du tout le mouvement des autres véhicules. C'est en soi un problème pour la validation de la planification dynamique de trajectoire si on veut tenir compte des interactions entre les véhicules.

Une autre approche pour tester un algorithme pourrait être de l'appliquer sur tous les véhicules de la scène. En effet, cela permettrait aux autres véhicules de faire également preuve d'anticipation et donc de réduire le nombre de conflits. En effet, l'optimisation d'un tel algorithme collaboratif est un sujet de recherche en soi, bien que le critère consistant à éviter les conflits dangereux soit relativement simple à réaliser. Cependant, l'objet de cette validation est d'évaluer la performance de l'algorithme dans un environnement inconnu constitué de véhicules non coopé-

ratifs et non communicants. Quoi qu'il en soit, le choix du modèle approprié à utiliser pour simuler les autres véhicules afin d'obtenir des conditions de validation satisfaisantes est loin d'être résolu et pourrait bien rester un problème ouvert encore longtemps.

2.5.6 Conclusion

L'approche de planification de trajectoire générique présentée dans cette section est capable d'ajuster dynamiquement la vitesse d'un véhicule automatisé dans différents scénarios se tenant dans un environnement semi-contraint. Elle combine plusieurs hypothèses de profils de vitesse générées et adaptées en permanence selon la situation. Une métrique de risque évalue plusieurs prédictions organisées en un arbre de décision lorsque le VA est en présence de plusieurs autres véhicules.

L'utilisation du TG comme critère de risque dans les situations d'intersection fait partie des contributions de l'algorithme. Cette métrique est généralement réservée à des contextes longitudinaux, où le VA suit un autre véhicule. On lui préfère souvent le TTC ou le PET, présentés comme plus sophistiqués mais pourtant largement inadaptés à l'évaluation du risque dans de nombreuses situations.

L'algorithme a fait l'objet de tests de validation en simulation sur trois scénarios typiques de l'environnement considéré, avec en tout plus de 3000 configurations différentes nécessitant une décision immédiate du VA afin d'éviter que la situation ne devienne critique. L'objectif de ces tests est d'effectuer ce qui s'approche le plus d'une « validation systématique » des performances de l'approche en matière de sécurité.

Les résultats montrent globalement une bonne robustesse de l'algorithme lorsqu'il est confronté à des situations critiques variées. Quelques cas de situations dangereuses n'ont toutefois pas pu être évités, et démontrent que le seuil de TG fixé à 0.7s est trop dangereux pour des scénarios où le VA est censé céder le passage aux autres véhicules. Au-delà du fait que les TG inférieurs à 1.0s sont de toute façon considérés comme trop risqués du point de vue des passagers humains, il est à noter que les conditions initiales des tests de validation étaient spécifiquement conçues pour générer des situations de conflits, à la limite des cas de collisions physiquement inévitables.

Enfin, les tests en simulation permettent de valider l'approche générale consistant à adapter continuellement la vitesse du véhicule, au lieu de prendre une décision à moyen terme en choisissant une manœuvre parmi un ensemble discret. Cela permet une certaine souplesse de la navigation : le VA peut s'insérer dans la circulation en adaptant sa vitesse de façon fluide, sans avoir à se stopper lorsqu'il décide de céder le passage. Même si ce comportement peut être observé en simulation (cf. <https://youtu.be/70TKJovhpXo>), des expérimentations sur véhicules ont permis de valider qualitativement cet aspect de l'approche en situation réelle.

2.6 Tests sur véhicules réels

L'algorithme introduit dans la section 2.5 a été implanté sur un véhicule automatisé de l'équipe RITS et a pu faire l'objet de tests sur le campus d'Inria Rocquencourt.

L'objectif était de tester la validité de l'approche en situation réelle. Bien qu'il n'était pas possible d'utiliser les véhicules équipés sur route ouverte, et donc dans un environnement semi-contraint, l'approche a néanmoins pu être testée sur une intersection non supervisée ne comportant aucun marquage au sol (cf. Fig. 2.37).

Le scénario de validation, impliquant deux véhicules, a les caractéristiques suivantes :

- Le VA est une voiture C1 rouge équipée d'un système « drive-by-wire » lui permettant d'être directement pilotée par l'approche d'adaptation de vitesse. Un GPS à correction RTK mesure le vecteur d'état du véhicule, c'est à dire sa position, sa vitesse et son angle de lacet.
- L'autre véhicule est une voiture C3 verte qui est conduite manuellement. Il est équipé des mêmes capteurs que la C1 afin de mesurer son vecteur d'état. Les deux véhicules peuvent communiquer entre eux via une connexion « V2V » sur le standard IEEE 802.11p.

- La vitesse maximale des véhicules a été fixée à 40 km/h, les véhicules n'étant pas autorisés à circuler à une vitesse supérieure dans le campus.
- L'algorithme de planification, ainsi que l'architecture logicielle automatisant le véhicule sont interfacés via le framework RTMaps, et l'ensemble est exécuté par un ordinateur embarqué dans le véhicule.
- Les deux voitures démarrent en même temps aux positions indiquées sur la Fig. 2.37. Les chemins qui leur sont assignés se rejoignent en une intersection sans signalisation.

L'objectif de ces tests était de valider qualitativement l'approche d'adaptation de vitesse en situation réelle, et d'ajuster les paramètres de la planification (seuil de TG, profils de vitesse de référence) à partir de données réelles.

Au total, 13 tests ont été menés avec différents comportements du conducteur de la C3, qui tentait de synchroniser son arrivée à l'intersection avec celle du véhicule automatisé.

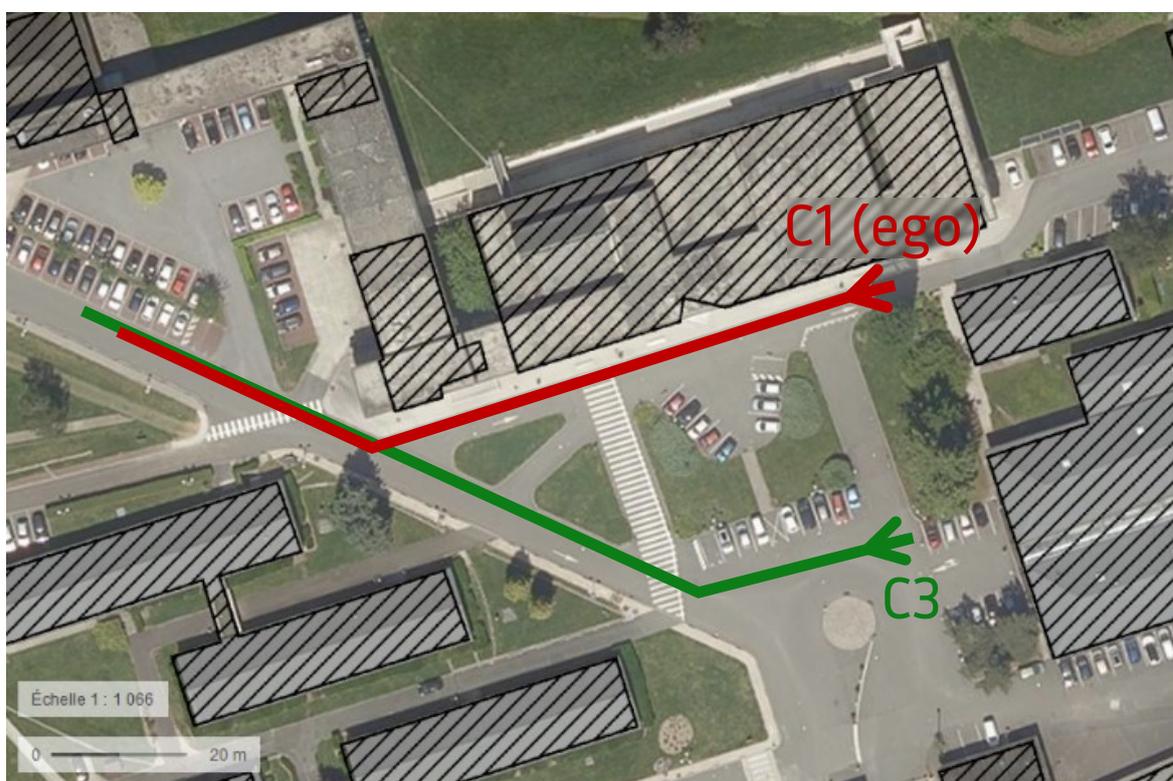


FIGURE 2.37 – Vue satellite de l'environnement de test sur le campus de Rocquencourt.

2.6.1 Implémentation des algorithmes sur les véhicules

L'implémentation de l'approche d'adaptation de vitesse sur les véhicules équipés de RITS a nécessité certaines modifications pour permettre une exécution en temps réel à partir des données fournies par les capteurs embarqués des véhicules.

Optimisations lors du calcul de l'estimation de risque

L'algorithme calculant le TG pour estimer le risque des trajectoires des deux véhicules doit être accéléré pour que son exécution s'effectue en temps réel.

Une première optimisation qui conserve le résultat exact est effectuée : elle consiste à indexer les trajectoires par leur abscisse curviligne, et à utiliser l'intersection géométrique des deux couloirs formés par les trajectoires des deux véhicules (qui correspond à la zone rouge de la Fig. 2.39). L'abscisse curviligne de la zone d'intersection permet de retrouver en temps constant les étapes utiles des deux trajectoires et en particulier celles durant lesquelles les véhicules sont susceptibles



FIGURE 2.38 – Intérieur du véhicule automatisé : un écran permet de suivre le déroulement de la planification en temps réel.

d'entrer en collision. Le TG est enfin calculé en identifiant la plus petite différence de temps entre deux étapes utiles qui entrent en collision.

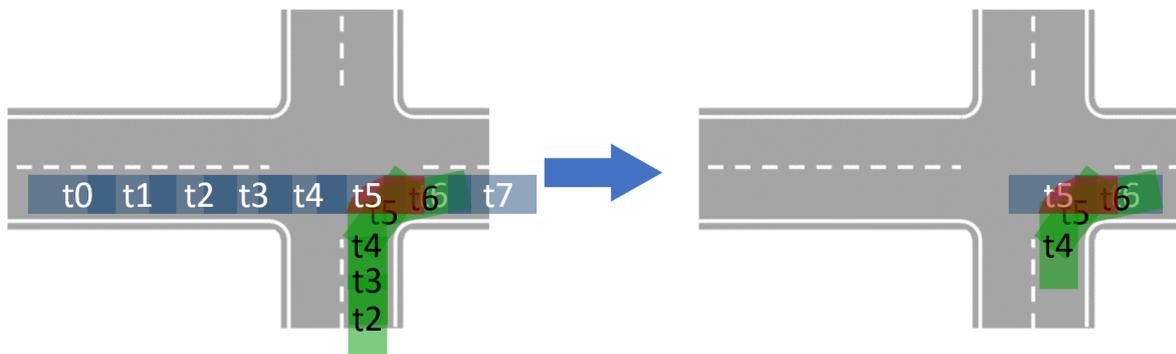


FIGURE 2.39 – Optimisation exacte du calcul du TG.

Une seconde optimisation implique une approximation dans le calcul de la valeur de TG. Elle consiste à remplacer les deux formes polygonales des véhicules par deux collections de disques approximaient ces véhicules (voir Fig. 2.40) lors du calcul d'intersection des zones occupées par les véhicules. La granularité de cette approximation est déterminée en fonction de la forme des véhicules et de la précision requise.

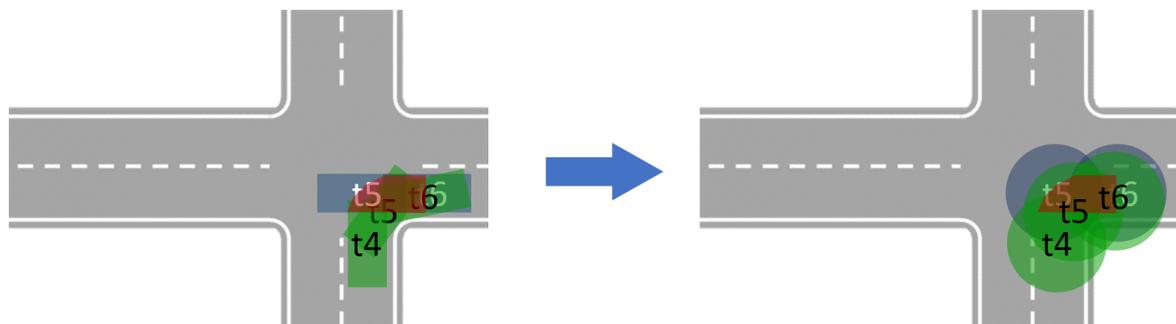


FIGURE 2.40 – Approximation par des collections de disques lors du calcul du TG.

Atténuation des entrées bruitées

Même après avoir appliqué un filtre passe bas sur les vecteurs d'état des deux véhicules afin d'en

diminuer la variance, l'incertitude des mesures angulaires reste forte. L'approche d'adaptation de vitesse est particulièrement sensible à la détection de la zone d'intersection entre les chemins des véhicules et, de ce fait, à la qualité de la mesure de l'angle de lacet des véhicules. Le couloir d'occupation de l'autre véhicule (la voiture C3) a ainsi été modifié pour tenir compte de l'incertitude croissante sur l'angle de lacet. Le couloir est élargi au cours du temps comme illustré sur la Fig. 2.41.

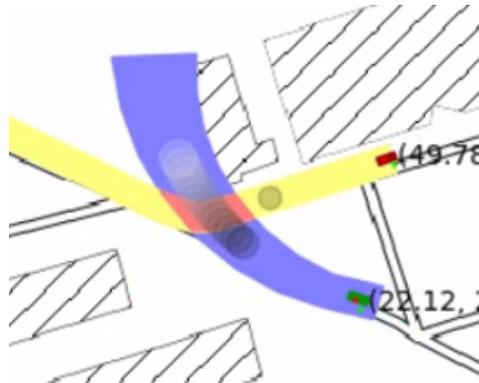


FIGURE 2.41 – Prise en compte de l'incertitude de la mesure de l'angle de lacet sur la forme du couloir de prédiction.

2.6.2 Expérimentations

Les expérimentations de l'algorithme d'adaptation dynamique de vitesse sur véhicules réels se sont globalement bien déroulées. Elles ont permis de valider en conditions réelles la pertinence de l'approche, consistant à ne pas recourir à un ensemble discret de manœuvres mais à ajuster en permanence une consigne de vitesse en fonction du contexte. Une évaluation qualitative nous a permis de constater que :

- Dans toutes les expérimentations, le profil de vitesse du véhicule a permis au VA de s'insérer ou de céder le passage confortablement sans passer par d'autres états (arrêt puis démarrage).
- Aucune intervention humaine n'a été requise sur le VA. Toutes les situations d'intersection se sont déroulées sans collision.
- Un délai supérieur à 0,5 s a été mesuré entre l'entrée d'une consigne de vitesse et son application à la sortie du module de contrôle.
- Au total, la C1 automatisée a cédé le passage 7 fois et s'est insérée devant l'autre véhicule 3 fois. Dans les 3 autres situations, les véhicules étaient trop éloignés pour avoir à interagir. Dans 2 des 7 situations où la C1 automatisée a cédé le passage à la C3, cette dernière s'est ensuite arrêtée à l'entrée de l'intersection, laissant la voie libre à la C1 qui a décidé à chaque fois de redémarrer pour s'engager sur l'intersection.
- La valeur du seuil de TG minimum acceptable, qui permet de déterminer le comportement de l'algorithme (plus ou moins « agressif ») et d'adapter les choix du véhicule en fonction des règles de priorité, a pu être testée en conditions réelles. Nous avons constaté qu'un TG égal à 1.5s utilisé pour le comportement qualifié de « prudent » durant les validations en simulation donnait l'impression d'une conduite agressive durant les expérimentations en conditions réelles. Ceci confirme les travaux de [BÄRGMAN et collab. \[2015\]](#) qui ont conclu que les valeurs inférieures à 2.0s sont réservées aux conducteurs pressés. De plus, le délai du contrôle du véhicule nécessitait de spécifier une marge de sécurité quant à la définition du seuil de TG. La valeur de ce seuil a par la suite été fixée à 3.0s lors des tests, ce qui procura aux conducteurs une sensation de sérénité jusqu'alors inexpérimentée.

Les Figures 2.43, 2.44 et 2.45 sont issues de vidéos effectuées lors de ces tests.

Les deux premiers tests affichés sont effectués avec un seuil de TG minimum de 3,0 s. Pour le premier test, la décision de céder le passage est prise, et la C1 ralentit jusqu'à 3,4 m/s. Le profil de vitesse lors de ce test est affiché Fig. 2.42. Le TG final entre les deux trajectoires après leur interaction est de $-2,25$ s, ce qui indique que le véhicule automatisé a effectivement adapté son profil de vitesse pour atteindre la valeur limite, mais le délai induit par le contrôle longitudinal ne permet pas de suivre le profil de vitesse de consigne.

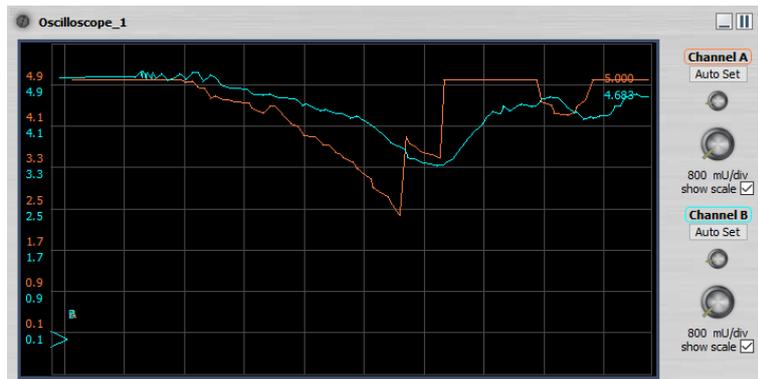


FIGURE 2.42 – Profil de la vitesse mesurée du véhicule (cyan) et consigne avec le profil de vitesse candidat (rouge). On observe 2 changements de décision discontinus sur ce dernier : du profil d'arrêt vers le profil constant, puis du profil constant vers le profil de croisière.

Durant le test montré Fig. 2.44, le comportement du conducteur de la C1 est légèrement différent, ce qui laisse le temps à la C1 pour s'insérer en premier sur l'intersection à une vitesse de 4,2 m/s. Le TG final est de 2,16 s, le véhicule automatisé a donc analysé que son insertion devant la C3 était compatible avec le seuil de risque de 3,0 s choisi, même si celui-ci est dépassé en pratique.

Sur la Fig. 2.45, on peut observer un test antérieur où la valeur du seuil était fixée à 2,0 s. La C1 a décidé de céder le passage sans trop baisser sa vitesse, comme le permet le seuil de TG minimum. Ici encore, le délai du contrôle ne permet pas de respecter tout à fait le profil de vitesse de consigne, ce qui induit un TG final de $-1,15$ s, perçu comme trop agressif du point de vue des passagers.

Perspectives d'améliorations

Bien que notre approche permette d'ajuster finement la vitesse du véhicule en s'appuyant sur plusieurs profils de vitesse de référence, la notion de confort n'a pas vraiment été envisagée lors de la définition de ces profils de référence. Pour améliorer le confort utilisateur, il serait tout à fait possible de procéder comme dans la méthode décrite en section 2.4. Les profils de référence seraient alors obtenus en utilisant les vitesses enregistrées lorsque des conducteurs humains circulent sur l'intersection considérée.

Une autre perspective d'amélioration est d'intégrer le délai de réponse complet du contrôle du véhicule. Ce délai a été partiellement compensé durant les tests en modifiant en amont la valeur de seuil de TG minimum, mais cette approximation n'est valide que dans une situation prédéterminée où la distance d'arrêt peut être estimée à l'avance. Il est nécessaire d'anticiper réellement ce délai en l'intégrant dans la modélisation MPC. A cette fin, un approximant de Padé pourrait être utilisé lors de la prédiction de mouvement.



FIGURE 2.43 – Décision de céder le passage en situation réelle. La C1 rouge (ego) ralentit à 3,4 m/s pour céder le passage à la C3 verte qui circule à 8,6 m/s. Le TG mesuré entre les deux trajectoires est de $-2,25$ s. L'écart entre 2 images successives sur la séquence est de 0,5 s.



FIGURE 2.44 – Décision de s’engager en situation réelle. La C1 rouge (ego) prend le virage à 4,2 m/s et s’insère 11,8 m devant la C3 verte qui circule à 9,4 m/s. Le TG mesuré entre les deux trajectoires est de 2,16 s. L’écart entre 2 images successives est de 0,5 s.



FIGURE 2.45 – Décision de céder le passage avec un seuil de TG agressif. La C1 rouge (ego) cède le passage à la C3 verte, en ne laissant qu'un faible délai entre les deux véhicules. Le TG mesuré entre les trajectoires est de $-1,15$ s. L'écart entre 2 images successives est de $0,5$ s.

2.7 Conclusion

Dans ce chapitre, nous avons introduit une stratégie de décomposition chemin-vitesse consistant à planifier la vitesse d'un véhicule automatisé sur un chemin établi d'avance. Le profil de vitesse est adapté dynamiquement en fonction des trajectoires prédites des véhicules qui vont croiser le chemin du VA. Cette approche a été appliquée dans le cadre de situations de conduite urbaine, sur des intersections non supervisées ainsi que dans un environnement semi-contraint.

L'algorithme adapte la vitesse du VA en combinant plusieurs hypothèses de profils de vitesse générés et adaptés en permanence selon la situation. Ces profils de référence sont inspirés de la conduite humaine afin que le VA suive un profil de vitesse qui soit qualitativement acceptable pour des passagers humains ainsi que pour les autres conducteurs. Par exemple, lorsque le VA anticipe un conflit avec un autre véhicule dont il croisera la route il est capable, comme le ferait un conducteur humain, de planifier un profil de vitesse adapté sans pour autant commencer à ralentir immédiatement.

Selon la littérature dédiée au facteur humain dans les situations de conduite, le PET s'avère être un critère adéquat pour quantifier le risque a posteriori lors d'un croisement à une intersection. Nous avons décidé d'intégrer cette métrique de risque dans le processus de décision dans des scénarios d'intersections en l'utilisant pour retirer les hypothèses de profils de vitesse trop risquées. Cette métrique n'est pas adaptée pour toutes les situations de croisement qui peuvent avoir lieu sur un environnement semi-contraint, cependant nous avons montré que le TG lui est équivalent dans un scénario de croisement à 90°, et permet de mieux généraliser sur des cas de figure plus divers.

L'approche a fait l'objet de scénarios de validation avec un outil de simulation spécifiquement développé pour les travaux présentés dans ce chapitre. Les scénarios de test reproduisent des situations sur des intersections, des carrefours giratoires, et sur un environnement semi-contraint avec plusieurs milliers de variantes pour chaque scénario. L'objectif de ces tests est d'effectuer ce qui s'approche le plus d'une validation systématique des performances de l'algorithme en matière de sécurité, avec des situations qui requièrent parfois une décision immédiate du VA pour qu'une collision soit évitée. De plus, des expérimentations sur véhicules ont permis de valider qualitativement la pertinence de l'approche en situation réelle.

Les résultats en simulation montrent globalement une bonne robustesse de l'algorithme lorsqu'il est confronté à des situations critiques variées. L'algorithme a permis d'éviter d'entrer en collision dans toutes les situations de croisement, que ce soit sur une intersection ou dans un environnement semi-contraint. Le TG effectif mesuré a posteriori permet d'observer que le seuil de risque qui a été spécifié est respecté en pratique, à condition que le MPC permette une bonne modélisation du module de contrôle. Des situations dangereuses ont néanmoins été rencontrées lorsque d'autres véhicules modifient leur trajectoire en tournant ou en accélérant vers le VA. Ce comportement peut être rencontré sur un environnement semi-contraint, par exemple dans le scénario « yield » étudié, lorsque les autres véhicules s'insèrent rapidement à proximité du VA en ayant la priorité. Le modèle de prédiction de mouvement utilisé par notre approche ne tient pas compte des priorités et ne permet donc pas d'anticiper un tel comportement. Il est à noter que les conditions initiales des tests de validation étaient spécifiquement conçues pour générer des situations de conflits, à la limite des cas de collisions physiquement inévitables.

Toutefois, la bonne adéquation entre le TG effectif a posteriori par rapport au TG choisi comme seuil de l'algorithme permet d'envisager de moduler le seuil de TG minimum en fonction du contexte de conduite. Par exemple, dans une situation urbaine bien identifiée, le TG minimum pourrait être fixé à 1.5s lorsque le véhicule a la priorité et à 3.0s lorsqu'il doit céder le passage. De manière générale, ce seuil devrait être augmenté lorsque la situation est incertaine ou si le comportement des autres véhicules semble anormal. De même, il peut être modulé en fonction des limitations de vitesse : lors d'une insertion sur une route prioritaire périurbaine, il est préférable d'utiliser un seuil de TG minimum plus élevé que dans le cas d'un cédez-le-passage en ville. De plus, lorsque la topologie de route le permet, il est nécessaire d'utiliser une prédiction de mouvement capable d'intégrer les intentions des autres véhicules.

Chapitre 3

Planification de chemin dans un environnement fortement dynamique

« Il vaut mieux suivre le bon chemin en boitant que le mauvais d'un pas ferme. »

Saint-Augustin

Sommaire

| | |
|---|-----------|
| 3.1 Introduction | 79 |
| 3.2 État de l'art | 79 |
| 3.2.1 Introduction | 79 |
| 3.2.2 Planification de chemin en présence d'obstacles fixes | 80 |
| 3.2.2.1 Graphes de visibilité | 80 |
| 3.2.2.2 Champs de potentiel artificiel | 80 |
| 3.2.2.3 Arbre à exploration rapide | 82 |
| 3.2.3 Notions importantes | 83 |
| 3.2.3.1 Espace de configuration | 83 |
| 3.2.3.2 Chemin de Dubins | 83 |
| 3.2.3.3 Ensembles atteignables | 84 |
| 3.2.4 Approches de planification de chemin dynamique | 84 |
| 3.2.4.1 L'espace temps-état | 84 |
| 3.2.4.2 Treillis d'états | 86 |
| 3.2.4.3 Cône de collision | 86 |
| 3.2.4.4 Obstacles vitesse | 87 |
| 3.2.4.5 Ensembles stochastiques atteignables | 88 |
| 3.2.4.6 États de collision inévitable | 89 |
| 3.2.4.7 Fenêtre dynamique | 90 |
| 3.2.4.8 Champs de potentiel artificiel | 91 |
| 3.2.5 Conclusion | 91 |
| 3.3 Reachable Interaction Sets : présentation de l'approche | 93 |
| 3.3.1 Cas d'un robot holonome | 93 |
| 3.3.2 Cas d'un robot soumis à des contraintes dynamiques non-holonomes | 94 |
| 3.3.3 Cas où les mouvements des obstacles mobiles ne sont pas déterministes | 95 |
| 3.4 Application des RIS à la planification de mouvement | 98 |
| 3.4.1 Stratégies locales de planification dynamique | 98 |
| 3.4.1.1 Planification réactive avec champ de potentiel | 99 |
| 3.4.1.2 Planification locale avec courbe de Bézier | 100 |

| | | |
|------------|---|------------|
| 3.4.1.3 | Planification locale hybride | 102 |
| 3.4.2 | Stratégie globale de planification dynamique | 102 |
| 3.4.2.1 | Adaptation d'une planification globale en visibilité au modèle de Dubins | 102 |
| 3.4.2.2 | Nouvel espace de configuration | 108 |
| 3.4.2.3 | Application de la planification globale aux RIS | 108 |
| 3.5 | Validation des approches de planification | 111 |
| 3.5.1 | Validation en présence d'obstacles au mouvements déterministes | 111 |
| 3.5.1.1 | Les scénarios de test | 111 |
| 3.5.1.2 | Résultats | 112 |
| 3.5.2 | Validation en présence d'obstacles au mouvements stochastiques | 118 |
| 3.5.2.1 | Les scénarios de test | 118 |
| 3.5.2.2 | Résultats | 122 |
| 3.6 | Conclusion | 128 |

3.1 Introduction

Considérons maintenant le problème de la planification de chemin d'un système robotique dans un environnement contenant de nombreux obstacles fortement dynamiques. La décomposition chemin-vitesse introduite précédemment conduit de manière classique à adapter le profil de vitesse du véhicule sur un chemin donné. Nous proposons ici la démarche inverse, qui consiste à rechercher un chemin admissible parmi les obstacles mobiles à partir d'une hypothèse formulée a priori sur la vitesse du véhicule.

A ce titre, nous introduisons dans ce chapitre la notion de « Reachable Interaction Sets » (RIS) comme nouveau cadre de travail permettant d'effectuer la planification de chemin parmi des obstacles très dynamiques.

La notion d'obstacles « très » ou « fortement » dynamiques dont il est question ici repose sur le constat que la plupart des approches de planification de trajectoire en présence d'obstacles mobiles repose sur l'hypothèse que le robot est capable d'évoluer à une vitesse égale ou plus grande que celle des obstacles, comme nous le verrons dans la section 3.2 (c'est explicitement le cas dans FUJIMURA et SAMET [1990], CHEON et KIM [2019], WILKIE et collab. [2009] et MARTINEZ-GOMEZ et FRAICHARD [2009]). Nous proposons donc une approche heuristique pour planifier une trajectoire qui reste efficace en présence d'obstacles plus rapides que le robot.

Cette approche est détaillée dans la section 3.3. Elle consiste à émettre une hypothèse a priori sur la vitesse suivie par le robot jusqu'à l'horizon temporel, ce qui permet de supprimer l'aspect temporel du problème de planification, le ramenant à une recherche de chemin duale. Pour cela, les RIS établissent une nouvelle représentation de l'espace dans lequel le robot se déplace (ici, un plan). Cette représentation contient les RIS, qui sont les zones que le robot doit contourner afin d'éviter d'entrer en collision avec les obstacles.

Différents algorithmes de planification de chemin présentés dans la section 3.4 sont utilisés avec les RIS, afin de constituer des approches de planification de trajectoire. Ces approches vont être comparées à celles de l'état de l'art en planification de mouvement dans un environnement dynamique via une phase de validation quantitative sur différents scénarios dans la section 3.5.

Enfin, ces travaux généralisent la définition des RIS pour un robot non [holonome](#) en adaptant l'approche à la modélisation d'un véhicule de Dubins. Le cas d'obstacles mobiles qui suivent une loi de mouvement stochastique est également étudié, afin d'intégrer la notion d'incertitude à la planification de trajectoire.

3.2 État de l'art

3.2.1 Introduction

La recherche d'un chemin admissible permettant à un véhicule d'accéder à une destination donnée en évitant les obstacles est un problème majeur en robotique. Ce problème demeure ouvert, puisque différentes approches de planification peuvent être appropriées, selon les contraintes exactes du problème. Ces contraintes peuvent être en matière de temps de calcul, en fonction des contraintes dynamiques du robot ou de sa forme, du type d'environnement considéré ou encore selon la qualité du chemin qui doit être obtenu. L'état de l'art présenté ici comparera les principales approches de planification de trajectoire dans un environnement dynamique.

Il convient de noter que la notion d'environnement dynamique utilisée ici fait référence à la présence d'obstacles qui se déplacent continuellement. En effet, ce terme est parfois utilisé pour un environnement quasi-statique dans lequel certains obstacles fixes changent d'état (ex : une porte qui s'ouvre). Certaines approches de recherche de chemin sont élaborées spécifiquement pour ces types d'environnement, qui demandent de replanifier sporadiquement le chemin, et non très régulièrement comme le requiert la présence d'obstacles mobiles. Ce type d'approches ne sera pas traité ici.

Les auteurs de **CANNY et REIF [1987]** ont démontré que la recherche de chemin sur un espace en 2 dimensions parmi des obstacles mobiles est NP-difficile, y compris lorsque le mouvement des obstacles est rectiligne et uniforme. On peut donc sérieusement douter que le problème de planification de chemin en environnement dynamique puisse être résolu par un algorithme en temps polynomial. C'est également la raison pour laquelle toutes les approches existantes sont soit fondées sur une heuristique soit sur une exploration exhaustive de l'espace après discrétisation du problème. Les approches par échantillonnage peuvent être interprétées comme étant une combinaison des deux, permettant d'explorer l'espace en suivant une loi aléatoire qui est spécifiée selon une hypothèse formulée a priori.

Cet état de l'art présentera tout d'abord en section 3.2.2 les stratégies générales de planification de chemin dans un environnement statique, puis introduira dans la section 3.2.3 certaines notions importantes liées à la planification. La section 3.2.4 présentera les principales approches de planification de trajectoire en présence d'obstacles mobiles. Enfin, ces approches seront comparées dans un tableau récapitulatif dans la section 3.2.5.

3.2.2 Planification de chemin en présence d'obstacles fixes

La planification de chemin d'un robot se déplaçant d'un point à un autre en présence d'obstacles fixes a fait l'objet de nombreux travaux (cf. **LATOMBE [1991]**, **LAVALLE [2006]**, **MINGUEZ et collab. [2008]**, **HOY et collab. [2015]**, **PADEN et collab. [2016]**, **MAC et collab. [2016]** pour un état de l'art sur le sujet). Notre objectif ici n'est pas d'effectuer une revue de la littérature mais plutôt de présenter trois grandes familles d'algorithmes de planification dont des extensions seront proposées dans le cas des environnements dynamiques : les approches combinatoires sur des graphes, les approches réactives sans planification, et les planifications de chemin par échantillonnage.

3.2.2.1 Graphes de visibilité

La planification par visibilité est l'une des approches les plus courantes permettant de calculer le plus court chemin entre deux points dans une scène comportant des obstacles polygonaux (**NILSSON [1969]**, **LAVALLE [2006]**). Elle s'appuie sur la construction d'un graphe de visibilité (*Visibility Roadmap*). C'est un graphe valué où les nœuds sont les sommets des enveloppes convexes des obstacles et les arêtes sont les segments qui relient chaque paire de nœuds mutuellement visibles, c'est à dire dont le segment qui les joint n'intersecte pas l'intérieur d'un obstacle (cf Fig. 3.1). Les valeurs associées aux arêtes sont les longueurs des segments correspondants. Le calcul du plus court chemin nécessite l'ajout des nœuds de départ et d'arrivée dans le graphe et s'effectue généralement en utilisant l'algorithme de Dijkstra. Le graphe de visibilité des obstacles statiques peut être pré-calculé et utilisé à plusieurs reprises en considérant des nœuds de départ et d'arrivée différents.

3.2.2.2 Champs de potentiel artificiel

L'approche des champs de potentiel artificiel (*Artificial Potential Fields*, ou *APF*) consiste à calculer le mouvement d'un robot considéré comme une particule soumise à des forces d'attraction et de répulsion créées par des champs de potentiels positif et négatif appropriés. Ces champs de potentiel permettent d'attirer le robot vers sa destination tout en l'éloignant des obstacles (cf Fig. 3.2). La vitesse de consigne du robot est assimilée à une force qui dérive de ces potentiels. L'intensité de ces forces est généralement proportionnelle à la distance ou au carré de la distance entre le robot et l'origine du potentiel (cf. **LATOMBE [1991]**). Les *APF* ont été formalisés pour la première fois dans **KHATIB [1985]** et sont à l'origine d'une littérature dédiée fournie dans le domaine de la robotique. Cette approche est de type réactive, ce qui signifie que sa simplicité conceptuelle et sa légèreté lors de l'implémentation ont pour corollaire une absence de planification sur le long terme et une forte sensibilité aux extremums locaux.

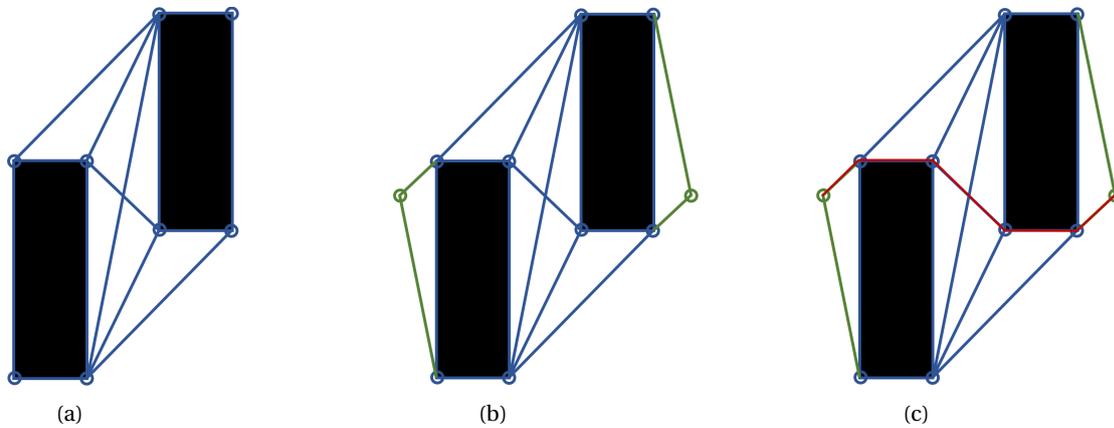


FIGURE 3.1 – Construction d'un graphe de visibilité. (a) les arêtes du graphe sont les segments visibles reliant 2 sommets d'obstacles (b) lors de la recherche de chemin, les nœuds de départ et de destination sont ajoutés ainsi que les arêtes correspondantes (c) une recherche de chemin dans le graphe trouve le chemin optimal

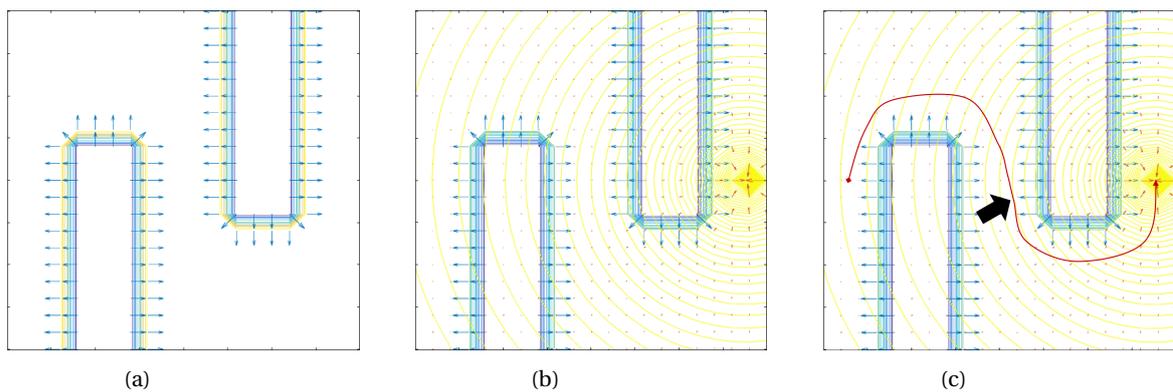


FIGURE 3.2 – (a) Potentiel répulsif (bleu) émanant des obstacles (b) Potentiel attractif (jaune) vers la destination (c) Tracé théorique du mouvement du robot (rouge) suivant le gradient du potentiel. La flèche indique une zone de minimum local où le robot risque de rester bloqué

Il est amusant de relever que l'idée d'un robot suivant un gradient de potentiel pour se déplacer est mentionnée comme telle près de 40 ans plus tôt dans la nouvelle de science-fiction « Runaround » d'Isaac Asimov (ASIMOV [1942]). Dans cette nouvelle, des scientifiques enquêtent sur le robot Speedy qui semble être devenu fou. Celui-ci s'avère être coincé dans un équilibre stable entre une force attractive qui le dirige vers l'objet de sa mission et une force répulsive qui l'empêche de s'approcher d'une zone trop dangereuse à sa propre survie. En effet, lorsqu'un obstacle est situé trop près de la destination, le comportement d'un robot suivant un APF peut s'avérer tout à fait erratique.

3.2.2.3 Arbre à exploration rapide

L'algorithme d'arbre à exploration rapide (*Rapidly Exploring Tree*, ou *RRT*) a été introduit par LAVALLE [1998]. L'approche fait partie de la famille des méthodes de planification de chemin par échantillonnage. Elle consiste à construire progressivement un arbre d'exploration de l'espace en tirant aléatoirement une position de l'espace et en l'utilisant comme direction pour agrandir l'arbre. Une branche est alors ajoutée à partir du nœud de l'arbre le plus proche de cette position aléatoire et se déploie vers celle-ci d'une distance fixée d'avance. En cas de collision avec un obstacle, la branche n'est pas ajoutée. Ce processus est réitéré jusqu'à ce qu'une des branches soit dans le voisinage de la destination : le chemin est obtenu en remontant les nœuds de l'arbre jusqu'à la racine placée au point de départ (cf. Fig. 3.3). De nombreuses améliorations incrémentales ont été proposées (LAVALLE et KUFFNER [2001], KARAMAN et FRAZZOLI [2011]), notamment l'algorithme « anytime » *RRT**, capable d'optimiser peu à peu le premier chemin trouvé jusqu'à ce que le temps de calcul qui lui est imparti soit écoulé.

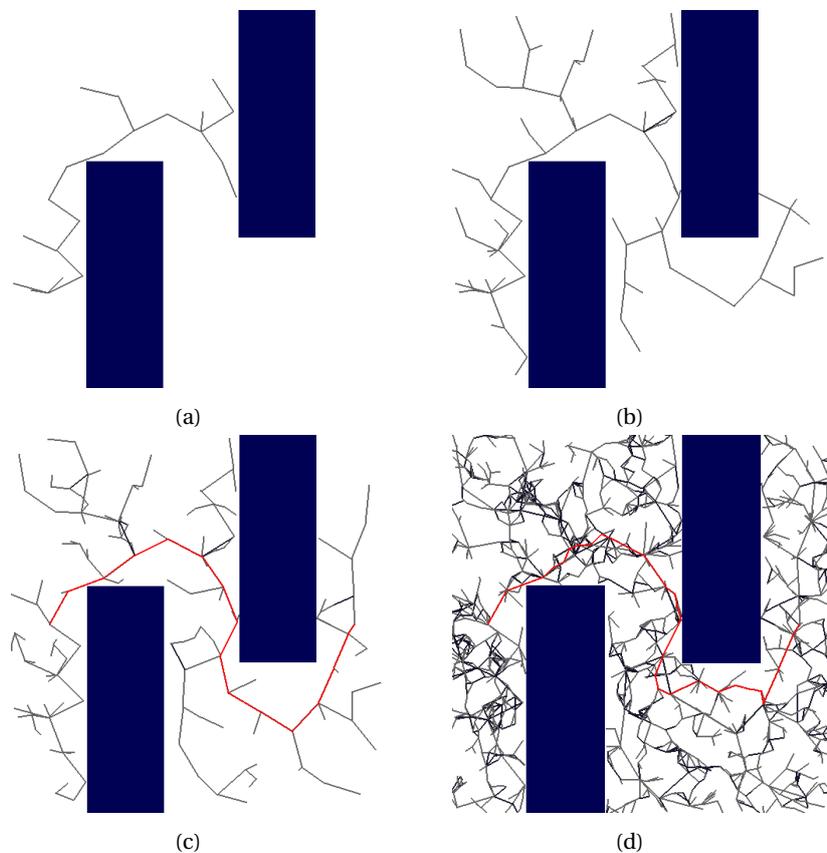


FIGURE 3.3 – Développement d'un arbre à exploration rapide. (c) Un premier chemin est trouvé lorsqu'un des nœuds se trouve suffisamment proche de la destination. (d) L'algorithme *RRT** continue de densifier l'arbre afin d'améliorer peu à peu le chemin, jusqu'à interruption

3.2.3 Notions importantes

Les trois notions introduites dans cette section ne sont pas à proprement parler des stratégies de planification. Elles sont néanmoins étroitement liées aux calculs de recherche de chemin, et feront notamment l'objet de développements dans le reste du chapitre.

3.2.3.1 Espace de configuration

L'espace de configuration d'un robot se déplaçant dans le plan 2D est l'ensemble des positions de l'environnement qui sont accessibles pour le point représentant la position du robot. Ce nouvel espace permet d'effectuer une planification de chemin duale pour laquelle on peut considérer que le robot est assimilé à un point. Les obstacles dans cet espace sont alors modifiés par rapport aux formes des obstacles réels. Le chemin trouvé (ou l'absence de chemin possible) sera alors valide dans l'espace initial pour le robot polygonal.

La première formulation de la notion d'espace de configuration se trouve dans [LOZANO-PEREZ et WESLEY \[1979\]](#) pour un robot polygonal dont l'orientation absolue reste la même lors de ses déplacements. Cet espace est calculé par extension des obstacles via une somme de Minkowski avec la forme du robot. Si l'orientation du robot n'est pas fixe lors de ses déplacements, ce qui est souvent le cas, les auteurs suggèrent d'utiliser la boîte englobante des polygones du robot pivotés pour chaque orientation possible à la place de la forme du robot dans la somme de Minkowski. En pratique, il est courant de calculer l'espace de configuration d'un robot en rotation par somme de Minkowski en utilisant le disque correspondant au cercle minimum contenant le polygone du robot. Ces deux espaces de configuration seront illustrés et comparés avec une nouvelle approche dans la section [3.4.2.2](#).

L'analyse de l'espace de configuration permet aux auteurs de [FUJIMURA et SAMET \[1990\]](#) de définir un « front de collision » pour chaque obstacle dans le voisinage du robot. En identifiant chaque cas d'interaction possible entre le robot et chacun de ces fronts de collision, il est possible de déduire une planification de trajectoire évitant les collisions pour un robot se déplaçant à une vitesse supérieure ou égale aux obstacles mobiles. Les cas de figures où des obstacles mobiles se superposent sont étudiés et se réduisent à deux situations élémentaires : la séparation et la fusion de deux fronts de collision. Selon les auteurs, cette approche pourrait être généralisée à un environnement non déterministe à condition d'avoir des hypothèses suffisantes sur les mouvements des obstacles mobiles.

3.2.3.2 Chemin de Dubins

Le terme de chemin de Dubins ([DUBINS \[1957\]](#), [LAVALLE \[2006\]](#)) se réfère au plus court chemin permettant de connecter deux points du plan par une courbe dont la courbure est limitée par une valeur maximale, les points de départ et d'arrivée étant chacun associé à une orientation requise, et le véhicule parcourant le chemin ne pouvant pas effectuer de marche arrière (cf. [Fig. 3.4](#)). Un chemin de Dubins est la succession d'un arc de cercle dont le rayon correspond à l'angle de braquage maximal, d'un segment de droite et d'un autre arc de cercle de même rayon que le premier. Le segment de droite est tangent aux deux arcs de cercles afin d'assurer la continuité C^1 de la courbe. Lorsque le véhicule est capable d'évoluer en marche arrière, le chemin optimal est alors une courbe dite de Reeds–Shepp ([REEDS et SHEPP \[1990\]](#)). L'exemple illustré [Fig. 3.5](#) montre que le chemin de Reeds–Shepp est plus court que le chemin de Dubins équivalent ([Fig. 3.4\(b\)](#)).

Un chemin de Dubins peut être parcouru à vitesse constante par un véhicule non-holonome hypothétique dont l'angle de braquage est borné mais peut être changé instantanément. En pratique, un véhicule non-holonome doit faire varier continûment son angle de braquage et nécessite ainsi de s'arrêter pour changer d'angle de braquage lorsque la courbure du chemin change. Afin de pouvoir être suivi par un véhicule non-holonome dont l'angle de braquage reste continu, le chemin doit vérifier une continuité C^2 et la dérivée de sa courbure doit être également bornée. Cette propriété peut être réalisée par l'ajout de clothoïdes afin d'effectuer une transition continue entre la courbure nulle du segment de droite et la courbure constante des arcs de cercle ([SCHEUER](#)

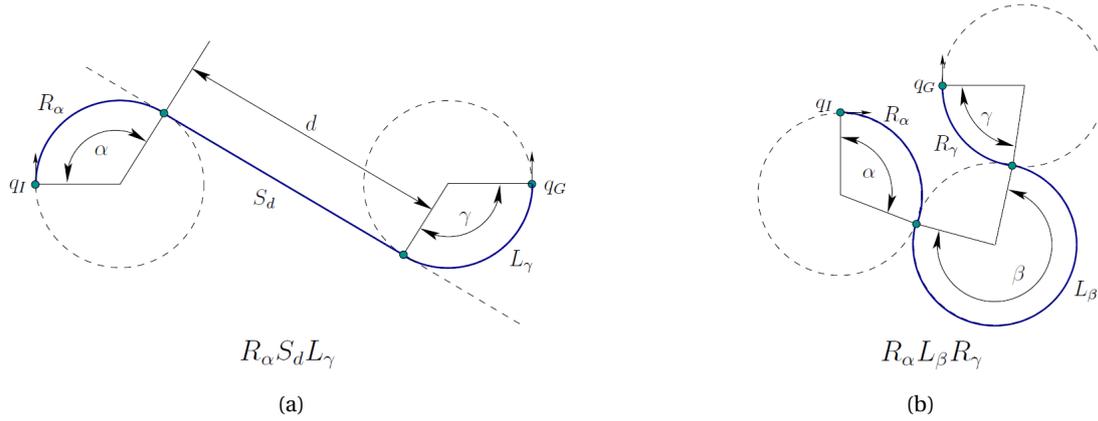


FIGURE 3.4 – Exemples de chemins de Dubins

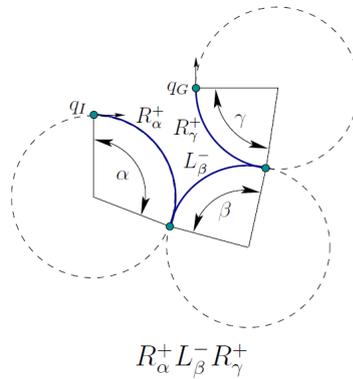


FIGURE 3.5 – Exemple de chemin de Reeds-Shepp

et FRAICHARD [1997]). De façon analogue aux courbes de Reeds-Shepp, il est également possible de générer une courbe optimale conservant la continuité de la courbure lorsque le véhicule est capable d'effectuer des marches arrières (FRAICHARD et SCHEUER [2004]).

3.2.3.3 Ensembles atteignables

Les ensembles atteignables (*Reachable Sets, RS*) d'un objet mobile soumis à des contraintes dynamiques caractérisent la région de l'espace que celui-ci peut atteindre en un intervalle de temps donné (cf L'AVALLE [2006]). Le calcul exact des ensembles *RS* d'un robot peut se déduire analytiquement via la mécanique hamiltonienne, en résolvant une équation de Hamilton-Jacobi aux dérivées partielles. Dans MITCHELL et collab. [2005], MARGELLOS et LYGEROS [2010] et GILLULA et collab. [2011], cette méthode est utilisée afin de constituer une garantie de sûreté à partir du scénario pire-cas où un obstacle mobile est supposé poursuivre le robot afin de provoquer une collision. L'exemple traité est souvent tiré du domaine aérien, afin de garantir l'évitement de collision pour un avion. Cependant, ce scénario pire-cas s'avère vite impossible à traiter pour des cas de figure non triviaux lorsque la densité d'obstacles augmente. Une approche similaire est utilisée dans le même contexte dans ZHOU et BARAS [2015], mais le calcul des *RS* est optimisé par une double optimisation convexe qui permet notamment une parallélisation efficace, ce qui rend l'approche plus abordable en présence de plusieurs obstacles mobiles.

3.2.4 Approches de planification de chemin dynamique

3.2.4.1 L'espace temps-état

Le concept de représentation dans l'espace temps-état (*State-Time Space*) a été introduit dans ERDMANN et LOZANO-PÉREZ [1987]. Il s'agit d'envisager la planification de trajectoire en présence

d'obstacles dynamiques se déplaçant dans un plan comme une planification de chemin dans un espace tridimensionnel où des obstacles statiques 3D représentent les trajectoires prédites des obstacles mobiles 2D. A l'instar des espaces de configuration qui permettent de représenter la géométrie du véhicule ego par un problème de planification dual où un véhicule ponctuel doit évoluer parmi des obstacles modifiés, cette représentation est un espace-temps de configuration où la planification de trajectoire est remplacée par une planification de chemin duale.

Dans FRAICHARD [1998], l'espace temps-état est intégré au sein d'une planification de trajectoire. Un chemin statique qui contourne les obstacles fixes est supposé connu. La planification de trajectoire revient à une planification de la vitesse du véhicule ego afin d'éviter les collisions avec les obstacles mobiles. L'espace-temps en deux dimensions (l'abscisse curviligne du chemin et le temps) est discrétisé régulièrement en nœuds ayant chaque trois descendants correspondants à trois valeurs d'accélération (minimale, nulle, et maximale), cf. Fig. 3.6. Sur cet arbre, le chemin optimal en temps de parcours est trouvé via un algorithme A* (LAVALLE [2006]) exécuté avec une fonction de coût simple.

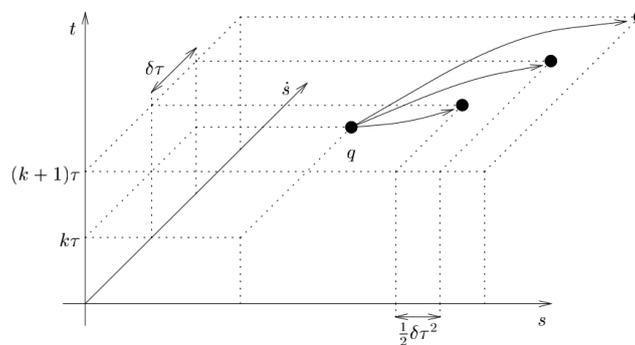


FIGURE 3.6 – Les nœuds dans l'espace temps-état (Fig. issue de FRAICHARD [1998])

Cette représentation temps-état peut être étendue à une recherche de trajectoire dans le plan, ce qui revient à une planification sur trois dimensions (HSU et collab. [2002], VAN DEN BERG et OVERMARS [2008], CHEON et KIM [2019]). La planification de trajectoire de Hsu et collab. [2002] est fondée sur un arbre RRT qui croît via un double échantillonnage : celui du nœud de départ et celui de l'entrée du contrôle (accélération et angle de lacet) afin de créer une nouvelle arête à condition que la portion de chemin soit navigable. L'expansivité d'une région de l'espace est la propriété qui quantifie à quel point une région est connectée au reste de l'espace accessible. Une zone de l'espace navigable est expansive si tout sous-ensemble situé à l'intérieur permet de transiter vers d'autres zones de l'espace navigable. A l'inverse, une zone qui ne serait ouverte que sur un couloir étroit sera définie comme peu expansive. Les auteurs étendent cette définition à l'espace temps-état afin d'améliorer l'efficacité de la stratégie d'échantillonnage. Un facteur de pondération est associé aux zones de l'espace pour favoriser l'ajout de nœuds dans les régions où leur densité est faible.

Dans VAN DEN BERG et OVERMARS [2008], les obstacles sont assimilés à des disques au mouvement inconnu mais dont la vitesse maximale est supposée connue. Les obstacles sont donc représentés comme des disques qui se dilatent au cours du temps à la vitesse maximale de leur déplacement (cf. Fig. 3.7). En termes de temps-état, l'occupation dans l'espace d'une trajectoire d'obstacle mobile est comprise dans un cône. Un chemin minimisant le temps de parcours entre ces cônes est calculé en utilisant des suites de segments de droite pour approximer des spirales contournant les cônes. Un graphe similaire au approches par visibilité est formé et une recherche de chemin via les algorithmes A* ou Dijkstra retourne le chemin optimal.

Une autre approche de planification pour environnement dynamique basée sur le temps-état est présentée dans CHEON et KIM [2019]. L'espace est exploré de manière sûre en utilisant la notion d'ICS (*Inevitable Collision States*). Trois chemins sont construits : le premier à partir du point de départ, un autre à partir de la destination, et un troisième qui les relie. L'approche est implé-

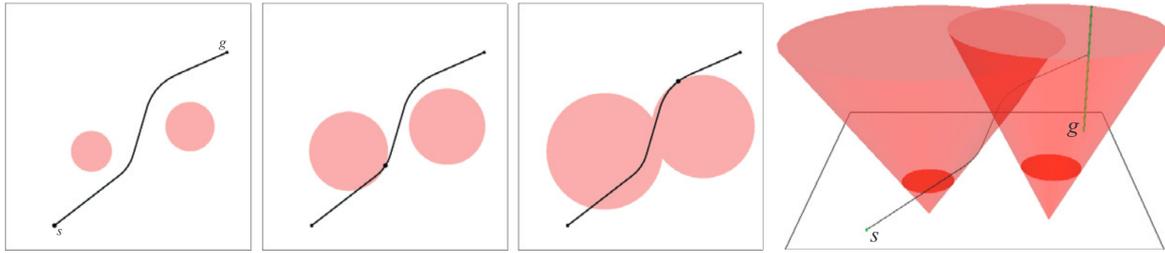


FIGURE 3.7 – Trajectoire évitant les Reachable Sets coniques des obstacles (Fig. issue de VAN DEN BERG et OVERMARS [2008])

mentée pour deux modélisations dynamiques du véhicule ego : un robot holonome et un robot différentiel. Il est à noter que les obstacles mobiles évoluent systématiquement à une vitesse inférieure ou égale à celle du véhicule ego, et leur densité dans la scène est relativement faible pour les scénarios considérés.

3.2.4.2 Treillis d'états

Un treillis d'états, ou *State Lattices* (PIVTOIRAICO et KELLY [2005]) est une discrétisation de l'espace de configuration en un ensemble d'états connectés sous la forme d'un graphe orienté. Les connections entre deux états représentent ainsi les transitions via un chemin accessible sur l'espace navigable. La représentation en treillis permet donc d'effectuer une planification de trajectoire sur un graphe, en ajoutant la garantie que les arrêtes puissent être effectivement suivies par le modèle non-holonome du véhicule. En raison de la complexité du graphe à construire, des améliorations ont été introduites pour optimiser la résolution du treillis. Les auteurs de LIKHACHEV et FERGUSON [2009] élaborent un treillis multi-résolution qui permet de construire une approximation du graphe tout en contrôlant l'erreur, et dont les requêtes de chemin sont *anytime* ce qui garantit une exécution en temps réel.

La planification de trajectoire basée sur des treillis d'états a été adaptée pour l'évitement d'obstacles dynamiques (ZIEGLER et STILLER [2009], WERLING et collab. [2012]). Du fait de l'ajout de la dimension temporelle, la complexité du treillis augmente fortement. Pour autant, la résolution du treillis doit rester suffisante afin de pouvoir décrire une stratégie d'évitement des collisions exécutable par le véhicule. Dans ces travaux, la vitesse d'exécution est nettement améliorée par un échantillonnage adapté du treillis, conditionné par la topologie des voies sur lesquelles le véhicule doit circuler. Les expérimentations montrent des temps de calcul compatibles avec le temps réel, mais reposent sur un environnement avec des contraintes fortes permettant de limiter la taille de l'espace des configurations à explorer.

3.2.4.3 Cône de collision

Le concept du cône de collision est introduit dans CHAKRAVARTHY et GHOSE [1998] pour deux objets de forme arbitraire se déplaçant à vitesse constante. Cette approche géométrique permet d'identifier l'ensemble des directions que le véhicule ego doit éviter pour ne pas entrer en collision avec l'obstacle. Pour cela, la norme de la vitesse du véhicule ego est fixée d'avance et supposée constante.

Cette approche est étendue afin de permettre d'éviter les collisions avec plusieurs obstacles qui se déplacent ou se déforment au cours du temps (CHAUDHARY et GHOSE [2017], SUNKARA et collab. [2019]). Bien que l'approche soit théoriquement valide, aucune validation pratique de ces approches dans un scénario comportant des obstacles mobiles n'a été proposée. Le cône de collision est cependant utilisé dans un scénario où un petit robot mobile doit éviter un obstacle de grande taille qui se déforme au cours du temps. Ce cas de figure peut se retrouver dans le contexte maritime par exemple, lorsqu'un bateau doit éviter le trait de côte avec un coefficient de marées important.

3.2.4.4 Obstacles vitesse

Le concept d'obstacles vitesse (*velocity obstacles*) est introduit dans FIORINI et SHILLER [1993] pour un environnement dynamique dans lequel les obstacles mobiles sont des disques évoluant à vitesse constante. C'est une approche géométrique qui représente dans l'espace des vitesses du robot l'ensemble des vecteurs vitesses qui mèneront à une collision, en supposant que les vitesses des objets demeurent constantes (voir Fig. 3.8). Sous ces hypothèses, une stratégie d'évitement d'obstacles peut être adoptée en appliquant au robot tout vecteur vitesse se situant en dehors des *velocity obstacles*. La représentation *velocity obstacles* peut être appliquée à un évitement d'obstacles mobiles en forme de disque dans un espace en deux dimensions ou à des obstacles sphériques dans un espace en trois dimensions. De nombreuses variantes de cette approche par obstacles de vitesse ont ensuite été proposées dans la littérature.

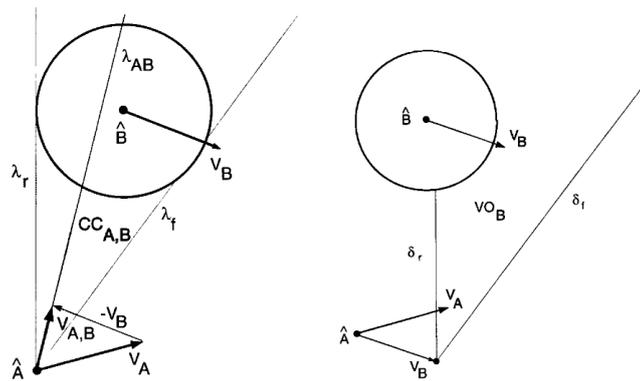


FIGURE 3.8 – Construction d'un obstacle vitesse par translation du cône de collision (à gauche) par le vecteur de la vitesse instantanée du robot (à droite). (Fig. issue de FIORINI et SHILLER [1998])

Cette approche est étendue par les mêmes auteurs dans FIORINI et SHILLER [1998], où une stratégie de planification est combinée avec la représentation des *velocity obstacles* afin d'atteindre une destination tout en respectant plusieurs critères (minimiser le temps de parcours, maintenir un TTC minimum, etc). Un arbre de recherche est créé afin d'explorer plusieurs hypothèses successives. Plusieurs heuristiques simples sont proposées afin de créer les branches de l'arbre : maintenir une vitesse maximale et orienter le robot dans la direction la plus proche de la destination, ou maintenir le robot dans la direction de la destination et ajuster la vitesse afin de conserver un TTC suffisant. Ces heuristiques restent néanmoins très « réactives », et ne constituent pas une planification de trajectoire globale.

Une planification de chemin réciproque est proposée dans VAN DEN BERG et collab. [2008], qui utilise les *velocity obstacles* dans des scénarios multi-agents où plusieurs robots doivent se coordonner pour atteindre leurs destinations sans entrer en collision. Une convention d'esquive est ajoutée à l'évitement d'obstacles afin que les agents se contournent mutuellement sur le côté. Le chemin est ainsi optimisé grâce à cette convention qui est supposée être respectée par tous les agents. Une implémentation open-source de cet algorithme est disponible en ligne (VAN DEN BERG et collab. [2016]).

Les *velocity obstacles* peuvent être intégrés dans le cadre d'un environnement incertain, où les mouvements des obstacles suivent une loi aléatoire. Dans KLUGE [2003], une version stochastique des *velocity obstacles* est réalisée en itérant sur les différentes valeurs de vitesse possibles afin d'obtenir une approximation raisonnable des zones dangereuses dans l'espace des vitesses. Le principal défaut de ce genre d'approche est naturellement le coût de l'itération.

Le concept de *velocity obstacles* est étendu dans WILKIE et collab. [2009] pour un robot non-holonome auquel s'imposent des contraintes cinématiques. Le contrôle optimal (angle de braquage et vitesse du robot) est défini comme celui minimisant le temps de trajet jusqu'à la destination, en l'absence d'obstacles mobiles. L'approche *Generalized Velocity Obstacles* applique au robot un contrôle au plus proche du contrôle optimal, parmi ceux ne menant pas à une collision

avec les *velocity obstacles*. Il est à noter que les obstacles et le robot sont supposés pouvoir atteindre la même vitesse maximale, il est ainsi toujours possible pour le robot d'évoluer à la même vitesse que tout obstacle. Dans ces travaux les obstacles sont des disques qui avancent en ligne droite à vitesse constante et qui peuvent changer brusquement de direction à une fréquence aléatoire.

L'hypothèse de vitesse constante pour les obstacles mobiles est relâchée dans [LARGE et collab. \[2002\]](#) au profit d'une généralisation des *velocity obstacles* à des mouvements non linéaires (*Non Linear Velocity Obstacles*). L'hypothèse d'angle de braquage constant est notamment utilisée pour décrire le mouvement des obstacles mobiles, mais toute prédiction de trajectoire peut être utilisée en théorie.

Enfin, cette approche non linéaire est intégrée à une planification de mouvement adaptée à un robot véhiculaire ([VASQUEZ et collab. \[2004\]](#), [LARGE et collab. \[2005\]](#)). Ils utilisent une approche probabiliste pour prédire le mouvement des obstacles mobiles et la combinent avec les *Non Linear Velocity Obstacles* afin de représenter l'espace des vitesses accessibles. Ils proposent également d'améliorer la planification de trajectoire itérative de [FIORINI et SHILLER \[1998\]](#) en utilisant l'algorithme A* pour trouver le meilleur chemin sur un arbre qui est mis à jour dynamiquement, ce qui évite d'avoir à le recalculer entièrement à chaque requête.

3.2.4.5 Ensembles stochastiques atteignables

Certains travaux proposent des approches de planification de trajectoire robustes même dans le cas où le mouvement des obstacles peut changer de façon aléatoire. L'analyse des ensembles stochastiques atteignables (*Stochastic Reachable Set*, ou *SR set*) permet d'étendre la notion des *Reachable Sets* à un environnement non déterministe ([ABATE et collab. \[2008\]](#), [SUMMERS et LYGE-ROS \[2010\]](#)). Dans [CHIANG et collab. \[2015a\]](#) et [MALONE et collab. \[2017\]](#), la stratégie de planification de trajectoire combine un chemin statique global et une approche réactive locale qui évite les collisions avec les obstacles mobiles. Les obstacles mobiles se déplacent selon des trajectoires rectilignes ou circulaires avec une norme de vitesse qui varie aléatoirement dans le temps selon une loi de probabilité connue. Il existe plusieurs types d'obstacles, chacun suivant une loi de probabilité différente. La densité de probabilité de présence en coordonnées relatives pour chaque type d'obstacle est calculée hors-ligne dans un cache mémoire à partir de la loi de probabilité : c'est le *SR set* de l'obstacle. La dimensionnalité du cache correspond à celle de l'espace dans lequel les objets se déplacent : les différents pas de temps sont concaténés ensemble. Lors de la phase de planification, une carte globale des densités de probabilité de présence pour tous les obstacles peut ainsi être dressée rapidement en utilisant ces caches. Le chemin statique est calculé par un arbre RRT et rejoint la destination en évitant les obstacles fixes. La planification de mouvement est effectuée de manière réactive par un champ de potentiel artificiel qui attire le robot vers les nœuds du chemin statique. Le champ répulsif est conçu de sorte que son intensité est proportionnelle à la densité de probabilité de présence des obstacles mobiles (cf. Fig. 3.9).

Les mêmes auteurs ont amélioré la prédiction de l'approche précédente en remplaçant le calcul intégral de la densité de probabilité de présence par la méthode de Monte-Carlo dont le nombre limité de tirages rend le calcul du cache bien plus efficace (environ 21s pour la méthode de Monte-Carlo pour un obstacle, comparé à plus de 4000s pour la génération du *SR set* complet). La méthode *Stochastic Ensemble Simulation (SES)* ([CHIANG et collab. \[2015b\]](#)) combine également une planification de chemin globale statique et une planification locale dynamique à partir du cache de densité de présence. La différence notable de cette approche est que la dimension temporelle n'est pas perdue : une carte de densité de probabilité de présence est générée pour chaque type d'obstacle et à chaque pas de temps. La planification locale de chemin est également effectuée avec un arbre RRT avec une résolution plus fine et un horizon temporel plus court que le chemin global. Un test de collision est effectué pour chaque nouvelle branche créée afin de vérifier qu'aucune portion de chemin ne soit située à une densité de probabilité de présence d'obstacles supérieure à un seuil critique. Deux stratégies distinctes sont proposées pour développer l'arbre : « direct », une heuristique qui tend à rejoindre au plus vite le chemin statique et « full »,

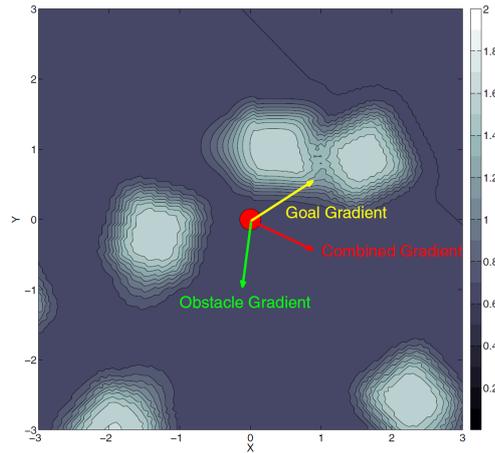


FIGURE 3.9 – méthode APF-SR : Illustration du champ de potentiel associé à la densité de probabilité de présence des obstacles mobiles, et des forces répulsives et attractives qui en dérivent (Fig. issue de CHIANG et collab. [2015a])

une recherche plus exhaustive de l'espace en cas d'échec de la première stratégie.

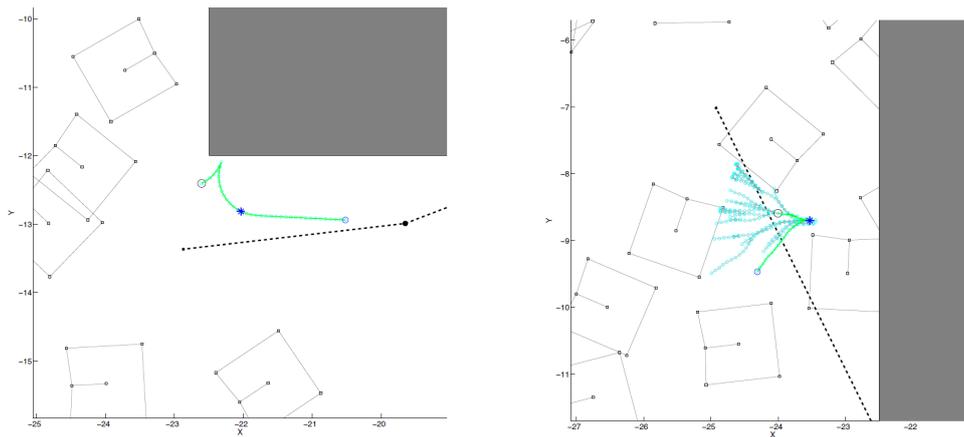


FIGURE 3.10 – méthode SES : À gauche l'heuristique « direct » qui tend à créer des branches (vertes) vers le chemin statique (noir). À droite la méthode « full » développe un arbre RRT plus complet lorsque la densité d'obstacles mobiles est trop forte (Fig. issue de CHIANG et collab. [2015b])

Une amélioration incrémentale de *SES* est le concept de *Dynamic Risk Tolerance (DRT)* introduit dans CHIANG et collab. [2017b]. L'approche utilise également la méthode de Monte Carlo pour estimer les probabilités de présence des obstacles stochastiques. Le seuil de risque utilisé lors du test de collision sur la carte des probabilités n'est plus constant, mais évolue dynamiquement en fonction du temps. Plusieurs profils de risque sont étudiés dans leurs travaux. Le profil retenu fixe un seuil constant qui accepte peu de risque à court terme puis qui augmente linéairement avec le temps afin d'accepter de plus en plus de risque à long terme. Cette gestion dynamique du risque donne plus de marge de manœuvre au robot à long terme tout en garantissant l'absence de collisions immédiates, ce qui surpasse les performances de *SES*.

3.2.4.6 États de collision inévitable

Le concept des états de collision inévitable (*Inevitable Collision States*, ou *ICS*) est introduit dans FRAICHARD et ASAMA [2003]. Ces états décrivent les situations dans lesquelles le robot n'est pas encore entré en collision avec un obstacle, mais telles qu'aucune de ses actions ne peut permettre d'éviter une collision future. Ils dépendent donc directement des capacités motrices du robot, en plus de la configuration des obstacles. Les *ICS* s'expriment formellement en décrivant

toutes les entrées de contrôle possibles du robot, et en calculant l'intersection des zones de collision inévitable spécifiques à chaque entrée de contrôle. Lorsque qu'aucune valeur de contrôle ne permet d'éviter la collision à partir d'un état, ce dernier est considéré comme un *ICS*. La caractérisation formelle des *ICS* nécessitant un calcul intensif, ils sont souvent l'objet d'une approximation plus rapide. Un algorithme vérifiant si un état donné est un *ICS* est proposé dans MARTINEZ-GOMEZ et FRAICHARD [2008], ce qui permet de ne pas avoir à calculer l'intégralité du domaine des *ICS* en amont.

Une planification de trajectoire partielle calculée à partir des *ICS* est introduite dans PETTI et FRAICHARD [2005]. Le fait de ne pas avoir de planification de trajectoire complète permet d'économiser du temps de calcul, et reste raisonnable dans un environnement dynamique où les prédictions à long terme restent incertaines. Cela nécessite cependant de s'assurer de la sûreté future de la trajectoire partielle, ce qui est facilité par l'utilisation des *ICS*. Un arbre RRT est développé dans l'espace de configuration du robot, en ne prenant pour nœuds que des états situés en dehors des *ICS*. Une vérification rapide de la sûreté est implémentée afin de générer les nouveaux nœuds. Trois trajectoires à angle de braquage constant sont générées à partir de ce nouvel état, en appliquant la décélération maximale du robot : le nouveau nœud est considéré comme étant en dehors des *ICS* à condition qu'au moins une de ces trajectoires soit exempte de collision.

La stratégie d'évitement d'obstacles présentée dans MARTINEZ-GOMEZ et FRAICHARD [2009] ne cherche pas à atteindre une destination, mais seulement à esquiver les obstacles mobiles en suivant une trajectoire qui évite les *ICS*. Cette approche réactive est effectuée par tirage aléatoire en appliquant la première entrée de contrôle admissible échantillonnée qui ne mène pas dans un *ICS*. Les auteurs introduisent la notion de *Safe Control Kernel*, un ensemble d'entrées de contrôle qui garantissent l'évitement de collision et qui est ajouté dans les éléments échantillonnés avant chaque tirage. Les obstacles mobiles suivent des trajectoires complexes suivant des lois aléatoires, mais une connaissance complète de leur mouvement jusqu'à un horizon temporel est fournie à la planification. Dans ces scénarios, l'accélération du robot est bornée mais pas sa vitesse, tandis qu'une limite de vitesse est appliquée aux obstacles mobiles.

3.2.4.7 Fenêtre dynamique

L'approche de la fenêtre dynamique (*Dynamic Window*) utilise une modélisation des capacités motrices du robot pour générer une approximation des trajectoires lui étant accessibles. Dans FOX et collab. [1997], une version modélisant ces trajectoires comme formant des séquences d'arcs de cercle est proposée. L'approche nécessite une représentation des trajectoires en arcs de cercle admissibles pour le robot dans un espace en deux dimensions : l'angle de braquage et la vitesse. Les trajectoires rentrant en collision avec un obstacle sont éliminées en indiquant les zones correspondantes dans cet espace. La fenêtre dynamique est ensuite calculée en indiquant les vitesses pouvant être atteintes en pratique par le robot en tenant compte de ses propriétés dynamiques (vitesse courante, accélération maximale, etc). L'entrée de contrôle optimale (vitesse et angle de braquage) est finalement déterminée par une fonction de coût qui :

1. minimise la distance à vol d'oiseau entre le robot et la destination
2. maximise la distance longitudinale avant la frontière d'un obstacle
3. maximise la vitesse instantanée du robot

Les auteurs de SEDER et PETROVIC [2007] étendent l'approche à un environnement dynamique (*Time Varying Dynamic Window*). Une hypothèse de trajectoire circulaire est prise pour anticiper le mouvement des obstacles, ainsi que pour caractériser le déplacement du robot. Une grille d'occupation est utilisée pour représenter les obstacles mobiles. Une trajectoire circulaire est ainsi attribuée pour chaque cellule de la grille. Un test de collisions potentielles est ensuite effectué avec cette grille d'occupation pour chaque hypothèse de trajectoire du robot. La grille est ensuite mise à jour en supprimant les positions initiales des obstacles et en ajoutant les positions où ces collisions potentielles sont détectées. Elle est utilisée pour calculer la trajectoire optimale pour le

robot. Le chemin optimal est déduit en appliquant une fonction de coût sur les trajectoires n'entrant pas en collision.

3.2.4.8 Champs de potentiel artificiel

De nombreux travaux étendent l'approche des champs de potentiel (*Artificial Potential Fields*) aux environnements dynamiques. Parmi ceux-ci, NAK YONG KO et BUM HEE LEE [1996] intègre la vitesse des obstacles dans la caractérisation du champ de potentiel répulsif. Dans GE et CUI [2002] et HUANG [2009], la fonction de potentiel tient compte à la fois de la position relative et de la vitesse relative de chaque obstacle par rapport au robot, ainsi que des position et vitesse relatives du robot par rapport à la cible (qui est éventuellement mobile). Une approche de planification de trajectoire est proposée dans TSOURVELOUDIS et collab. [2001] en combinant un champ de potentiel électrostatique (*Electrostatic Potential Field*) à un mécanisme de logique floue (*fuzzy logic engine*).

3.2.5 Conclusion

Le tableau synoptique qui suit compare les 8 principales approches présentées ci-dessus, sur la base des 3 critères suivants :

1. *Possibilité d'une planification complète* : est-il possible d'obtenir un chemin complet entre le point de départ et le point d'arrivée, qui permet d'anticiper la trajectoire du robot ego sur le long terme?
2. *Obstacles très dynamiques* : l'approche permet-elle de traiter un environnement très dynamique où les vitesses des obstacles peuvent dépasser celle du robot ego?
3. *Qualité du chemin* : la trajectoire planifiée est-elle optimale en terme de longueur et de temps de parcours?

Dans le Tableau 3.1, les cellules en rouge indiquent un défaut rédhibitoire pour que la planification de trajectoire en question soit utilisée en présence d'obstacles très dynamiques. Les cellules en orange indiquent un défaut notable qui n'empêche cependant pas le recours à l'approche.

| | Planification complète | Obstacles fortement dynamiques | Qualité du chemin |
|---|---|--|---|
| L'espace temps-état | En présence d'obstacles fortement dynamiques, la complexité algorithmique est trop importante. Il faut fixer un chemin statique ou bien considérer des obstacles moins mobiles que le robot | | Bonne qualité du chemin |
| Treillis d'états | L'espace de recherche doit être fortement réduit (ex : circulation sur une voie) | Compatible | Bonne qualité du chemin, qui dépend néanmoins de la densité du treillis |
| Cône de collision | Contrôle réactif, pas de planification de trajectoire | Seule la vitesse instantanée des obstacles mobiles est prise en compte (hormis NLVO) | La trajectoire suivie est assez discontinue du fait du contrôle réactif |
| V-Obstacles | | | |
| Champs de Potentiel Artificiel | | | |
| Fenêtre Dynamique | | | |
| États de Collision Inévitable | Généralement utilisés avec des stratégies de planification partielle | Compatible | Généralement utilisés avec des planifications par échantillonnage (forte discontinuité) |
| Ensembles Stochastiques Atteignables | Oui | Compatible | Utilisé avec des planifications par échantillonnage (forte discontinuité) |

TABLEAU 3.1 – Tableau récapitulatif des principales approches robotiques de planification de trajectoire en présence d'obstacles mobiles

Cet état de l'art des approches robotiques en matière de planification de trajectoire permet de constater que la majorité des approches ne sont pas compatibles avec la présence d'obstacles potentiellement plus rapides que le robot ego, dit « obstacles très dynamiques ».

Une nouvelle formulation duale du problème de planification va être introduite. Elle servira de point de départ à nos algorithmes de recherche de chemin. Cette approche ne peut être directement mise en relation avec les familles d'algorithmes présentées dans ce tableau. Ses performances pourront néanmoins être comparées avec celles de l'état de l'art, notamment les approches réactives les plus utilisées comme *APF* et *velocity obstacles*, ainsi que les approches adaptées aux mouvements stochastiques des obstacles comme celles fondées sur les ensembles stochastiques atteignables.

3.3 Reachable Interaction Sets : présentation de l'approche

On introduit ici une nouvelle représentation temps-espace, les *Reachable Interaction Sets* ou RIS, qui tient compte d'une hypothèse sur le profil de vitesse du robot ego. Bien que cette représentation tienne compte de la dimension temporelle, elle est effectuée dans l'espace de configuration du robot, donc dans le plan 2D pour le cas d'un robot se déplaçant sur une surface. Cette propriété permet d'assimiler la planification de trajectoire dans un environnement dynamique à un problème de planification de chemin sur une carte indiquant les RIS comme pseudo-obstacles. Planifier un chemin qui contourne ces pseudo-obstacles quasi-statiques est un problème dual qui revient à calculer une trajectoire qui contourne les obstacles mobiles au cours du temps.

On suppose que le mouvement des obstacles mobiles peut être raisonnablement anticipé durant un horizon temporel t_H . Si on fixe une valeur pour la norme de la vitesse du robot ego et si l'on suppose que sa vitesse de lacet restera constante durant cette même période, le problème de planification de trajectoire revient localement à choisir la valeur de la vitesse de lacet. En calculant l'intersection des trajectoires prédites des obstacles mobiles avec la zone accessible du robot (son *Reachable Set*), on obtient les zones d'interactions accessibles dans lesquelles des collisions sont possibles : ce sont les *Reachable Interaction Sets*. Si les hypothèses sur la prédiction du mouvement des obstacles et sur le type de mouvement du robot ego sont avérées, aucune collision ne pourra intervenir en dehors des RIS. La planification de trajectoire revient alors à rechercher un chemin qui évite ces zones, en respectant au mieux l'hypothèse de mouvement de vitesse de lacet constante pour le robot ego. L'utilisation d'un horizon de temps fuyant (*Receding Horizon*) permet de compenser à chaque pas de temps les erreurs qui peuvent être introduites lorsque les différentes hypothèses ne sont pas tout à fait valides.

3.3.1 Cas d'un robot holonome

Considérons le cas simple d'un robot ego holonome A ponctuel se déplaçant à vitesse constante $v_A = \|\vec{v}_A\|$ en présence d'un obstacle B dont la forme est un polygone P_B et dont la trajectoire future est parfaitement connue et est notée $\{(x_B(t), y_B(t), \theta_B(t)), t \in [0, t_H]\}$, où t_H est l'horizon temporel. L'hypothèse que A est ponctuel n'est en soit pas restrictive car le recours à un espace de configuration approprié peut ramener toute forme du robot à cette formulation du problème.

Supposons, sans perte de généralité, que A soit initialement situé à l'origine O du repère et que B ait pour position $p_B = (x_B, y_B)$ à $t = 0$. Lorsque A se déplace à vitesse constante, la position de $A(t)$ est située sur un cercle $C(O, t \cdot v_A)$ de rayon $t \cdot v_A$ centré sur l'origine O. Lorsque B se déplace durant l'intervalle $[0, t_H]$, l'obstacle $B(t)$ correspond au polygone P_B translaté et orienté selon la trajectoire prédite. Une collision peut survenir entre A et B lorsque $A(t)$ et $B(t)$ se superposent. Les $RIS_{AB}(0, t_H)$ sont alors définis comme suit :

$$RIS_{AB}(0, t_H) = \{p \in A(t) \cap B(t) \text{ for } t \in [0, t_H]\} \quad (3.1)$$

Afin d'interpréter géométriquement l'ensemble $RIS_{AB}(0, t_H)$ défini par l'équation 3.1, on considère l'espace-temps 3D (X, Y, t) en ajoutant la dimension temporelle.

- Les points (x, y, t) des futures positions accessibles par A appartiennent à un cône droit à base discoïdale ayant l'origine $(0, 0, 0)$ pour apex et le cercle $C((0, 0), t_H \cdot v_A)$ pour base dans le plan (X, Y, t_H) .
- Les points (x, y, t) des positions futures atteintes par B appartiennent à un volume \mathcal{B} qui est l'extrusion du polygone P_B par la courbe 3D correspondant à la trajectoire de B dans l'intervalle $[0, t_H]$. Le volume \mathcal{B} est compris entre le polygone $P_B(0)$ dans le plan $(X, Y, 0)$ et le polygone $P_B(t_H)$ dans le plan (X, Y, t_H) .
- L'ensemble $RIS_{AB}(0, t_H)$ correspond à la projection sur le plan $(X, Y, 0)$ de la surface résultant de l'intersection du cône avec \mathcal{B} (cf. Fig. 3.11).

$RIS_{AB}(0, t_H)$ peut être vide ou composé de plusieurs régions non connexes. Dans le cas où la trajectoire de B est rectiligne, $RIS_{AB}(0, t_H)$ est composé d'au plus deux régions distinctes.

Enfin, si \mathcal{O} est l'ensemble des obstacles présents dans le voisinage de A durant $[0, t_H]$, nous avons $RIS_A(0, t_H) = \cup_{B \in \mathcal{O}} RIS_{AB}(0, t_H)$.

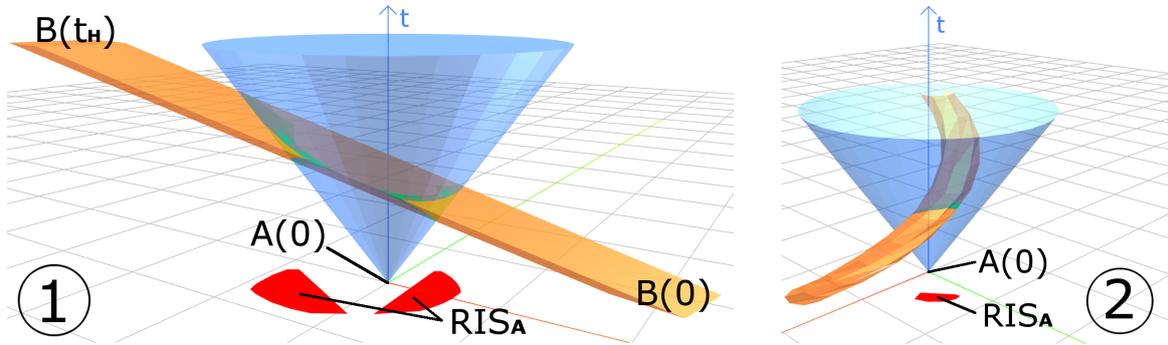


FIGURE 3.11 – Deux exemples des régions $RIS_{AB}(0, t_H)$ avec un robot A holonome. Sur le plan $t = 0$, $RIS_{AB}(0, t_H)$ est affiché en rouge. La trajectoire de B est représentée par le volume orange et le cône des positions accessibles par A par le cône bleu. Sur (1), B suit une trajectoire rectiligne uniforme et le RIS est constitué de deux régions séparées. Sur (2), B suit une trajectoire circulaire uniforme et le RIS forme une seule région connexe.

Le calcul de $RIS_{AB}(0, t_H)$ présenté ci-dessus peut être étendu à des cas de mouvement plus complexes du robot. Il suffit de construire la surface \mathcal{S}_A représentant les positions potentielles de A dans l'espace-temps en considérant les nouvelles contraintes de mouvement de A, de calculer son intersection avec le volume \mathcal{B} pour chaque obstacle B voisin de A dans l'intervalle de temps $[0, t_H]$ et de projeter cette intersection sur le plan $(X, Y, 0)$.

3.3.2 Cas d'un robot soumis à des contraintes dynamiques non-holonomes

Considérons à présent un autre modèle dynamique pour le robot A et supposons qu'il se déplace avec les contraintes du véhicule de Dubins en matière d'accélération et de vitesse de lacet (cf. LAVALLE [2006]) :

$$\begin{cases} \dot{x}(t) = v(t) \cdot \cos(\theta(t)) \\ \dot{y}(t) = v(t) \cdot \sin(\theta(t)) \\ \dot{\theta}(t) = \omega(t) \end{cases} \quad (3.2)$$

$$v(t) = \min(v_A(0) + a_{max} \cdot t, v_{Amax}) \quad (3.3)$$

Où $\omega(t)$ est la vitesse de lacet et $v(t)$ la vitesse longitudinale du robot ego.

La surface conique représentant les positions accessibles par A dans l'espace-temps (X, Y, t) est alors remplacée par la surface \mathcal{S}_A construite par les points (x, y, t) selon l'équation 3.4 :

$$\begin{cases} \forall u \in [0, t_H], \\ \omega(t) = \omega_{max}, \forall t \in [0, u] \\ \omega(t) = 0, \forall t \in [u, t_H] \end{cases} \quad (3.4)$$

Afin d'envisager une prédiction « pire-cas » qui maximise la portée de déplacement du robot ego, on suppose que ce dernier se déplace en tournant avec la vitesse de lacet maximale ω_{max} durant une durée u , puis avance en ligne droite jusqu'à la fin de l'horizon temporel t_H . De plus, on suppose que la vitesse longitudinale $v(t)$ prend initialement pour valeur $v_A(0)$, la vitesse instantanée de A, puis applique l'accélération maximale a_{max} de manière constante jusqu'à obtenir la vitesse maximale atteignable v_{Amax} (cf. équation 3.3).

La surface \mathcal{S}_A représentant les positions potentielles de A dans l'espace-temps est construite en agrégeant toutes les valeurs de u , et en ajoutant par symétrie les trajectoires pour lesquelles le robot tourne dans l'autre direction ($\omega(t) = -\omega_{max}$). Si la surface \mathcal{S}_A s'auto-intersecte, les régions de la surface se retrouvant à l'intérieur sont supprimées avant de calculer l'intersection avec le volume \mathcal{B} correspondant à la trajectoire de l'obstacle B. Un exemple de la surface \mathcal{S}_A est présenté Fig. 3.12.

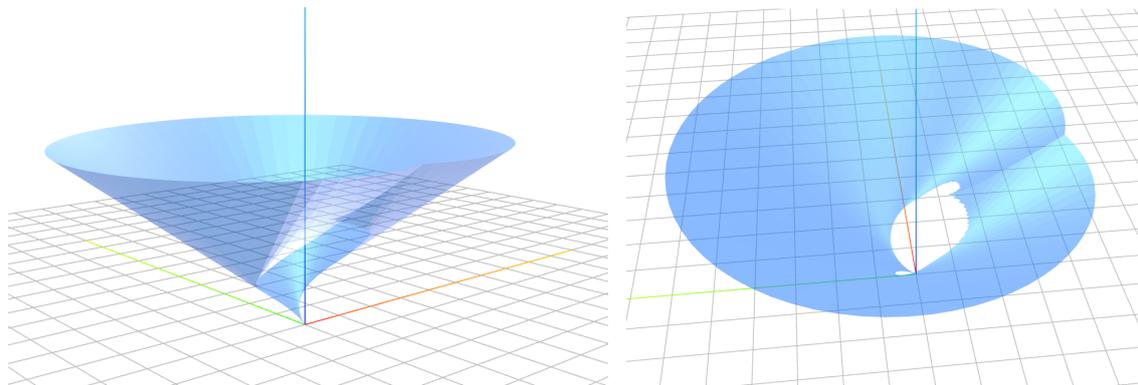


FIGURE 3.12 – Deux vues d'une même surface \mathcal{S}_A représentant le mouvement d'un robot non holonome suivant un chemin tel que présenté par l'équation 3.4.

En pratique, la surface \mathcal{S}_A est calculée une fois pour toutes à partir des propriétés du véhicule et stockée afin d'être réutilisée rapidement lors de la planification. De la même manière que pour le cas holonome, $RIS_{AB}(0, t_H)$ correspond à la projection de l'intersection entre \mathcal{S}_A et \mathcal{B} sur le plan 2D $(X, Y, 0)$. Une vue interactive en 3 dimensions de la génération de plusieurs $RIS_{AB}(0, t_H)$ est disponible sur <http://deboc.fr/ris>.

3.3.3 Cas où les mouvements des obstacles mobiles ne sont pas déterministes

Nous nous intéressons ici au cas où les mouvements des obstacles mobiles ne sont pas déterministes, mais suivent des lois aléatoires. En supposant que le processus stochastique qui régit le mouvement d'un obstacle soit connu, il est possible de l'utiliser pour anticiper les mouvements futurs de l'obstacle et planifier une trajectoire. Cependant, il est nécessaire de prendre en compte l'incertitude de cette prédiction.

La stratégie utilisée est inspirée des travaux de [CHIANG et collab. \[2015a\]](#). Au cours du temps, la position exacte d'un obstacle n'est pas connue mais on peut estimer la zone dans laquelle il risque d'être présent selon un certain seuil de probabilité choisi. A chaque pas de temps, le polygone qui représente l'obstacle sera non plus défini par sa forme mais par la zone où la probabilité de présence de l'obstacle dépasse un certain seuil.

Considérons que la vitesse de chaque obstacle varie au cours du temps selon une loi de probabilité connue, et que sa vitesse angulaire (i.e. son rayon de braquage) est constante. Une représentation de la probabilité de présence de chaque obstacle dans l'espace-temps peut être établie par une collection de cartes de densité de probabilité de présence de l'obstacle dans le plan 2D, une carte étant générée pour chaque pas de temps t de l'intervalle $[0, t_H]$.

Dans les scénarios considérés, les mouvements des différents obstacles mobiles suivent une même loi stochastique. Les cartes de densité de probabilité associées à cette loi sont calculées en effectuant 500 tirages de Monte-Carlo qui simulent le mouvement d'un obstacle sur la durée de

l'horizon temporel. La collection de cartes dépend de la vitesse angulaire fixée pour l'obstacle, de sorte qu'il est nécessaire de la générer pour chaque valeur de vitesse angulaire possible du scénario. Pour une loi stochastique donnée, les cartes de probabilité de présence ainsi générées sont sauvegardées en cache afin d'être utilisées rapidement lors de la phase de planification. Trois exemples de ces cartes de probabilité de présence sont présentés en Fig. 3.14, 3.15 et 3.16 : la vitesse longitudinale de l'obstacle est déterministe (fixée à 2.24 unités par pas de temps) pour la collection de cartes de la Fig. 3.14, et les deux lois stochastiques exprimées en Fig. 3.13 sont utilisées pour définir la vitesse dans les cartes présentées en Fig. 3.15 et 3.16.

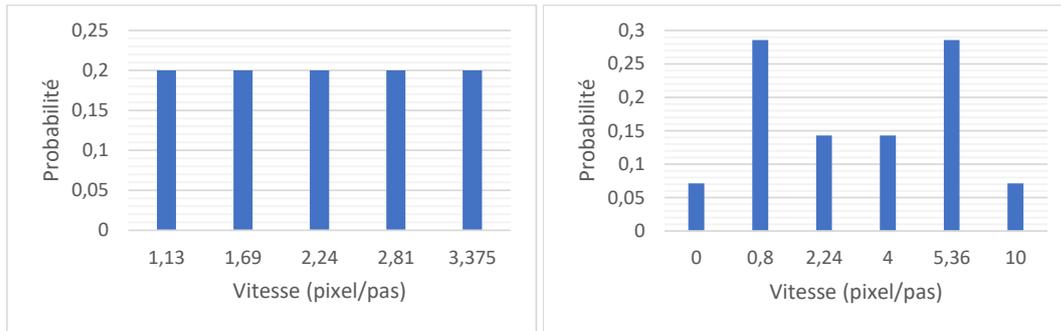


FIGURE 3.13 – Deux lois stochastiques pour la vitesse longitudinale d'un obstacle mobile.

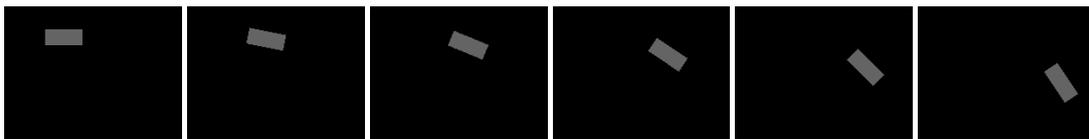


FIGURE 3.14 – Cartes de densité de probabilité de présence pour un obstacle suivant une vitesse déterministe. Le rayon de braquage de l'obstacle mobile est fixe.

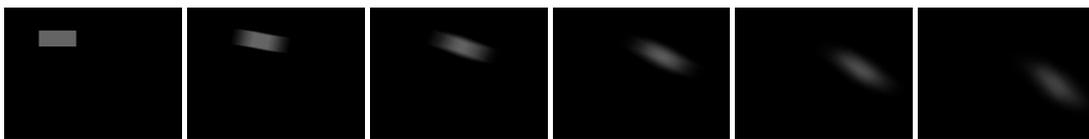


FIGURE 3.15 – Cartes de densité de probabilité de présence pour un obstacle suivant une vitesse stochastique à variance modérée. Le rayon de braquage de l'obstacle mobile est fixe.

Lors du calcul du RIS, ces cartes sont utilisées en les translatant et les orientant selon la position des obstacles mobiles à chaque instant t . Un seuil de probabilité est introduit afin de déterminer un contour d'obstacle pour chaque pas de temps jusqu'à l'horizon temporel. Les polygones résultants correspondent aux zones où la probabilité de présence future de l'obstacle est la plus forte. Pour chaque obstacle B , un volume 3D \mathcal{B} formé en connectant les polygones créés à chaque instant t de l'intervalle $[0, t_H]$ remplace l'extrusion du polygone P_B par la courbe 3D correspondant à la trajectoire de B dans l'intervalle $[0, t_H]$ construit dans le cas où le mouvement de B est déterministe.

L'influence du seuil de probabilité est très importante sur la forme des polygones retenus comme obstacles à éviter. En fixant un seuil trop élevé, seules les zones où l'on est sûr de trouver un obstacle mobile seront retenues pour générer des polygones, ce qui peut s'avérer risqué. À l'inverse, avec un seuil trop faible, l'espace navigable risque d'être encombré de polygones à éviter alors qu'ils représentent une faible probabilité de présence d'un obstacle mobile.

Au lieu d'utiliser un seuil statique, nous nous sommes inspirés de l'approche de seuil dynamique introduite dans [CHIANG et collab. \[2017b\]](#), qui consiste à faire varier dynamiquement le seuil en fonction du temps, afin de garantir l'évitement de collisions à court terme tout en limitant



FIGURE 3.16 – Cartes de densité de probabilité de présence pour un obstacle suivant une vitesse stochastique à forte variance. Le rayon de braquage de l'obstacle mobile est fixe.

l'occupation de l'espace par des polygones pour les pas de temps proches de l'horizon temporel. Cependant les auteurs utilisent une fonction linéaire pour définir le seuil, ce qui nécessite de définir empiriquement un seuil minimal P_{min} , un seuil maximal P_{max} , ainsi que deux durées de transition t_1 et t_2 (cf. équation 3.5).

$$\begin{cases} \text{pour } t < t_1, & \text{seuil} = P_{min} \\ \text{pour } t_1 \leq t \leq t_2, & \text{seuil} = P_{min} + (P_{max} - P_{min}) \cdot \frac{t-t_1}{t_2-t_1} \\ \text{pour } t > t_2, & \text{seuil} = P_{max} \end{cases} \quad (3.5)$$

Nous avons choisi d'ajuster le seuil en fonction d'un seul paramètre $Aire_{max}$, homogène à la surface totale des polygones d'obstacles présents dans l'environnement. Notre approche consiste à ajuster le seuil à chaque pas de temps de sorte que la surface totale des contours d'obstacles ne soit jamais supérieure à $Aire_{max}$ (cf. Algorithme 2). De plus, la valeur du seuil doit être une fonction monotone croissante sur $[0, t_H]$.

```

Data : cartesObstacle, t, Airemax, dernierSeuil
Result : polygoneObstacle(t), dernierSeuil
seuil = dernierSeuil;
polygoneObstacle = calculerPolygoneSeuil(cartesObstacle(t), seuil);
while calculerAirePolygone(polygoneObstacle) > Airemax do
    |   seuil += 1;
    |   polygoneObstacle = calculerPolygoneSeuil(cartesObstacle(t), seuil);
end
return polygoneObstacle, seuil;

```

Algorithme 2 : Calcul du polygone par seuillage dynamique à un pas de temps t donné

La Fig. 3.17 montre différents volumes \mathcal{B} construits avec différentes valeurs de seuil pour un même obstacle B .

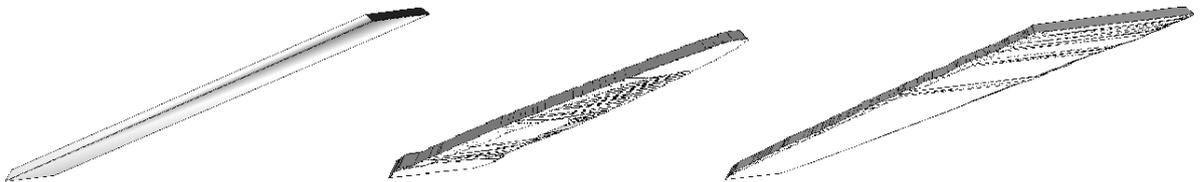


FIGURE 3.17 – Influence du seuil sur le volume de l'obstacle stochastique : à gauche un obstacle au mouvement déterministe, au milieu la densité de probabilité de présence d'un obstacle au mouvement stochastique avec un seuil élevé (les zones de probabilité faible sont éliminées), à droite la même densité de probabilité avec un seuil plus faible (toutes les zones sont conservés et elles s'étendent du fait de l'incertitude)

Comme dans le cas déterministe, $RIS_{AB}(0, t_H)$ correspond aux zones formées par la projection sur le plan $t = 0$ de l'intersection de \mathcal{B} avec la surface associée au mouvement du robot A dans l'espace-temps (X, Y, t) .

3.4 Application des RIS à la planification de mouvement

Les RIS présentés dans la section précédente sont utilisés dans le calcul de planification de mouvement d'un robot A se déplaçant à une vitesse $v_A(t)$, de norme inférieure à v_{Amax} , dans un environnement comportant des obstacles fixes et mobiles et se dirigeant vers une destination $Dest$.

Deux stratégies de planification sont proposées : des méthodes locales de planification dynamique où la gestion des collisions avec les obstacles mobiles via les RIS s'effectue localement jusqu'à une destination intermédiaire et une méthode globale de planification où l'évitement des collisions s'effectue jusqu'à la destination finale $Dest$.

Notations : afin de simplifier les notations, les paramètres temporels seront considérés comme implicites de sorte que les notations $RIS_{AB}(0, t_H)$, $RIS_A(0, t_H)$, $A(t)$, $\vec{v}_A(t)$ et $B(t)$ seront remplacées par RIS_{AB} , RIS_A , A , \vec{v}_A et B . \mathcal{O} représentera l'ensemble des obstacles mobiles B présents dans le voisinage de A durant l'intervalle de temps considéré.

3.4.1 Stratégies locales de planification dynamique

Ces méthodes combinent deux approches de planification de chemin : une planification globale de chemin qui ne tient compte que des obstacles statiques et une planification de mouvement dynamique qui adapte localement le chemin selon les mouvements des obstacles mobiles présents dans le voisinage du robot ego A.

Planification globale évitant les obstacles statiques

La planification globale est effectuée seulement une fois pour chaque environnement statique donné. Toute méthode de planification de chemin statique pourrait convenir pour générer un chemin C_G évitant les collisions et rejoignant le point de destination $Dest$. Cette planification globale est combinée avec les planifications locales comme suit :

- Le chemin C_G est construit en utilisant l'approche par échantillonnage RRT* (cf. [KARAMAN et FRAZZOLI \[2011\]](#)), en considérant seulement les obstacles statiques, la position initiale du robot et sa destination.
- Les différentes planifications locales présentées dans la suite utilisent ce chemin C_G comme guide afin de contourner les obstacles statiques et ainsi d'éviter de rester bloqué dans une situation d'extremum local. A chaque pas de temps, un point intermédiaire D est sélectionné sur C_G et est utilisé comme destination locale pour la méthode de planification locale de chemin. Le choix de D est effectué en mesurant la distance $dist_{AC}$ entre A et C_G et en établissant le point $proj_A$, projeté de A sur C_G : D est le premier point de C_G situé après $proj_A$ et situé à une distance $dist_D = \max(\maxDist - dist_{AC}, 0)$ du point $proj_A$ (cf. Fig. 3.18), où \maxDist est la distance que A peut parcourir durant un temps donné (par la suite, cette durée est fixée à $\frac{4}{5} \cdot t_H$).

Cette définition de la distance $dist_D$ permet d'orienter le robot A vers le chemin lorsqu'il est trop éloigné de son projeté $proj_A$ et de le diriger vers un point situé plus en avant lorsqu'il est assez proche du projeté.

Les planifications locales étudiées

Les RIS présentés précédemment sont intégrés au sein de différentes approches de planification locale de chemin :

- **ris-*apf*** : une méthode réactive utilisant des champs de potentiel artificiel,
- **ris-*bézier*** : une méthode de planification utilisant des courbes de Bézier,
- **ris-*hybrid*** : une méthode hybride combinant les deux autres approches.

Ces approches suivent ces mêmes étapes à chaque pas de temps :

1. la destination intermédiaire D est déterminée en fonction des paramètres du robot A,

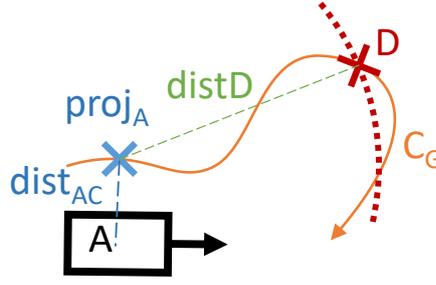


FIGURE 3.18 – Définition de la destination intermédiaire D à partir du projeté du robot sur le chemin global C_G . Plus le robot est proche de son projeté, plus D est avancé en aval sur C_G .

2. La collection des polygones RIS_A est mise à jour en fonction des différents obstacles B de \mathcal{O} et des paramètres de A, comme expliqué dans la section précédente,
3. la planification locale est effectuée afin de trouver un chemin reliant A à D en évitant les obstacles formés par RIS_A ,
4. le vecteur vitesse de consigne est transmis au module de contrôle qui adapte la vitesse du robot A en fonction de ses capacités motrices réelles.

A partir d'un vecteur vitesse de consigne, le module de contrôle met à jour la vitesse et l'orientation de A en fonction du modèle dynamique utilisé. Pour un robot holonome, la direction du vecteur consigne est conservée, et sa norme est bornée par v_{Amax} . Pour un robot non-holonome respectant le modèle de Dubins, l'angle de la direction du vecteur ne peut varier de plus de $\pm\omega_{max}$, et la dérivée de sa norme est bornée par une accélération/décélération maximale de $\pm a_{max}$.

Parfois, A peut se trouver à l'intérieur d'une région de RIS_A sans pour autant être en collision avec un obstacle. Une telle situation peut survenir dans le cas où la prédiction de mouvement d'un obstacle s'avère trop peu précise, ou bien en cas de légère erreur numérique du module de contrôle, ou encore lorsque l'hypothèse initiale sur le mouvement de A (chemin rectiligne ou construit selon l'équation 3.4) n'est pas respectée. Ce dernier cas survient pratiquement toujours lorsque le robot est en train de contourner un obstacle mobile. Afin de minimiser le risque induit par cette situation, la forme de chaque obstacle B est dilatée d'une marge de sécurité m en effectuant une somme de Minkowski entre un disque de rayon m et le polygones P_B de \mathcal{B} , avant de calculer les régions RIS_{AB} correspondantes. Chaque approche de planification locale propose une stratégie pour éloigner A de RIS_A quand il est situé à l'intérieur d'une région de RIS_A .

3.4.1.1 Planification réactive avec champ de potentiel

La planification *ris-apf* consiste à établir un champ de potentiel quasi-statique afin d'effectuer un contrôle réactif. Les régions de RIS_A remplacent les obstacles B de \mathcal{O} . L'hypothèse quasi-statique consiste à supposer que ces zones à éviter ne sont pas modifiées au cours du temps. La définition des potentiels répulsif et attractif (voir équations 3.6, 3.7 et 3.8) sont celles d'un champ de potentiel standard dans un environnement statique comme dans MINGUEZ et collab. [2008] (cf. Fig. 3.19(a)).

$$\vec{F}_{rep}(B) = -K_{rep} \left(\frac{1}{d(A, RIS_{AB})^2} - \frac{1}{d_0^2} \right) \vec{u}_{A-RIS} \quad (3.6)$$

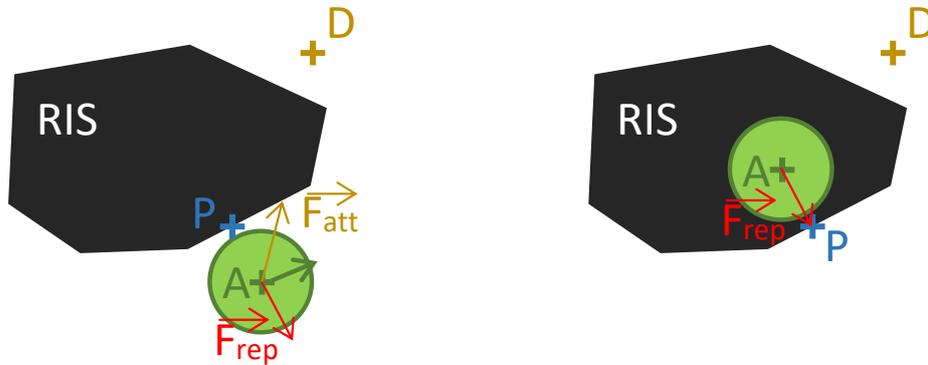
$$\vec{F}_{att} = K_{att} \frac{\vec{AD}}{\|\vec{AD}\|} \quad (3.7)$$

$$\vec{F}_{total} = \vec{F}_{att} + \sum_{B \in \mathcal{O}} \vec{F}_{rep}(B) \quad (3.8)$$

Où $\vec{u}_{A-RIS} = \frac{\vec{AP}}{\|\vec{AP}\|}$, P étant le point le plus proche de A sur le bord de RIS_A . Les valeurs de la portée d'influence des obstacles d_0 , ainsi que celles des poids K_{att} et K_{rep} sont fixées empiriquement (ici, $d_0 = 500$, $K_{att} = 10$ et $K_{rep} = 20000$).

Le vecteur vitesse consigne de A prend directement la valeur de la force résultante qui dérive des champs de potentiel.

Dans le cas où A se situe à l'intérieur d'une région de RIS_A , le signe du champ de potentiel répulsif est inversé afin d'attirer le robot vers le bord extérieur le plus proche (cf. Fig. 3.19(b)).



(a) La vitesse de consigne du robot est composée d'un vecteur répulsif calculé à partir du projeté P du robot A sur le bord du polygone et d'un vecteur attractif vers la destination D.

(b) Lorsque le robot est à l'intérieur d'un RIS, seul le champ de potentiel répulsif est utilisé pour le faire sortir du polygone. Dans cette situation, le vecteur est dans la direction du projeté P.

FIGURE 3.19 – Composantes des potentiels de *ris-apf*

3.4.1.2 Planification locale avec courbe de Bézier

L'objectif de cette approche locale est de construire une réelle planification sur le moyen terme plutôt qu'une approche de contrôle réactif.

La planification *ris-bézier* génère un chemin constitué d'une ou deux courbes de Bézier (MORTENSON [1997]) jointes continûment et dont les extrémités partagent une même tangente. De plus, le chemin reliant A à D ainsi constitué vérifie une **continuité G1** et sa courbure est minimisée, ce qui donne au chemin le double avantage de pouvoir être suivi par un robot non-holonome et qui lui évite d'invalider l'hypothèse initiale sur le mouvement de A. Cette planification simple, limitée à deux courbes de courbure maximale bornée, a l'avantage d'être légère en calculs mais en contrepartie peut échouer à fournir un chemin dans certaines configurations qui ne sont pourtant pas insolubles.

Différents paramètres sont mis à jour à chaque pas de temps (cf. Fig. 3.20) : les positions de A et de D, la vitesse instantanée \vec{v}_A de A et \vec{u}_D l'orientation du robot désirée au point D (cf. Fig. 3.20.1). La première étape de la planification consiste à construire la courbe de Bézier \mathcal{C}_0 reliant A à D. Elle est définie par les points de contrôle A, Ca, Cd, D (cf. Fig. 3.20.2), où Ca est placé de sorte que $A\vec{C}a = d \cdot \frac{\vec{v}_A}{\|\vec{v}_A\|}$ et Cd tel que $D\vec{C}d = d \cdot \vec{u}_D$ avec $d = \min(d_0, \frac{d(A,D)}{2})$, (d_0 étant fixé empiriquement à 40).

Si la courbe \mathcal{C}_0 n'intersecte pas RIS_A , elle est utilisée directement comme chemin de consigne. Dans le cas contraire, \mathcal{C}_0 est remplacée par \mathcal{C}_1 qui est composée de deux courbes de Bézier définies par les points de contrôle (A, Ca, Ci₁, I) et (I, Ci₂, Ct, T). \mathcal{C}_1 doit être de continuité G1 au point de jonction I et ne doit pas intersecter RIS_A .

Les positions des points I, Ci₁ et Ci₂ (cf. Fig. 3.20.3 et 3.20.4) sont calculées comme suit. Soit O le premier point de \mathcal{C}_0 intersectant RIS_A , \vec{w}_0 le vecteur unitaire tangent à \mathcal{C}_0 au point O et l_0 la droite perpendiculaire à \vec{w}_0 et passant par O. Le point I est le point de l_0 le plus proche de 0 qui permet à \mathcal{C}_1 de ne pas intersecter RIS_A . Ci₁ et Ci₂ sont placés sur la droite de direction \vec{w}_0 qui passe par I. Ils sont situés de part et d'autre du point I, à une distance d de celui-ci. La position de I est recherchée par itération de part et d'autre de O sur la droite l_0 en utilisant des pas d'itération empiriquement fixés à 10.

Lorsque la distance entre O et I est trop grande, la courbure du chemin \mathcal{C}_1 devient trop grande pour que la courbe puisse être vraiment utilisable comme chemin. Dans ce cas de figure, la dé-

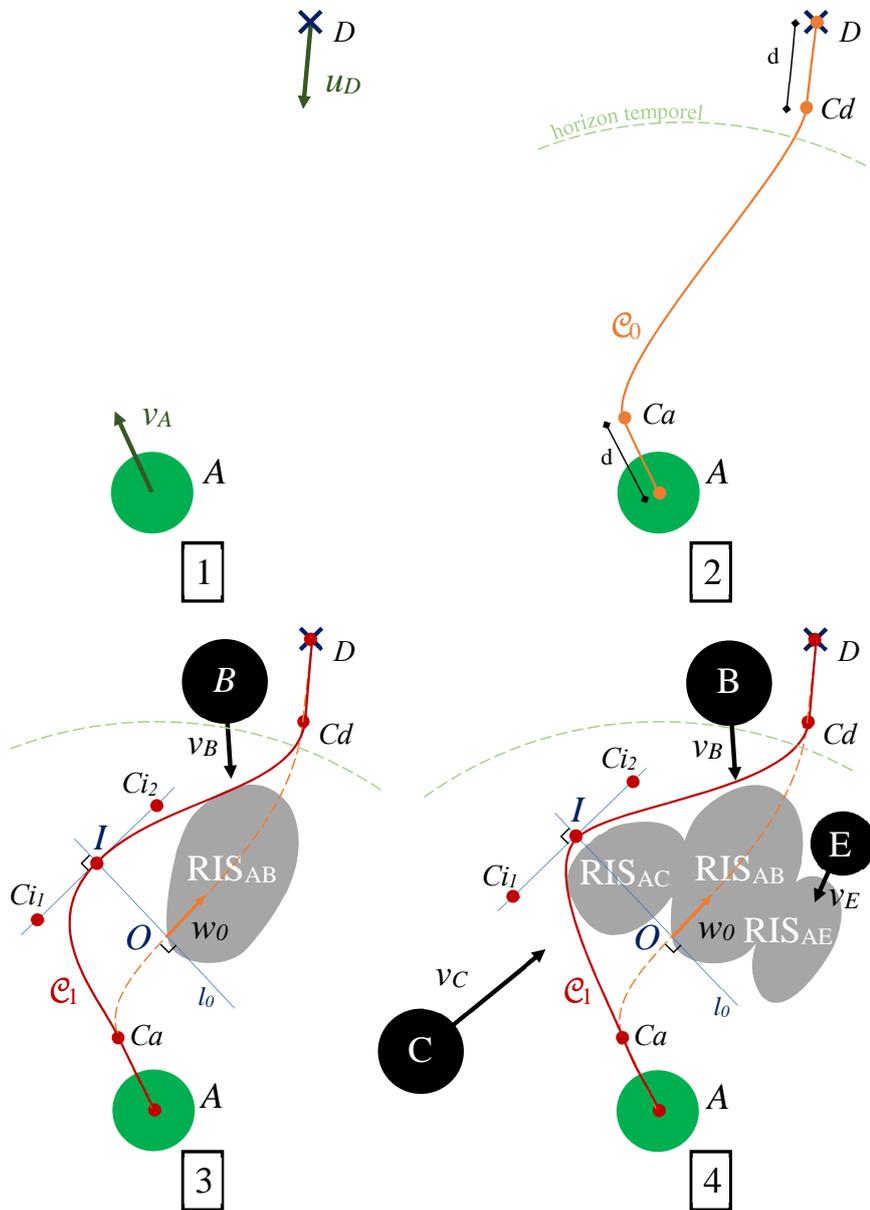


FIGURE 3.20 – Schéma de la construction du chemin *ris-bézier*. L'horizon temporel en pointillé représente la limite de la région atteignable par A durant la période $[0, t_H]$. Si RIS_A n'est pas vide, il est situé à l'intérieur de cette région. La destination intermédiaire D peut être située au-delà de cet horizon. (1) : les paramètres initiaux A, D, \vec{v}_A et \vec{u}_D . (2) : la courbe initiale \mathcal{C}_0 , utilisée comme chemin si RIS_A est vide ou ne l'intersecte pas. (3) et (4) : une nouvelle courbe \mathcal{C}_1 constituée de deux courbes de Bézier est utilisée lorsque \mathcal{C}_0 intersecte RIS_A . Elle est construite par itération en variant la position du point I le long de l_0 .

cision prise par l'approche *ris-bézier* est d'arrêter le robot A et d'attendre une situation plus propice. Lorsque le robot A est déjà à l'intérieur de RIS_A , le chemin \mathcal{C}_0 est utilisé par défaut afin de tenter d'extraire A de la région de danger, sachant que la décision de s'arrêter s'avérerait encore plus risquée. Ce raisonnement n'est valable que dans le cadre robotique où l'on ne distingue que de manière binaire une situation de collision : si l'objectif était d'atténuer la force d'une collision possible, la stratégie serait différente. Une fois le chemin de consigne retenu (\mathcal{C}_0 , \mathcal{C}_1 ou arrêt du véhicule), il est fourni au module de contrôle qui ajustera la vitesse de A en fonction.

3.4.1.3 Planification locale hybride

L'approche *ris-hybrid* combine les deux approches précédentes. A chaque instant t :

- Elle calcule la planification de chemin suivant la méthode *ris-bézier* et l'utilise lorsque la courbe \mathcal{C}_0 ne rencontre aucune région de RIS_A ou lorsque la courbe \mathcal{C}_1 est définie.
- Dans les cas où aucun chemin convenable ne peut être trouvé par *ris-bézier*, notamment lorsque le robot A est initialement situé dans une région de RIS_A , c'est l'approche *ris-apf* qui est utilisée afin d'obtenir une vitesse de consigne, à défaut d'obtenir une planification de chemin.

3.4.2 Stratégie globale de planification dynamique

Une planification globale de chemin permet d'adapter complètement la trajectoire du robot en fonction des mouvements des obstacles mobiles. Ce type d'approche nécessite plus de calcul, mais permet d'obtenir un chemin plus cohérent, sans discontinuité entre un chemin local et un global, ainsi que d'éviter que le robot ne soit bloqué dans une impasse formée par des obstacles mobiles. Nous présentons ici une nouvelle approche de planification globale qui sera utilisée pour trouver un chemin en contournant les régions RIS.

Afin d'obtenir un chemin optimal nous avons choisi de recourir à un graphe de visibilité, ce qui garantit de trouver le plus court chemin. L'horizon temporel pour le calcul des RIS permet de limiter la complexité du calcul du graphe, puisque seuls les RIS des obstacles passant à proximité du robot sont considérés. Les éventuels obstacles statiques sont également considérés afin de calculer un chemin global reliant le robot à sa destination.

Pour étendre cette approche à la planification de chemin pour un robot non-holonome et de forme polygonale quelconque, nous introduisons deux nouvelles contributions :

- Une approche de planification de chemin de Dubins qui utilise un graphe de visibilité,
- Un nouvel espace de configuration pour un robot polygonal qui tient compte de son orientation future.

3.4.2.1 Adaptation d'une planification globale en visibilité au modèle de Dubins

A notre connaissance, les approches de planification par graphe de visibilité établies dans la littérature sont restreintes au cas d'un robot holonome capable de suivre exactement le contour des obstacles en changeant instantanément la direction de sa vitesse. Cependant les robots non-holonomes, plus souvent utilisés en pratique, sont soumis à des contraintes d'accélération maximale et de vitesse angulaire limitée. Nous proposons ici une approche simple permettant d'adapter un chemin optimal pour robot holonome en un chemin de Dubins quasi-optimal pour un robot non-holonome soumis à une contrainte de vitesse angulaire limitée. Le rayon r de la courbure maximale pouvant être suivie par le robot (ou « rayon de braquage ») est le seul paramètre suffisant et nécessaire pour décrire les capacités motrices d'un tel robot. Cette méthode utilise le chemin optimal pour un robot holonome trouvé par une approche de visibilité, et l'adapte afin qu'il respecte les contraintes non holonomes d'un véhicule de Dubins.

Considérons la ligne brisée que constitue un chemin de visibilité calculé pour un robot holonome. Hormis ses deux extrémités, qui sont les points de départ et de destination, les points

du chemin sont tous par construction des sommets des polygones d'obstacles. Une portion de ce chemin est schématisée Fig. 3.21(a) où A, B et C sont des points du chemin. B est le sommet d'un obstacle et est relié à ses deux voisins par les deux segments de droite a et c . On peut se ramener à cette situation pour toute portion d'un chemin de visibilité. L'angle formé par les deux arêtes adjacentes en B est toujours saillant car on considère uniquement des obstacles convexes (le graphe de visibilité est construit à partir des enveloppes convexes des obstacles). Il est également possible qu'un des deux segments a ou c , voire les deux à la fois, soient des arêtes de l'obstacle.

Supposons qu'un robot, dont le rayon de braquage est limité à un rayon de courbure r et initialement positionné en A, parvienne au point B en ayant suivi le segment a . Ce robot non-holonyme ne peut suivre directement le segment c car cela nécessiterait un rayon de courbure égal à 0. Cependant il est possible d'établir un chemin de Dubins lui permettant d'atteindre le point C en combinant l'arc de cercle de rayon r de centre O_a puis le segment de droite c_a tangent à cet arc et passant par C, comme illustré sur la Fig. 3.21(b). Le point O_a est situé sur la perpendiculaire au segment a qui passe par B, à une distance r du côté de l'obstacle. Dans l'hypothèse où le robot a suivi le segment a , ce chemin de Dubins est optimal pour atteindre C.

Un autre chemin de Dubins passant par B est montré sur la Fig. 3.21(c) : il s'agit ici de conserver le segment c initial et de former un chemin de Dubins en combinant l'arc de cercle de centre O_c et de rayon r avec le segment a_c qui lui est tangent et passe par A.

Tous les chemins de Dubins qui utilisent le rayon de courbure minimal pour relier A à C en passant par le point B sont constitués d'un arc de cercle de rayon r ayant pour centre un point O_x et de deux segments tangents à cet arc passant respectivement par A et par C. L'ensemble des points O_x constitue l'arc de cercle de rayon r centré en B qui est compris entre les points O_c et O_a . Cet arc de cercle des points O_x est affiché en rouge sur la Fig. 3.21(d).

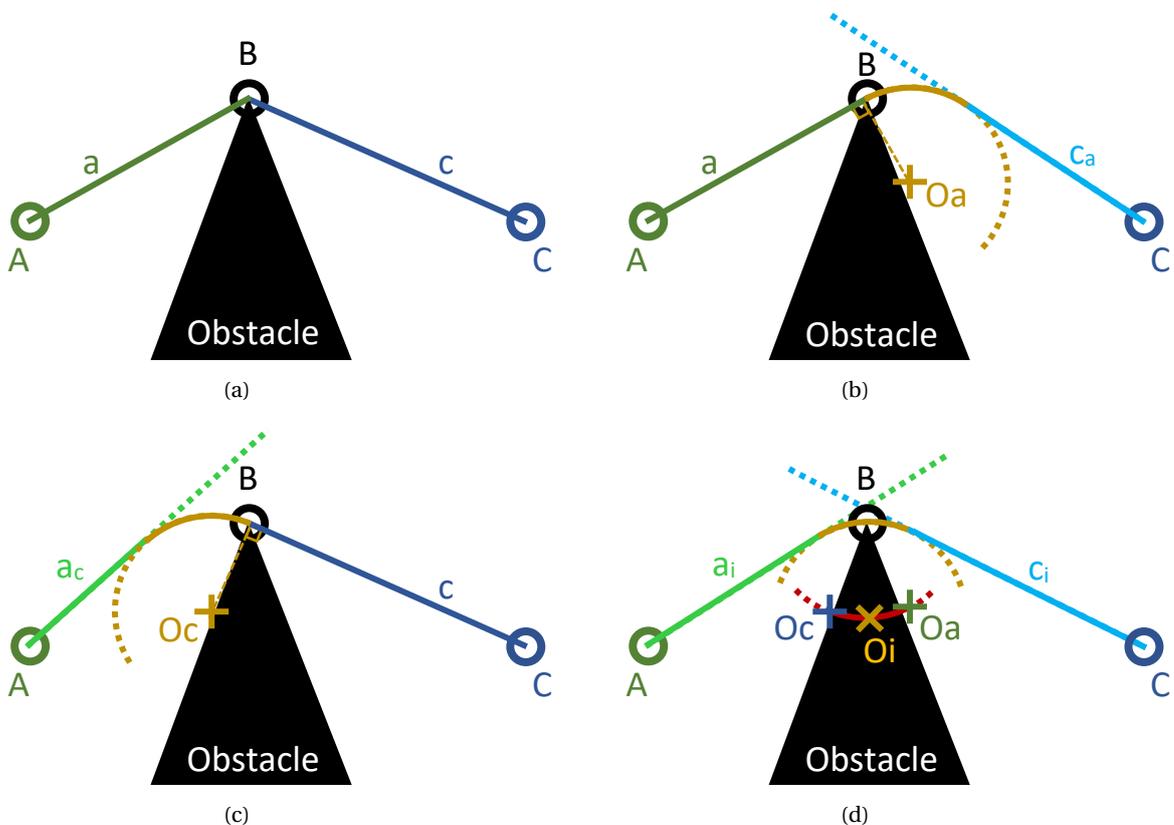


FIGURE 3.21 – Le chemin de visibilité autour d'un sommet d'obstacle B. Les segments a et c font partie du chemin global trouvé pour un robot holonyme contournant l'obstacle. Les segments c_a , a_c , a_i et c_i ont été créés tangentiellement aux arcs de cercle de centres O_a , O_c et O_i , respectivement.

En pratique, et afin de limiter la complexité du calcul, seules 3 hypothèses seront envisagées

pour placer le centre de l'arc de cercle : le point Oa , le point Oc et un point Oi qui est l'intersection de la bissectrice de l'angle \widehat{OcBOa} avec l'arc de cercle des points Ox .

L'obtention d'un chemin de Dubins complet entre la position de départ et la destination se fait en considérant ces trois hypothèses de cercles pour chaque point du chemin initial. Un graphe des combinaisons est construit afin de représenter ces différentes hypothèses de cercle. Pour chaque point intermédiaire du chemin initial (le point B dans l'exemple de la Fig. 3.21), trois nœuds sont ajoutés dans le graphe pour représenter les trois cercles centrés sur un point Ox . Deux nœuds sont également ajoutés pour les points de départ et de destination. Donc, si N est le nombre de points intermédiaires du chemin (en excluant les points de départ et de destination), il y a $3N + 2$ nœuds dans le graphe.

Les arêtes de ce graphe représentent les segments qui relient deux cercles associés à des points adjacents sur le chemin initial. Un exemple de graphe contenant $6 + 9(N - 1)$ arêtes est présenté sur la Fig. 3.22. Sur toutes les figures, les segments de couleur plus foncée appartiennent au chemin de visibilité initial. Ceux de couleur plus claire sont créés tangentiuellement à un ou deux arcs de cercle. On peut donc d'emblée remarquer sur la Fig. 3.22 que les segments du chemin de visibilité sont présents dans le graphe, mais ne pourront jamais constituer à eux seuls un chemin de Dubins valide entre les points de départ et de destination.

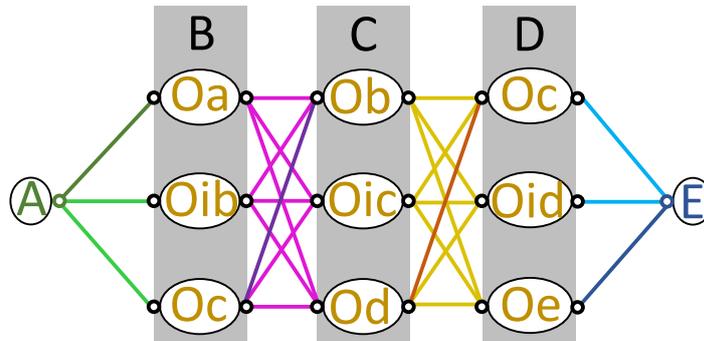


FIGURE 3.22 – Exemple d'un graphe des combinaisons possibles d'arcs de cercle.

Une fois ces éléments définis, l'approche de recherche d'un chemin de Dubins complet consiste à tester la validité de chaque arête du graphe. Une arête est invalide lorsque aucun segment de droite ne peut être placé entre deux arcs de cercle (cf. Fig. 3.23(b)) ou lorsqu'une portion d'arc de cercle ou de segment entre en collision avec un obstacle (cf. Fig. 3.23(c)). Un exemple de chemin complet valide est donné sur la Fig. 3.23(d).

Enfin, le plus court chemin de Dubins est calculé en appliquant l'algorithme de Dijkstra sur le graphe des combinaisons, en pondérant chaque arête par les longueurs des segments et arcs de cercle qui y sont associés.

Limites de la planification globale d'un chemin de Dubins

Des exemples de chemins de Dubins complets sont présentés sur la Fig. 3.24. L'approche n'a pas pu trouver de chemin valide pour deux des destinations, indiquées par des X rouges. Ces positions sont pourtant atteignables en absolu par les capacités motrices du robot, mais il faudrait pour cela ajouter dans le graphe des arcs de cercle supplémentaires qui n'ont pas de rapport avec le chemin initial calculé pour un robot holonome. Cette situation peut être observée sur la Fig. 3.25(a), où l'approche présentée ici échoue à trouver un chemin de Dubins valide. Une solution simple pourrait être d'ajouter un arc de cercle contenant la destination qui permet au robot d'effectuer un demi-tour. Cependant, nous n'avons pas introduit ce type de solution dans l'approche car elle ne serait que partielle. En effet, on peut trouver des environnements d'une complexité arbitraire dans lesquels une planification de chemin complexe est nécessaire pour permettre au robot d'effectuer un demi-tour, comme le montre l'exemple de la Fig. 3.25(b). Nous limiterons donc notre approche de planification non-holonome aux cas de figure où les points du chemin initial sont

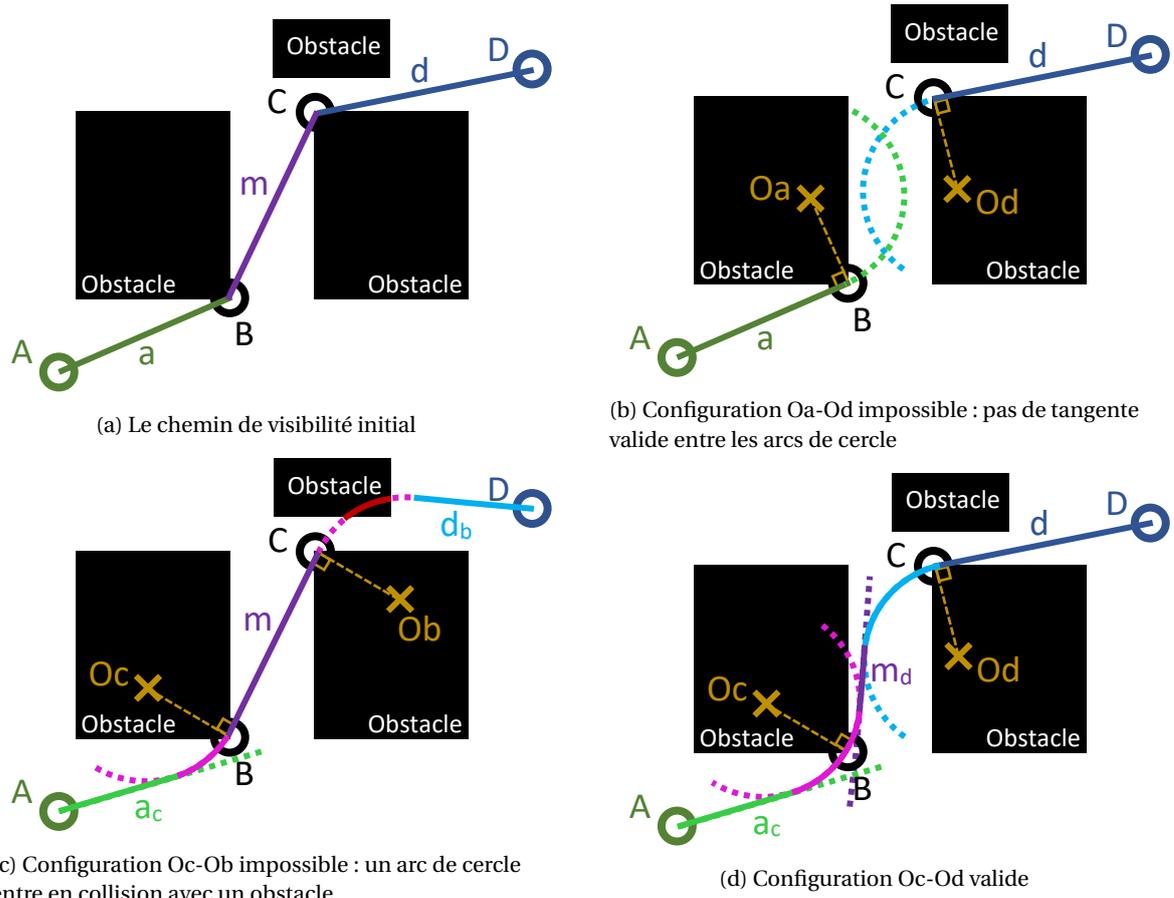


FIGURE 3.23 – Construction du chemin de Dubins complet. Certaines arêtes du graphe des combinaisons sont impossibles.

suffisants pour trouver un chemin de Dubins. En pratique, cette solution convient pour les cas de figure qui nous intéressent, dans lesquels l'horizon temporel du calcul des RIS évite la présence d'obstacles à proximité de la destination.

Il existe également trois autres limitations à cette approche :

- Le chemin trouvé peut ne pas être le plus court possible car seules trois hypothèses de cercles sont envisagées. Afin d'obtenir le chemin de Dubins optimal il faudra pouvoir étendre le graphe des combinaisons à tous les centres de cercles O_i possibles.
- Parfois, le chemin initial ne peut absolument pas être adapté en un chemin de Dubins. C'est le cas quand il passe par un couloir étroit formant un virage serré, par exemple. Il est alors nécessaire de « condamner » ce passage étroit et de relancer la recherche de chemin en mettant à jour le graphe de visibilité, afin de tenter de trouver un autre passage. Au-delà du temps de calcul plus élevé, le problème dans cette configuration est qu'il peut s'avérer difficile de modifier le graphe de visibilité pour « condamner » uniquement le passage trop étroit.
- Le chemin de Dubins obtenu ne peut être suivi que par un véhicule théorique capable de modifier instantanément la courbure de son chemin. Afin de permettre à un véhicule plus réaliste de suivre le chemin à vitesse constante, il serait cependant possible d'ajouter des clothoïdes entre les arcs de cercle et les segments de droite afin de garantir la continuité de la courbure du chemin (cf. FRAICHARD et SCHEUER [2004]).

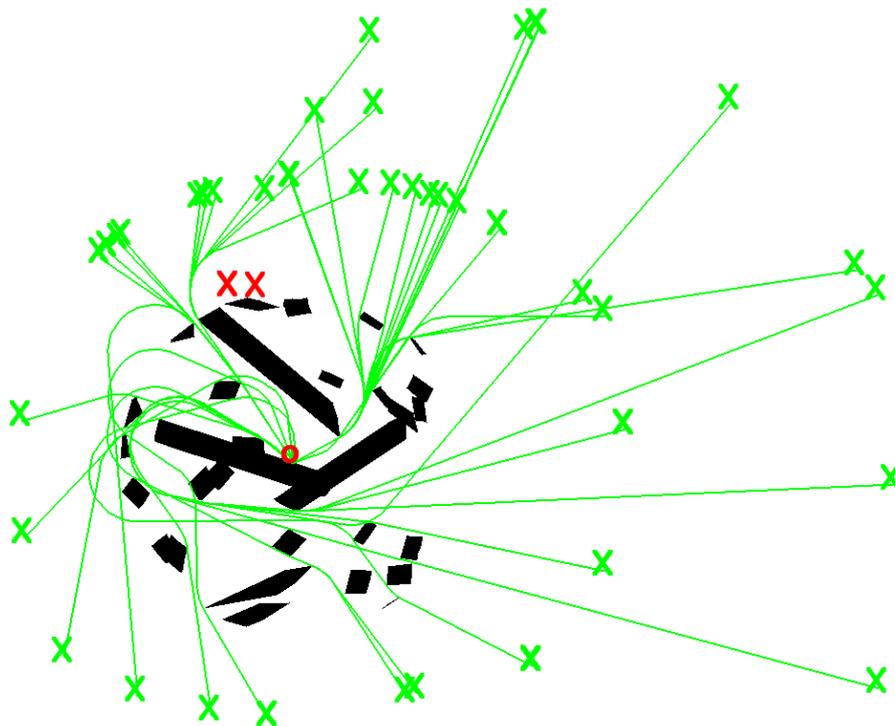
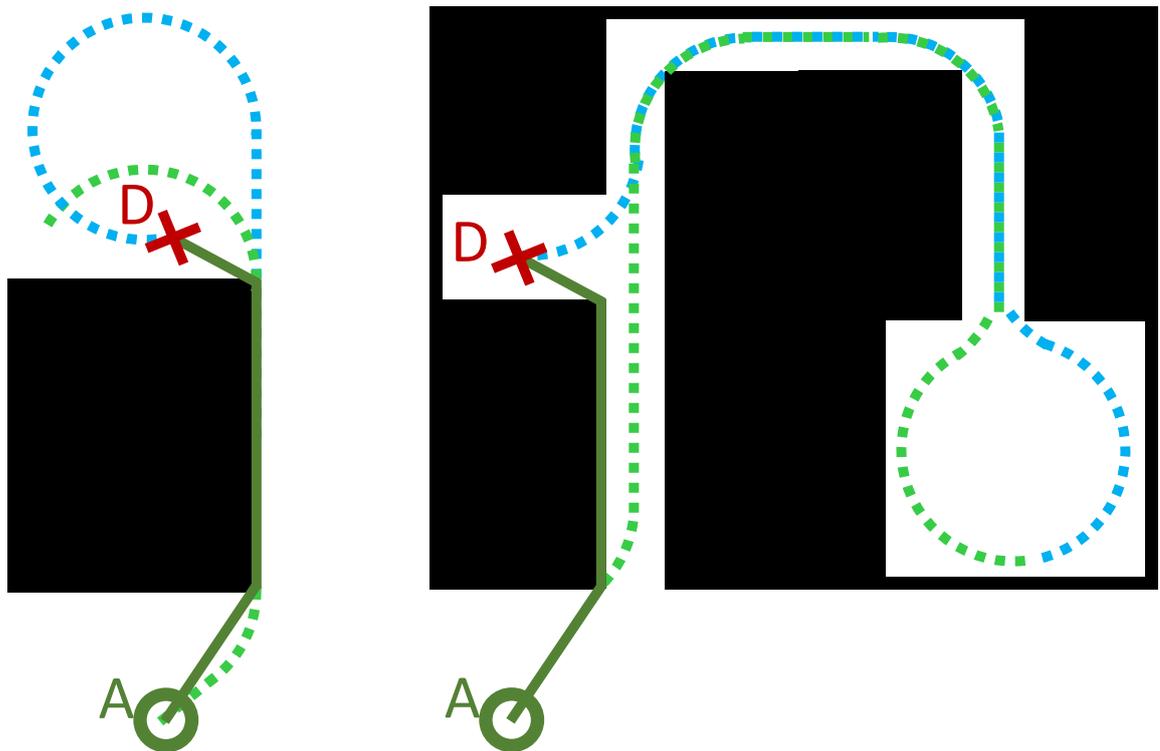


FIGURE 3.24 – Exemples de chemins de Dubins complets. Les obstacles sont en noir, le point de départ est indiqué par le O rouge, les chemins valides sont en vert et les destinations inatteignables sont marquées par des X rouges.



(a) Cas d'échec simple : le chemin de Dubins valide (cyan) ne peut pas être trouvé par l'approche présentée, qui échoue (vert clair).

(b) Cas d'échec complexe : trouver un chemin de Dubins valide nécessite de trouver un endroit pour effectuer un demi-tour (en vert clair puis cyan), ce qui peut nécessiter une recherche de chemin de difficulté arbitraire.

FIGURE 3.25 – Cas de figure d'échec de l'approche, où le chemin initial (vert foncé) n'est pas une base suffisante pour calculer le chemin de Dubins car le robot doit s'en éloigner pour parvenir à effectuer un demi-tour.

3.4.2.2 Nouvel espace de configuration

Deux principaux espaces de configuration sont utilisés en recherche de chemin. Ils permettent de remplacer un problème de planification de chemin d'un robot polygonal se déplaçant parmi des obstacles par celui du calcul du chemin d'un robot ponctuel parmi des obstacles dont les contours sont modifiés. Ces deux espaces de configuration sont présentés dans les Fig. 3.26(b) et 3.26(c) pour un même scénario d'exemple. Ils sont tous deux calculés par application d'une somme de Minkowski entre les polygones des obstacles et celui du robot. Pour le premier espace de configuration, l'orientation du robot est supposée constante comme si le robot se déplaçait uniquement en translation. Pour le second, toutes les rotations possibles du robot sont prises en compte, ce qui est une hypothèse plus sûre mais aussi parfois trop conservatrice. En effet, dans la Fig. 3.26(c) l'espace de configuration obtenu empêche le robot de passer entre les deux obstacles de droite (l'étoile et le rectangle).

On remarque que la somme de Minkowski sur le polygone du robot à orientation constante peut être construite géométriquement comme suit :

1. Pour chaque sommet S_o d'un obstacle, on place autant de polygones orientés du robot que de sommets S_r du polygone du robot, en faisant coïncider les points S_o et S_r ,
2. Parmi les polygones ainsi placés on ignore ceux qui intersectent l'obstacle,
3. En reliant les centres des polygones restants en tournant autour de l'obstacle, on obtient le polygone résultant de la somme de Minkowski.

Les polygones orientés en question sont affichés en vert clair sur la Fig. 3.26(b).

Nous proposons une nouvelle approche d'espace de configuration en partant de cette interprétation géométrique et en remplaçant, pour chaque sommet d'obstacle S_o , l'orientation fixe du robot par la direction du vecteur \vec{AS}_o , où A est la position du robot. Cet espace de configuration est illustré dans la Fig. 3.26(d), dans le même scénario que pour les deux autres. Cette approche tient compte du fait que le robot choisit d'abord son chemin puis change d'orientation en conséquence, ce qui permet de prendre une meilleure décision en tenant compte en amont de ce changement d'orientation. On remarque que cette approche est plus sûre que l'espace de configuration à orientation constante, dont l'hypothèse s'avère fautive lorsque le robot est non-holonyme et n'avance pas tout droit, et moins conservatrice que l'espace de configuration avec des disques, ce qui se remarque par les larges couloirs navigables laissés entre les polygones de l'espace de configuration. La conséquence de ce nouvel espace de configuration sur le volume 3D d'un obstacle construit dans l'espace-temps (X, Y, t) peut être observée sur la Fig. 3.27.

3.4.2.3 Application de la planification globale aux RIS

La méthode de planification complète de chemin nommée *ris-path* combine l'approche de calcul d'un chemin de Dubins par un graphe de visibilité et l'espace de configuration qui ont été présentés précédemment. Cette planification est utilisée pour trouver un chemin entre la position du robot en tenant compte de son orientation et la destination globale, en évitant les polygones des RIS. Comme dans les approches de planification précédemment décrites, les polygones des obstacles réels ne sont pas considérés.

Dans certaines situations, il est possible que le robot se retrouve à l'intérieur des RIS. Cela peut notamment être le cas lorsque :

- l'hypothèse formulée pour le mouvement du robot A s'avère invalide (cf. la définition de la surface \mathcal{S}_d). En effet, si le chemin global calculé pour éviter les obstacles est trop différent de cette hypothèse, le calcul des RIS peut ne plus être valide à cause du déplacement du robot. Cependant, l'aspect dynamique de la planification de chemin corrige en permanence ces éventuelles erreurs.
- la précision de la prédiction de mouvement des obstacles est insuffisante
- la trajectoire planifiée n'a pas été parfaitement respectée par le module de contrôle du robot

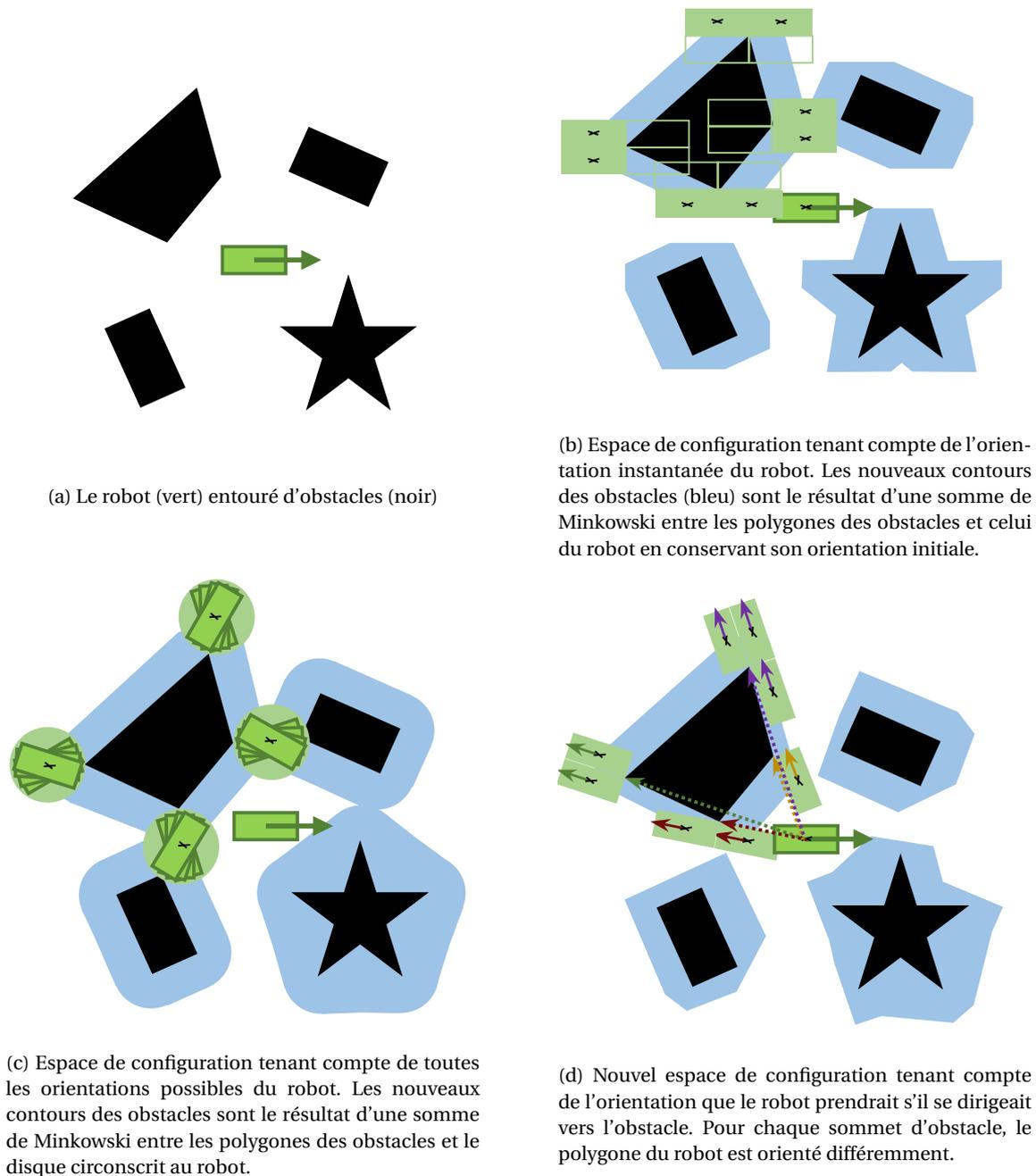


FIGURE 3.26 – Les différents espaces de configuration

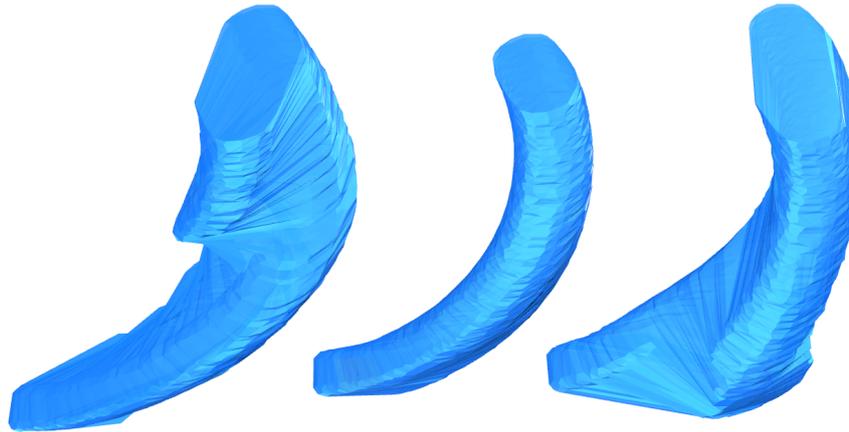


FIGURE 3.27 – Volume 3D d'un obstacle avec le nouvel espace de configuration. Les trois mêmes trajectoires d'un obstacle sont présentées. Le volume du milieu est généré en considérant l'obstacle réel. Les deux autres sont générés en utilisant le nouvel espace de configuration, pour deux orientations différentes du robot.

Afin d'éviter que l'entrée du robot à l'intérieur des RIS présente immédiatement un risque trop élevé, une marge de sécurité est ajoutée par application d'une somme de Minkowski d'un disque sur les polygones des RIS.

Dans les cas où le robot pénètre à l'intérieur d'un polygone de RIS, la priorité est d'éviter une collision et l'algorithme le fait sortir au plus vite de la zone. Pour cela, la planification de chemin complète est remplacée par une approche simple avec 7 tentacules de courbure constante et de vitesse maximale. Les rayons de courbure de ces 7 trajectoires d'urgence ont pour valeurs $\{-3 \cdot r, -2 \cdot r, -r, 0, r, 2 \cdot r, 3 \cdot r\}$, r étant le rayon de courbure minimal que peut suivre le robot. Sur la Fig. 3.28, on peut observer ces tentacules ainsi que le déroulement de l'algorithme qui s'effectue ainsi : si le robot est dans la marge de sécurité, il faut trouver une trajectoire de sortie à condition qu'elle n'intersecte pas le polygone des RIS calculé sans marge, car le risque de collision y est encore plus élevé. Si le robot se retrouve à l'intérieur de ce polygone sans marge, la stratégie se contente de trouver une trajectoire qui en sort au plus vite.



(a) Itération sur 5 tentacules lorsque le robot est dans un RIS calculé avec marge

(b) Itération sur 5 tentacules lorsque le robot est dans un RIS calculé sans marge, puis seconde itération pour sortir du RIS avec marge

FIGURE 3.28 – Trajectoires d'urgence pour sortir le robot d'un RIS. Les trajectoires retenues sont terminées par un disque de couleur.

3.5 Validation des approches de planification

3.5.1 Validation en présence d'obstacles au mouvements déterministes

Nous avons comparé sur des scénarios identiques nos 3 approches issues du calcul des RIS (*ris-apf*, *ris-bézier* et *ris-hybrid*) aux 3 approches suivantes de planification de mouvement :

- **static-apf** : une méthode de champ de potentiel classique qui considère que tous les obstacles sont statiques (cf. [LATOMBE \[1991\]](#))
- **dynamic-apf** : une méthode de champ de potentiel adaptée aux environnements dynamiques qui tient compte des vitesses des obstacles (cf. [GE et CUI \[2002\]](#))
- **vo** : l'approche V-obstacles (cf. l'algorithme introduit dans ? et dont le code source est disponible via [VAN DEN BERG et collab. \[2016\]](#))

Nous avons implémenté les deux algorithmes de champ de potentiel en langage python à partir des spécifications décrites dans la littérature associée. Pour l'approche V-obstacles le code C++ fourni en ligne pour l'algorithme de planification réciproque *ORCA* a été utilisé. Nous l'avons modifié afin de supprimer l'aspect collaboratif de la planification de mouvement et ne garder que son évitement d'obstacles mobiles. Les régions RIS sont obtenues par une intersection de maillages 3D entre l'occupation de l'espace-temps par les trajectoires des obstacles mobiles et celle de la trajectoire prédite du robot ego. Ce calcul d'intersection est effectué par la bibliothèque logicielle *PyMesh*. La projection sur le plan $t = 0$ ainsi que les planifications de trajectoire à partir des RIS sont implémentées en python.

3.5.1.1 Les scénarios de test

Afin de pouvoir comparer ces 6 approches, des simulations ont été effectuées dans 4 environnements différents :

- **free** : une surface de 800x800 sans obstacles statiques dans laquelle circulent des obstacles aux trajectoires rectilignes ou circulaires. Le robot A démarre à la position (50, 750) et doit atteindre la destination située à (750, 50). Cet environnement est représenté Fig. 3.29(a).
- **free-straight** : une surface de 800x800 sans obstacles statiques dans laquelle circulent des obstacles aux trajectoires rectilignes. Le robot A démarre à la position (50, 750) et doit atteindre la destination située à (750, 50). Cet environnement est représenté Fig. 3.29(b).
- **door** : une surface de 800x800 contenant deux obstacles statiques rectangulaires formant une porte que le robot doit traverser ainsi que des obstacles mobiles dont les trajectoires peuvent être rectilignes ou circulaires. A démarre à la position (200, 700) et doit atteindre la destination (600, 700). Cet environnement est représenté Fig. 3.29(c).
- **cross** : une surface de 1600x800 sans obstacles statiques sur laquelle circulent deux flux de véhicules en sens contraire à la manière d'une route à deux fois deux voies. Le robot A démarre à la position (800, 750) et doit traverser les quatre voies afin de parvenir sur la droite (x, 50). Cet environnement est représenté Fig. 3.29(d).

Les mesures de distance, données sans unité, ont été implémentées en pratique en nombre de pixels. Le robot ego est en forme de disque de rayon 30 unités pouvant se déplacer à une vitesse maximale v_{Amax} de 4 unités par pas de temps. Pour chaque environnement, les expérimentations sont effectuées en variant le nombre d'obstacles mobiles et les trajectoires qu'ils suivent. Les vitesses des obstacles sont tirées aléatoirement dans l'intervalle $[0, v_{Amax}]$ pour les variantes *slower* et dans l'intervalle $[v_{Amax}, 2 \cdot v_{Amax}]$ pour les variantes *faster*. Les obstacles peuvent se superposer entre eux et leur trajectoire n'est pas influencée par la présence du robot ou des autres obstacles. Lorsqu'un obstacle atteint un des bords dans l'environnement *cross*, il est déplacé instantanément de l'autre côté de l'environnement de façon à ce que les flux d'obstacles soient continus. Dans les trois autres environnements, la trajectoire d'un obstacle atteignant un bord est comme

réfléchi (une des composantes de vitesse de l'obstacle change de signe). Ainsi, la densité d'obstacles reste constante pour tous les environnements, et leurs trajectoires sont complètement déterminées par leur état initial. Pour chaque environnement et pour chaque variante en matière de densité d'obstacles et de vitesses, 100 états initiaux sont générés aléatoirement comme suit :

- La forme de chaque obstacle est choisie aléatoirement, soit un disque dont le rayon est dans $[10, 60]$ soit un rectangle d'une longueur tirée dans $[10, 60]$ et moitié moins large.
- Sa position initiale est tirée aléatoirement sur la surface de l'environnement à l'exception d'une zone qui entoure la position initiale de A. Dans l'environnement *cross*, les obstacles sont placés aléatoirement sur une des 4 voies.
- Sa vitesse est tirée aléatoirement en fonction de la variante dans les intervalles de vitesses *slower* ou *faster*. Dans *cross*, ces 2 variantes n'existent pas et les vitesses sont tirées dans l'intervalle $[0, 2 \cdot v_A \max]$.
- Dans les environnements *free* et *door*, un angle de braquage est attribué à chaque obstacle afin de déterminer sa trajectoire circulaire. Celui-ci est tiré aléatoirement dans $[-\frac{\pi}{80}, \frac{\pi}{80}]$, en radians par pas de temps.

3.5.1.2 Résultats

Les différentes approches seront principalement évaluées en comparant leur taux de succès, qui est égal, en considérant les 100 scénarios de chaque variante, au nombre de scénarios pour lesquels A réussit à atteindre la destination, divisé par 100.

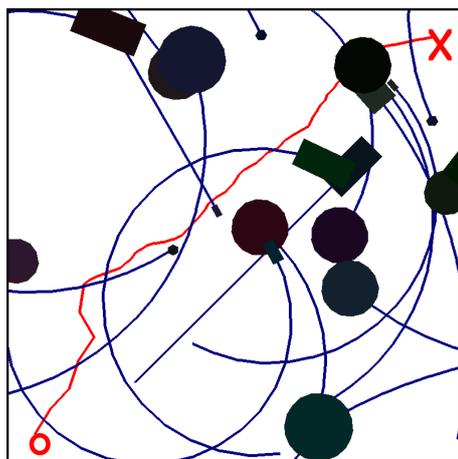
Les graphes de la Fig. 3.30 présentent les taux de succès d'un robot en présence d'obstacles moins rapides que lui (vitesses dans $[0, v_A \max]$), dans les environnements *free* (a), *free-straight* (b) et *door* (c). De même, ces taux de succès sont présentés Fig. 3.31 pour un robot en présence d'obstacles plus rapides que lui (vitesses dans $[v_A \max, 2 \cdot v_A \max]$). Nous pouvons remarquer que :

- Dans tous les environnements, la présence d'obstacles plus rapides fait baisser très nettement les performances de toutes les approches par rapport aux mêmes situations avec des obstacles plus lents. Cependant on peut remarquer que les performances de planification utilisant les RIS sont bien moins impactées par la vitesse des obstacles mobiles.
- Dans tous les scénarios comportant des obstacles plus rapides que le robot, l'approche *ris-hybrid* obtient le meilleur taux de succès et une des deux autres approches basées sur les RIS se positionne toujours en deuxième place.

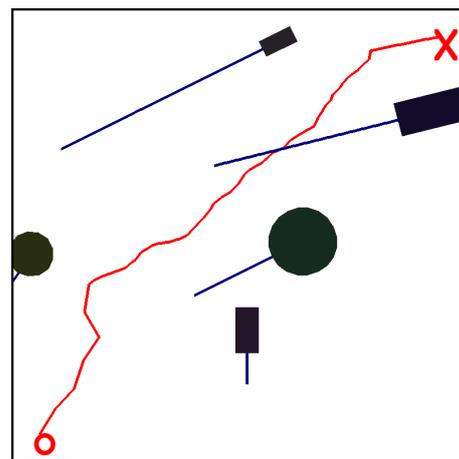
L'algorithme de planification *ris-hybrid* bénéficie donc du double avantage conféré par le chemin efficace de *ris-bézier*, qui lui permet de rester le moins longtemps possible dans l'environnement, et la réactivité de *ris-apf*, dont le champ de potentiel est plus adapté pour contourner un danger imminent. L'approche *ris-bézier* est par ailleurs loin d'être optimisée, puisque le robot choisit la solution simple de s'arrêter lorsque aucune courbe \mathcal{C}_1 ne peut être trouvée.

Les performances obtenues par les approches dans l'environnement *cross* sont présentées sur la Fig. 3.32. Les approches utilisant les RIS s'avèrent être également les plus efficaces pour traverser les flux de véhicules sans collision dans l'environnement *cross*. Dans ces scénarios, les approches *ris-bézier* et *ris-hybrid* ont des performances quasiment identiques et sont bien plus robustes que *ris-apf*. Cela démontre l'avantage de la planification « à moyen terme » qu'apportent les courbes de Bézier par rapport à une approche réactive qui échoue souvent en tentant de s'insérer dans une impasse.

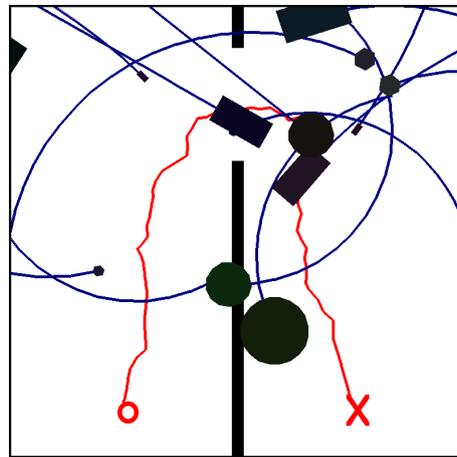
Les graphes des Fig. 3.33, 3.34 et 3.35 montrent les longueurs des chemins parcourus dans les mêmes situations. Seuls les cas de succès ont été utilisés pour calculer la moyenne des longueurs de chemin. Ces longueurs sont normalisées par la longueur du plus court chemin reliant le point de départ à la destination et évitant les obstacles statiques. On remarque que l'approche *ris-bézier* surpasse toutes les autres sur ce critère, en étant capable de trouver un chemin dont la longueur est très proche de l'optimum.



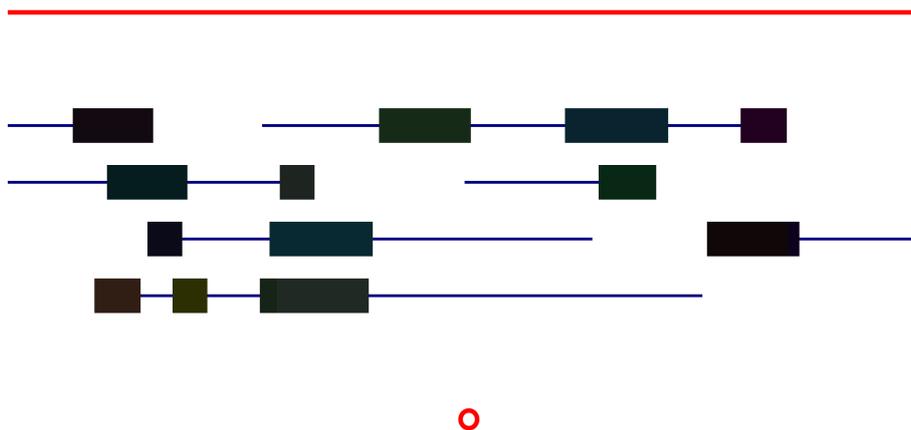
(a) Environnement *free* avec 20 obstacles mobiles



(b) Environnement *free-straight* avec 5 obstacles mobiles



(c) Environnement *door* avec 15 obstacles mobiles



(d) Environnement *cross* avec 15 obstacles mobiles

FIGURE 3.29 – Les quatre environnements de validation. Les trajectoires futures des obstacles sont indiquées en bleu. Les points de départ et de destination sont respectivement marqués par un O et un X rouge. Le chemin global évitant les obstacles statiques est calculé par un arbre d'exploration RRT et est tracé en rouge. L'environnement *cross* est plus spécifique : tout point sur la ligne horizontale rouge convient comme destination. Comme il n'y a pas d'obstacle statique à éviter dans cet environnement, aucun chemin global n'est fixé et le projeté du robot sur la ligne de destination est donné à chaque pas de temps comme destination intermédiaire.

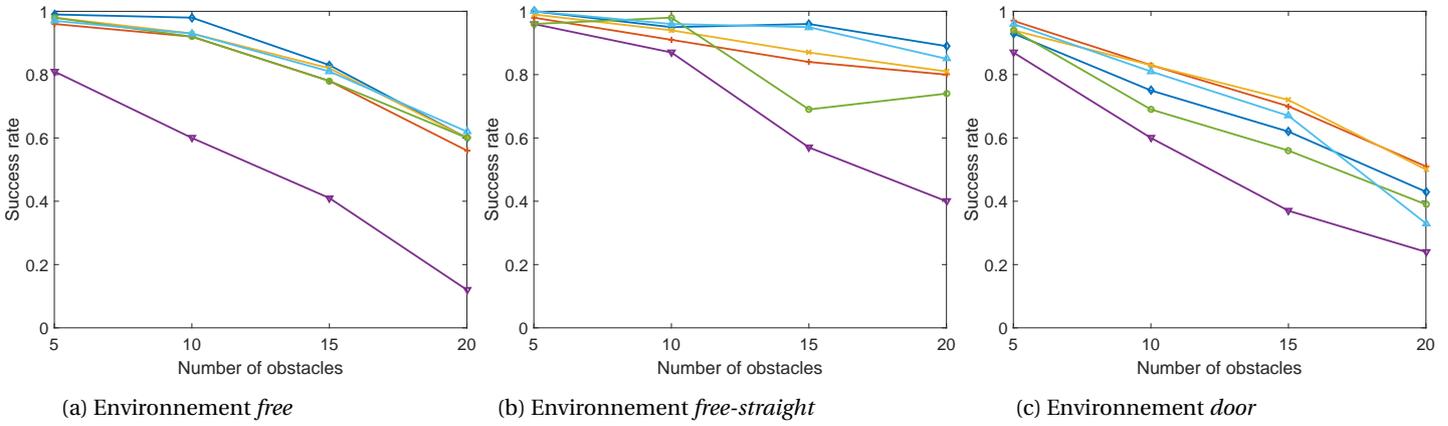


FIGURE 3.30 – Taux de succès avec des obstacles plus lents que le robot

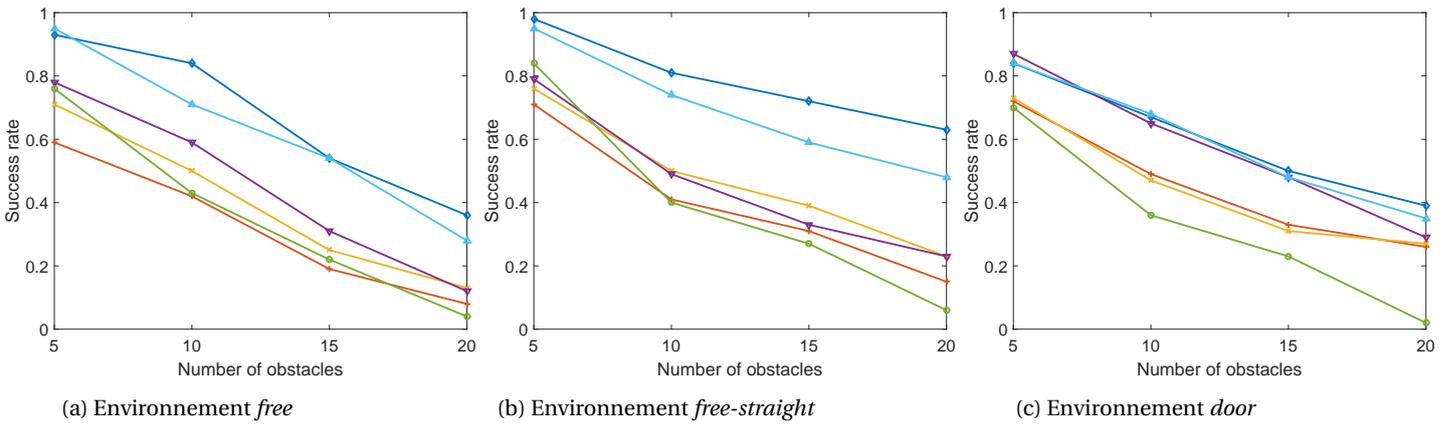


FIGURE 3.31 – Taux de succès avec des obstacles plus rapides que le robot

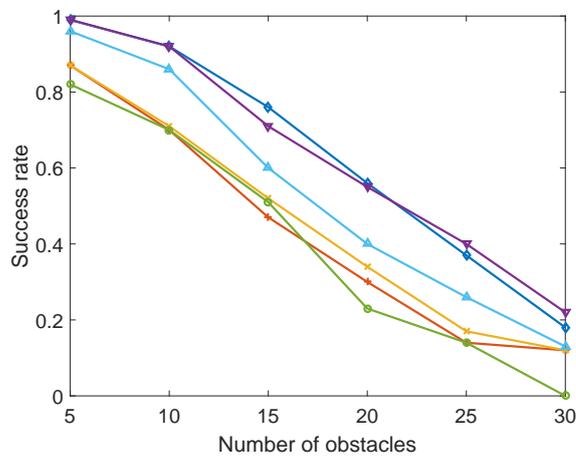


FIGURE 3.32 – Taux de succès pour l'environnement *cross*

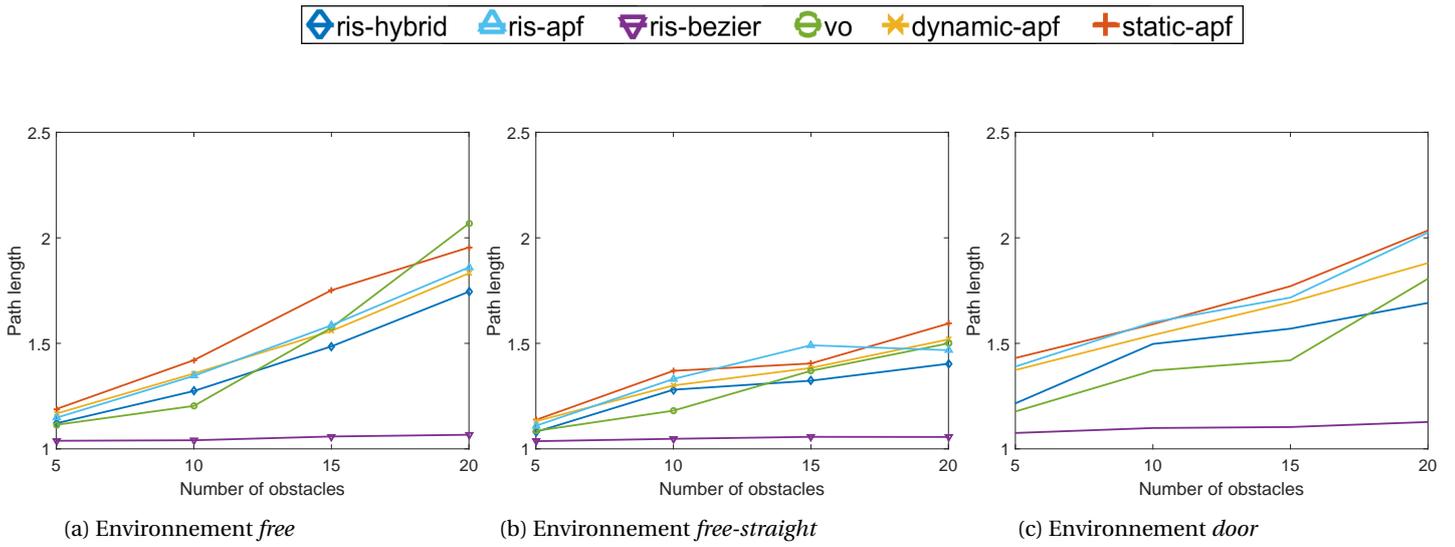


FIGURE 3.33 – Longueur du chemin avec des obstacles plus lents que le robot

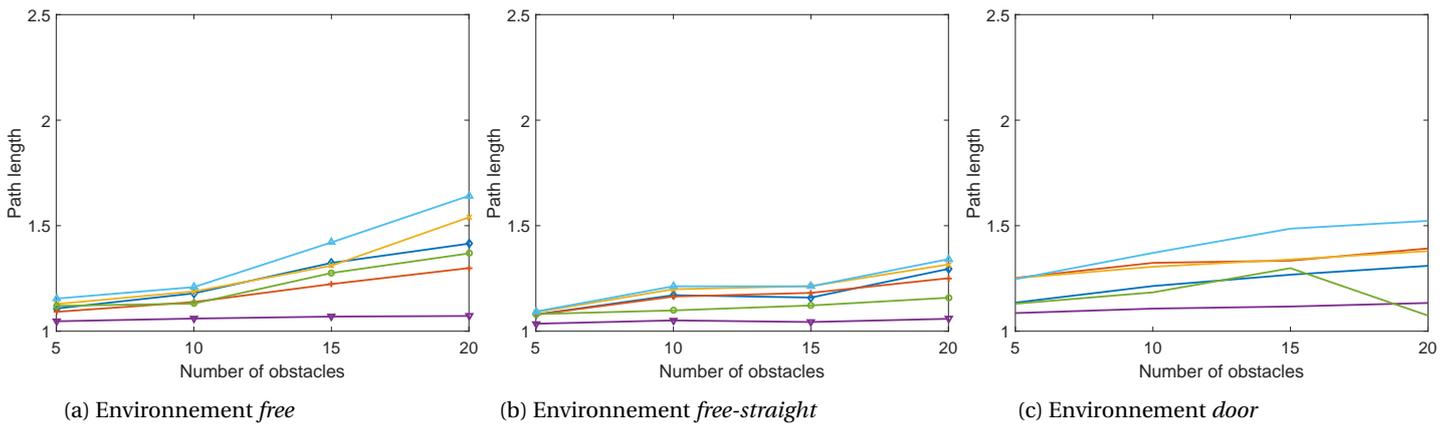


FIGURE 3.34 – Longueur du chemin avec des obstacles plus rapides que le robot

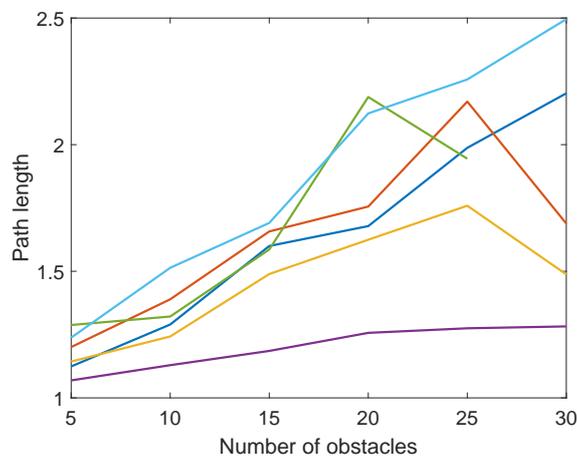


FIGURE 3.35 – Longueur du chemin pour l'environnement cross

La séquence d'images de la Fig. 3.36 permet d'observer le comportement de l'approche *ris-hybrid* dans l'environnement *door*. Les polygones des RIS (en gris clair) sont affichés, ainsi que les éléments géométriques qui permettent de définir le chemin calculé par l'approche.

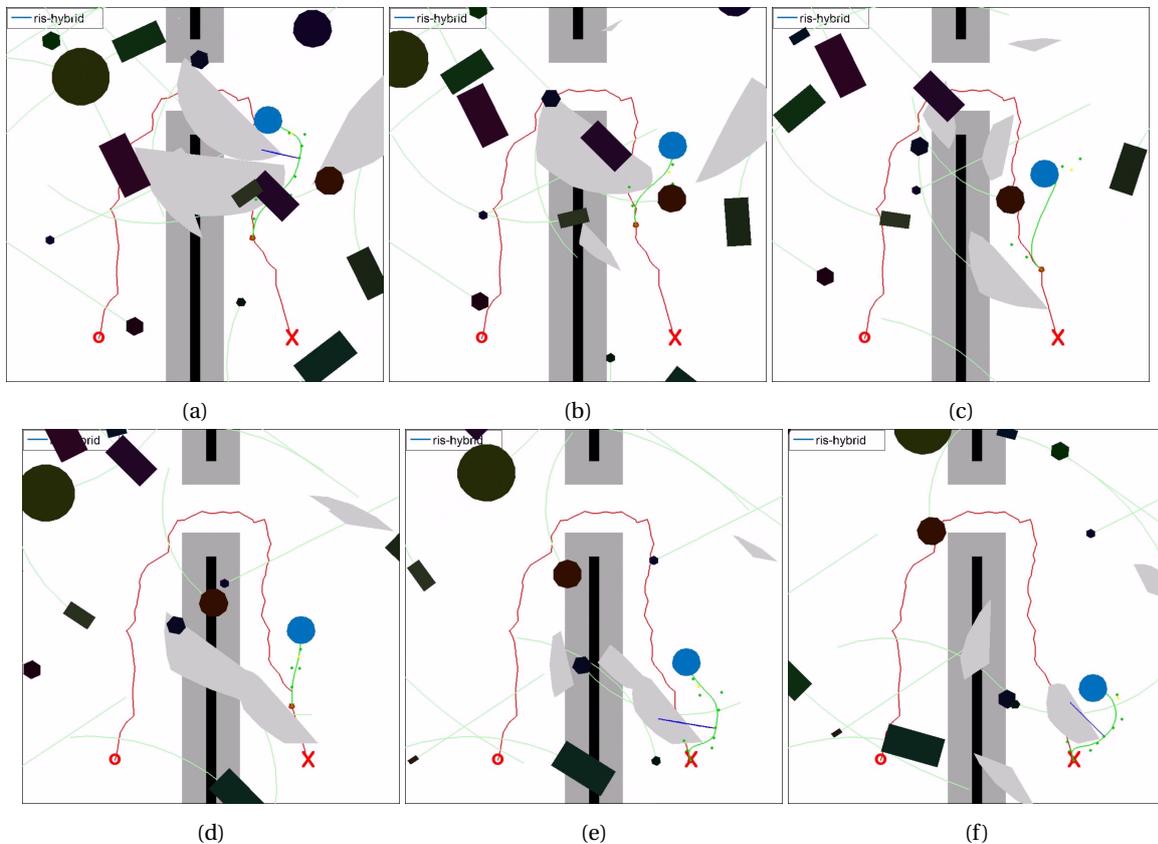


FIGURE 3.36 – Aperçu des polygones de RIS (en gris clair) durant la planification de chemin *ris-hybrid* dans l'environnement *door*. Le disque bleu représente le robot, et son chemin est tracé en vert. On peut observer les points de contrôle en vert ainsi que, sur les dernières images, la droite l_0 en bleu foncé (ces éléments sont introduits dans la section 3.4.1.2).

Comparons le comportement des algorithmes de planification sur trois scénarios de l'environnement *free-straight* présentés dans les Fig. 3.37, 3.38 et 3.39. Les obstacles sont en noir et plusieurs robots ego sont représentés avec différentes couleurs, en fonction de l'approche de planification suivie. Les obstacles mobiles ainsi que les robots sont en forme de disque et une trace de leur trajectoire passée est laissée visible. Cette trace est composée de 10 pas de temps régulièrement espacés durant l'horizon temporel de 50. On peut ainsi remarquer que, dans ces trois scénarios, les obstacles sont plus rapides que les robots.

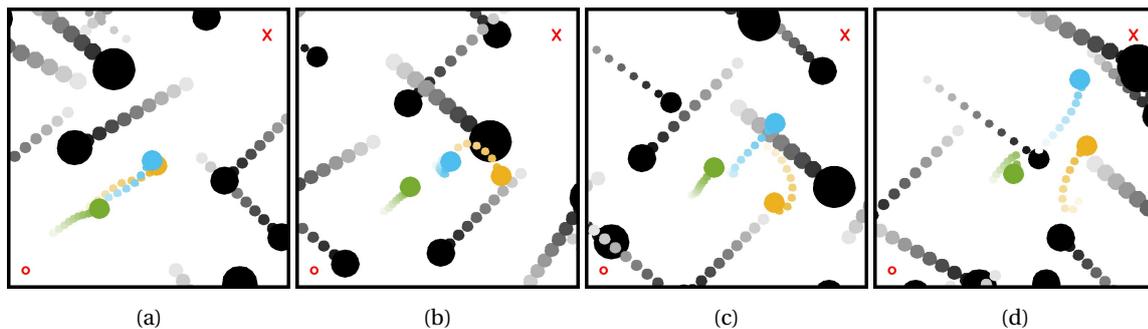


FIGURE 3.37 – Séquence comparant les planifications des approches *ris-apf* (cyan), *dynamic-apf* (jaune) et *vo* (vert) en présence de 10 obstacles plus rapides (noir).

- La Fig. 3.37 présente un scénario où toutes les approches réussissent à atteindre la destination sans entrer en collision avec les obstacles. Le robot *vo* est plus lent que les autres car cette approche n'utilise la vitesse maximale qu'aux abords d'un obstacle. À l'étape (b), les approches *ris-apf* et *dynamic-apf* qui agissaient alors de façon similaire prennent une décision différente : *ris-apf* ralentit en vue de laisser passer un obstacle qui arrive rapidement depuis la gauche, tandis que *dynamic-apf* continue d'avancer et se retrouve dans le passage de l'obstacle. Le robot suivant le champ de potentiel *dynamic-apf* se retrouve alors « poussé » vers la droite en tentant d'esquiver l'obstacle, ce qui finit par lui faire prendre du retard par rapport à *ris-apf*. On peut noter l'avantage des RIS, qui confèrent à une stratégie de champ de potentiel statique une meilleure anticipation des mouvements que ne le fait un champ de potentiel qui tient compte des vitesses instantanées des obstacles.

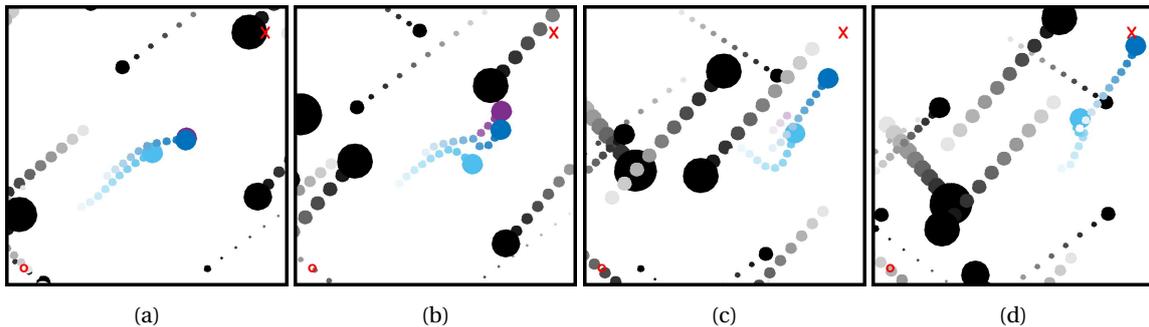


FIGURE 3.38 – Séquence comparant les planifications des approches *ris-apf* (cyan), *ris-bézier* (violet) et *ris-hybrid* (bleu) en présence de 10 obstacles plus rapides (noir).

- Sur la Fig. 3.38 les trois approches RIS sont comparées et l'approche *ris-hybrid* s'avère la plus robuste et la plus rapide. À l'étape (b), *ris-apf* anticipe le mouvement d'un obstacle mais prend du retard par rapport aux autres approches qui ont trouvé un chemin plus efficace pour le contourner. Remarquons également que *ris-hybrid* a dû brusquement recourir au champ de potentiel car sa trajectoire effectue un virage abrupte. La planification *ris-bézier* n'adoptant pas la même stratégie, le robot associé se retrouve à l'intérieur d'une région RIS. Il tente alors de continuer sa course en suivant la courbe \mathcal{C}_0 , mais finit par entrer en collision avant l'étape (c). Enfin, l'étape (d) montre l'avance acquise par l'approche *ris-hybrid* par rapport à *ris-apf* lorsqu'elle parvient à la destination.

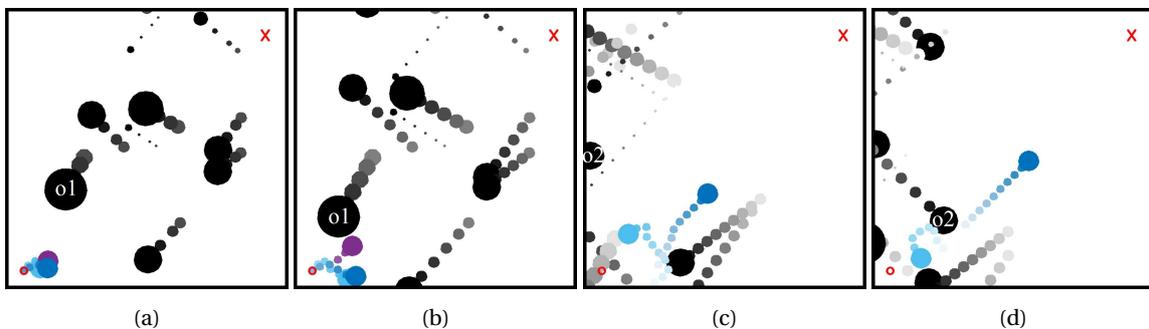


FIGURE 3.39 – Séquence comparant les planifications des approches *ris-apf* (cyan), *ris-bézier* (violet) et *ris-hybrid* (bleu) en présence de 10 obstacles plus rapides (noir).

- Sur la Fig. 3.39 les trois approches RIS sont à nouveau comparées. Dans ce scénario, *ris-hybrid* est la seule méthode qui permet au robot d'atteindre la destination en ayant évité toute collision. À l'étape (b), *ris-bézier* est en danger car aucun chemin n'a pu être trouvé pour contourner

l'obstacle $o1$ qui s'approche rapidement en face de lui. Le robot associé choisit donc de s'arrêter, ce qui ne l'empêche pas d'entrer en collision avec $o1$ quelques pas de temps plus tard. À l'étape (c), les trajectoires de *ris-apf* et *ris-hybrid*, jusqu'alors identiques car guidées par le même champ de potentiel, divergent : *ris-apf* s'éloigne de la destination en raison de l'arrivée rapide de l'obstacle $o2$, tandis que *ris-hybrid* est parvenu à trouver un chemin valide qui le rapproche de la destination tout en évitant $o2$. À l'étape (d), on peut constater que *ris-apf* est resté bloqué dans le coin inférieur gauche et finira par entrer en collision avec un obstacle. Le robot suivant la stratégie *ris-hybrid* parviendra à la destination après avoir bénéficié successivement des deux autres approches.

Une vidéo présentant plusieurs autres scénarios de validation est disponible en ligne : <https://youtu.be/Ccu41b56PjI>.

3.5.2 Validation en présence d'obstacles au mouvements stochastiques

Nous considérons la validation des approches de planification dans un environnement avec des obstacles mobiles dont le mouvement suit un processus stochastique. L'approche *ris-path* introduite section 3.4.2.3 a été comparée sur des scénarios identiques à 6 autres approches de planification issues de l'état de l'art :

- **static-apf** : une méthode de champ de potentiel classique qui considère que tous les obstacles sont statiques (cf. LATOMBE [1991])
- **dynamic-apf** : une méthode de champ de potentiel adaptée aux environnements dynamiques qui tient compte des vitesses des obstacles (cf. GE et CUI [2002])
- **vo** : l'approche V-obstacles (cf. l'algorithme introduit dans ? et dont le code source est disponible via VAN DEN BERG et collab. [2016])
- **apf-sr** : une approche utilisant un champ de potentiel classique généré par les ensembles stochastiques atteignables des obstacles (cf. CHIANG et collab. [2015a])
- **ses** : une approche utilisant un arbre à exploration rapide dont les nœuds sont générés en dehors des ensembles stochastiques atteignables des obstacles (cf. CHIANG et collab. [2015b])
- **drt** : une amélioration incrémentale de *ses* qui introduit la notion de seuil de risque dynamique (cf. CHIANG et collab. [2017b])

Nous avons implémenté les trois algorithmes qui utilisent les ensembles stochastiques atteignables en langage python à partir des spécifications décrites dans la littérature associée. Les implémentations des trois autres approches sont détaillées section 3.5.1.

L'approche *ris-path* utilise la collection de cartes de densité de probabilité de présence d'un obstacle (cf. section 3.3.3). Cette collection comprend également les polygones associés aux différents seuils de risque attribués à chaque pas de temps. Les régions RIS sont obtenues par un calcul en 2 dimensions, en faisant à chaque pas de temps l'intersection entre le polygone associé au mouvement du robot et les polygones de présence des obstacles. Puis, ces zones d'intersection sont fusionnées afin d'obtenir les polygones des RIS.

La planification de trajectoire de l'approche *ris-path* à partir des RIS est implémentée en python.

3.5.2.1 Les scénarios de test

Afin de pouvoir comparer ces 7 approches, des simulations ont été effectuées dans 4 environnements différents :

- **open** : une surface de 1000x1000 sans obstacles statiques dans laquelle circulent des obstacles dont les angles de braquage sont constants. Le robot A démarre à la position (250, 500) et doit atteindre la destination située à (750, 500). Le robot ego est holonome et assimilé à un point. Cet environnement est représenté Fig. 3.41(a).

- **gate** : une surface de 1000x1000 contenant deux obstacles statiques rectangulaires formant une porte que le robot doit traverser ainsi que des obstacles mobiles similaires à deux dans l'environnement *open*. A démarre à la position (200, 900) et doit atteindre la destination (800, 900). Le robot ego est holonome et assimilé à un point. Cet environnement est représenté Fig. 3.41(c).
- **lanes** : une surface de 1000x1000 sans obstacles statiques sur laquelle circulent deux flux de véhicules en sens contraire à la manière d'une route à deux fois trois voies. Le robot A démarre à la position (500, 350) et doit traverser les quatre voies afin de parvenir à la position (500, 650). Le robot ego est holonome et assimilé à un point. Cet environnement est représenté Fig. 3.41(d).
- **open-rect** : une surface de 1500x1500 sans obstacles statiques dans laquelle circulent des obstacles mobiles similaires à deux dans l'environnement *open*. Le robot A démarre à la position (375, 750) et doit atteindre la destination située à (1125, 750). Le robot ego est un véhicule de Dubins de la forme d'un rectangle de 32x16. Cet environnement est représenté Fig. 3.41(a).

Les mesures de distance, données sans unité, ont été implémentées en pratique en nombre de pixels. Les valeurs ont été spécifiées en s'inspirant de la comparaison d'approches effectuée dans la revue de littérature [CHIANG et collab. \[2017a\]](#), afin de conserver la densité d'obstacles dans l'environnement et le rapport de la vitesse du robot par rapport à celles des obstacles.

Dans tous les environnements, la vitesse du robot ego est limitée à une vitesse maximale v_{Amax} de 1.34 unités par pas de temps. Lorsqu'il est holonome, le robot peut faire varier instantanément sa vitesse. Le modèle de véhicule de Dubins utilisé dans l'environnement *open-rect* a une accélération limitée à 0.7 unités par pas de temps carré. Il peut faire varier son angle de lacet de 0.09 radians à chaque pas de temps.

Dans les environnements où le robot est holonome (*open*, *gate* et *lanes*), les différentes approches de planification sont calculées tous les 10 pas de temps. Cela permet d'accélérer les calculs sans pour autant impacter les performances (cf. Fig. 3.40). Dans ces environnements, l'horizon temporel utilisé est de 50 pas de temps, pour toutes les approches.

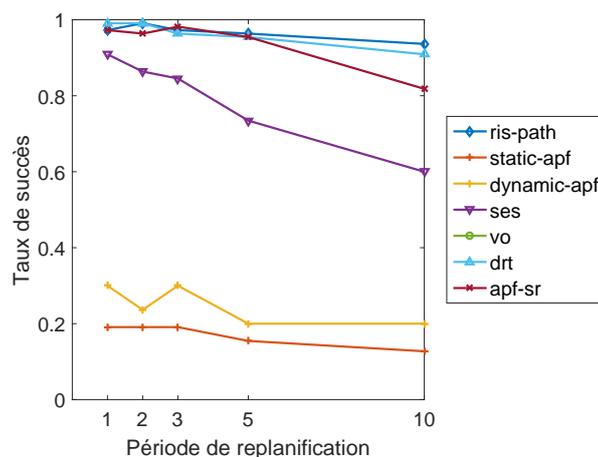


FIGURE 3.40 – Tests sur la période de replanification dans l'environnement *open* et pour un robot holonome.

Pour l'environnement *open-rect*, les calculs de planification sont effectués à chaque pas de temps, en raison de la difficulté de la tâche pour un robot non holonome et dont la forme n'est plus réduite à un point. L'horizon temporel dans cet environnement est également augmenté, à 100 pas de temps.

Les obstacles sont toujours en forme d'un rectangle de 42x18. Ils peuvent se superposer entre eux et leur trajectoire n'est pas influencée par la présence du robot ou des autres obstacles.

Lorsqu'un obstacle atteint un des bords dans l'environnement *lanes*, il est déplacé instantanément de l'autre côté de l'environnement de façon à ce que les flux d'obstacles soient continus. Dans les trois autres environnements, la trajectoire d'un obstacle atteignant un bord est comme réfléchi (une des composantes de vitesse de l'obstacle change de signe). Ainsi, la densité d'obstacles reste constante pour tous les environnements.

Les vitesses instantanées des obstacles changent tous les 10 pas de temps, en étant tirées selon la loi aléatoire attribuée au scénario. Voici les différentes lois aléatoires de vitesse disponibles :

- **var-faible** : cette loi de probabilité des vitesses décrite par le Tableau 3.2 correspond à celle utilisée dans le comparatif de CHIANG et collab. [2017a]. La variance des vitesses est relativement faible. Cette loi est utilisée dans tous les environnements (*open*, *gate*, *lanes* et *open-rect*).
- **var-élevée** : cette loi de probabilité est décrite par le Tableau 3.3. La variance des vitesses est plus élevée et on peut remarquer que les obstacles sont en moyenne nettement plus rapides que le robot. Cette loi n'est utilisée que dans les environnements *open* et *gate*.
- **var-très-élevée** : cette loi de probabilité est décrite par le Tableau 3.4. La variance des vitesses est très élevée : les obstacles peuvent atteindre des vitesses dépassant plus de 7 fois celle du robot, puis s'arrêter instantanément. Cette loi est utilisée dans les environnements *open* et *gate*.

| | | | | | |
|-------------|---------------|---------------|---------------|---------------|---------------|
| vitesse | 1.13 | 1.69 | 2.24 | 2.81 | 3.375 |
| probabilité | $\frac{1}{5}$ | $\frac{1}{5}$ | $\frac{1}{5}$ | $\frac{1}{5}$ | $\frac{1}{5}$ |

TABLEAU 3.2 – La loi de vitesse *var-faible*

| | | | | |
|-------------|---------------|---------------|---------------|---------------|
| vitesse | 0.8 | 2.24 | 4.0 | 5.36 |
| probabilité | $\frac{2}{6}$ | $\frac{1}{6}$ | $\frac{1}{6}$ | $\frac{2}{6}$ |

TABLEAU 3.3 – La loi de vitesse *var-élevée*

| | | | | | | |
|-------------|----------------|----------------|----------------|----------------|----------------|----------------|
| vitesse | 0.0 | 0.8 | 2.24 | 4.0 | 5.36 | 10 |
| probabilité | $\frac{1}{14}$ | $\frac{4}{14}$ | $\frac{2}{14}$ | $\frac{2}{14}$ | $\frac{4}{14}$ | $\frac{1}{14}$ |

TABLEAU 3.4 – La loi de vitesse *var-très-élevée*

Pour chaque environnement et pour chaque variante en matière de densité d'obstacles et de loi de vitesse, 100 états initiaux sont générés aléatoirement comme suit :

- La position initiale de chaque obstacle est tirée aléatoirement sur la surface de l'environnement à l'exception d'une zone qui entoure la position initiale de A. Dans l'environnement *lanes*, les obstacles sont placés aléatoirement sur une des 6 voies.
- Dans les environnements *open*, *gate* et *open-rect*, un angle de braquage est attribué à chaque obstacle. Celui-ci est tiré aléatoirement dans $\{-\frac{\pi}{106}, -\frac{\pi}{160}, -\frac{\pi}{320}, 0, \frac{\pi}{320}, \frac{\pi}{160}, \frac{\pi}{106}\}$, en radians par pas de temps. Pour autant, ces trajectoires ne sont pas circulaires puisque les vitesses des obstacles changent au cours du temps.

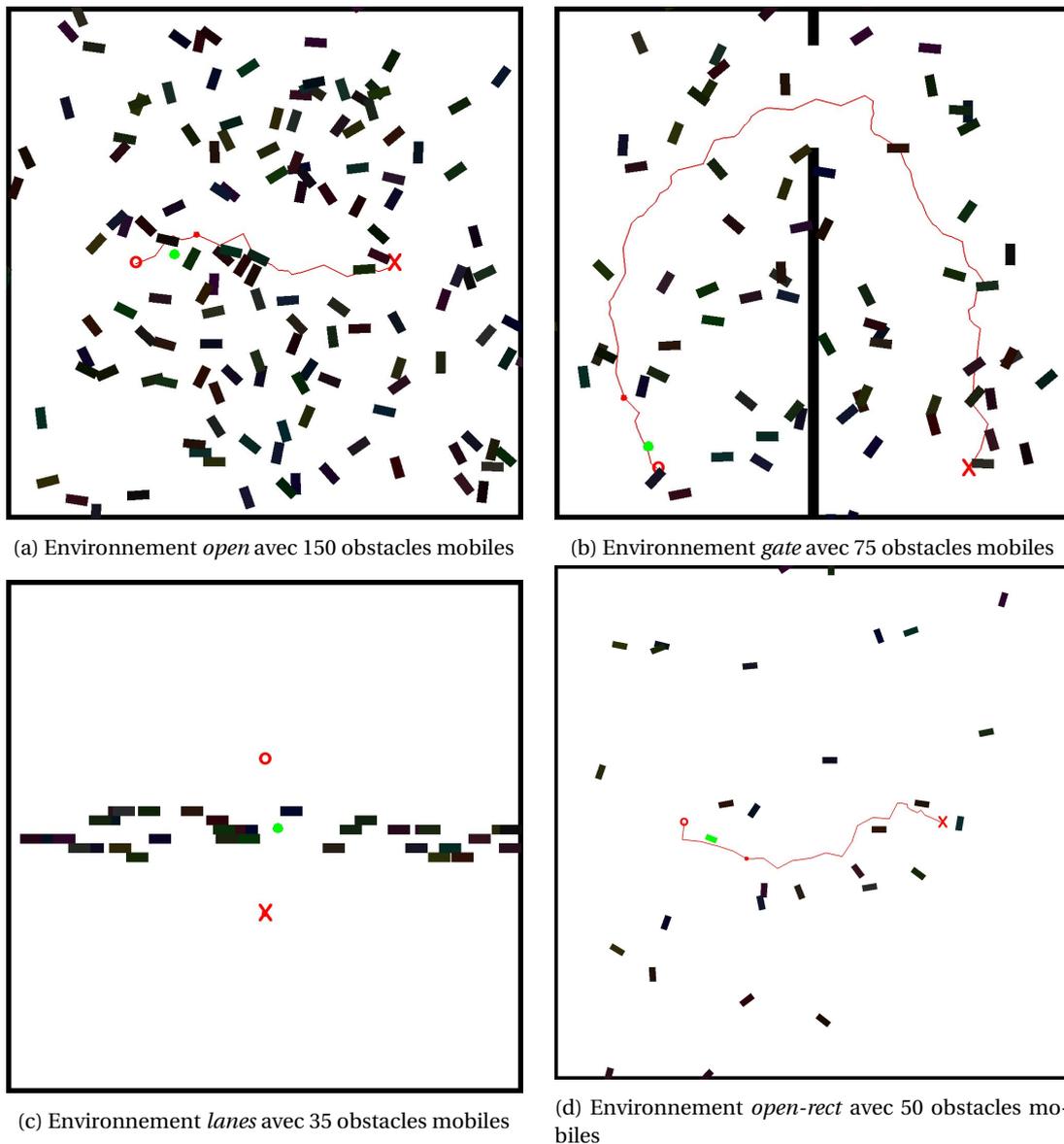


FIGURE 3.41 – Les quatre environnements de validation. Les points de départ et de destination sont respectivement marqués par un O et un X rouge. Le chemin global évitant les obstacles statiques est calculé par un arbre d'exploration RRT et est tracé en rouge. Un disque vert de 10 unités de rayon indique la présence du robot ponctuel. La forme exacte du robot rectangulaire est tracée en vert dans la Figure (d).

3.5.2.2 Résultats

Les différentes approches sont évaluées en comparant quantitativement :

- leur **taux de succès**, égal au nombre de scénarios pour lesquels A réussit à atteindre la destination divisé par 100.
- la **longueur moyenne du chemin**, normalisée par la longueur du plus court chemin possible entre le point de départ et la destination. Les cas d'échec ne sont pas considérés.
- la variation totale de **l'angle de lacet**, mesurée en radian. Elle est calculée en intégrant les variations d'orientation du robot, en valeur absolue, sur toute la durée du trajet. Cette valeur mesure de combien le robot a fait varier son orientation tout au long du chemin. Les cas d'échec ne sont pas considérés.

Les graphes des Figures 3.43, 3.44 et 3.45 présentent les performances des approches dans l'environnement *open*. Observons tout d'abord les taux de succès à gauche des figures. Dans toutes les variantes en matière de densité d'obstacles mobiles et pour les 3 lois de vitesses stochastiques des obstacles, l'approche *ris-path* affiche un taux de succès nettement supérieur à ceux des autres méthodes. Les résultats confirment le fait que les toutes méthodes de planification utilisant une prédiction du futur mouvement stochastique des obstacles ont un taux de succès supérieur aux méthodes réactives (*static-apf*, *dynamic-apf* et *vo*), qui ne peuvent intégrer à leurs calculs que la position et la vitesse moyenne de chaque obstacle. On peut également observer que, d'une manière générale, lorsque la densité d'obstacles commence à augmenter, le taux de succès de *ris-path* décroît moins fortement que ceux des autres méthodes, bien qu'il se mette à décroître plus fortement ensuite.

Les deux autres métriques ont pour objet de quantifier la qualité du chemin trouvé. On observe que les approches *vo* et *ses* réussissent à trouver des chemins sensiblement plus courts et droits que les autres approches. Certaines approches comme *apf-sr*, *drt* et *ris-path* trouvent souvent des chemins plus longs, dépassant de 2 fois la distance entre le point de départ et la destination. Il est intéressant de remarquer à quel point les métriques de longueur de chemin et de variation de l'angle de lacet sont fortement corrélées pour toutes les approches à l'exception notable de *ris-path*. Cela suggère que *ris-path* parvient à générer des trajectoires plus rectilignes par rapport aux autres approches, même si ses chemins ne sont pas les plus courts. A ce stade il est important de noter que ces deux mesures sont biaisées du fait qu'elles ne moyennent que les chemins parvenant à la destination. Ainsi, si une approche échoue dans tous les scénarios sauf un, dans lequel elle parvient à suivre un chemin rectiligne, elle verra sa longueur de chemin moyenne égale à 1 et sa variation de lacet à 0. Une description plus qualitative des chemins obtenus sera faite par la suite, en visualisant des exemples de chemins obtenus.

De façon similaire, les graphes des Figures 3.47, 3.48 et 3.49 présentent les performances des approches dans l'environnement *gate* pour les trois lois de vitesses disponibles. On note que l'ajout d'obstacles statiques formant une porte à travers laquelle le robot doit passer a pour principal effet de complexifier le problème de planification, et ainsi de diminuer légèrement le taux de succès de toutes les approches. On peut toutefois remarquer que l'écart entre l'approche *ris-path* et les autres semble augmenter sur cet environnement par rapport à *open*. Ces observations s'expliquent simplement par le fait que la présence de la porte force le robot à rester plus longtemps au sein de l'environnement, pour lequel la densité et le type d'obstacles sont les mêmes que dans l'environnement *open*.

Les résultats dans l'environnement *lanes* sont montrés sur la Fig. 3.51. Dans cet environnement la densité d'obstacles est bien plus forte au niveau des 6 voies de circulation, c'est pourquoi on observe une diminution plus rapide du taux de succès lorsque le nombre d'obstacles augmente, par rapport aux autres environnements. En matière de taux de succès, l'ordre des approches les plus performantes est le même que dans les autres environnements. On note que les chemins des approches *static-apf* et *dynamic-apf* sont sensiblement plus longs dans cet environnement. Cela s'explique par le fait que les champs de potentiel ne laissent pas facilement le robot s'insérer entre deux obstacles rapprochés, ce qui est souvent nécessaire pour pouvoir traverser

l'environnement *lanes*.

Les résultats dans l'environnement *open-rect* sont montrés sur la Fig. 3.52. Le taux de succès des approches réactives est globalement meilleur que dans l'environnement *open*, ce qui s'explique par la plus faible densité des obstacles mobiles (l'environnement est plus large, cf. Fig. 3.41). A l'inverse, les performances des autres approches de planification sont moins bonnes, ce qui montre la difficulté de calculer un chemin pour un véhicule de Dubins non holonome, a fortiori qui n'est plus ponctuel. En effet, il est bien plus difficile pour ce robot d'éviter la trajectoire future d'un obstacle mobile, même si celle-ci est correctement anticipée. L'approche *ris-path* affiche des performances en matière de taux de succès nettement meilleures que toutes les autres approches, ce qui montre que cette approche est adaptée à la planification de modèles de véhicules plus complexes.

Afin de mieux distinguer les effets du nouvel espace de configuration introduit dans la section 3.4.2.2, les performances de deux approches *ris-path* ont été comparées dans la Fig. 3.53. L'environnement considéré est équivalent à *open-rect*, mais le robot est d'une forme allongée cinq fois plus longue que large (80x16). Il est à noter que la densité des obstacles reste relativement faible dans cet environnement. Les deux approches de planification sont les mêmes, mais la première est effectuée à partir d'une somme de Minkowski entre les contours des obstacles et la forme du robot, tandis que la seconde utilise les polygones définis par le nouvel espace de configuration que nous avons introduit. On peut constater l'intérêt de recourir à cet espace de configuration, tant pour le taux de succès que pour la longueur du chemin.

Les tracés des chemins des différentes approches dans l'environnement *open* sont affichés Figures 3.55 et 3.56, où les robots sont en présence de respectivement 50 et 150 obstacles. Ces exemples permettent de visualiser qualitativement la forme des chemins obtenus par les différentes approches. On remarque tout d'abord que la planification complète de l'approche *ris-path* lui permet de suivre un chemin souvent bien plus direct en direction de la destination que ne le permettent les autres approches, qui utilisent le chemin global calculé par un arbre d'exploration RRT. On observe également que ce chemin est plus « lisse », moins fragmenté que les autres. Le chemin calculé par *ris-path* parvient régulièrement à rester rectiligne sur une longue durée (lorsque les mouvements des obstacles le permettent), ce qui n'est pratiquement jamais le cas des autres approches notamment en raison de leur stratégie réactive ou par échantillonnage. De fait, même en l'absence de tout obstacle, un chemin planifié par une approche par échantillonnage ne serait jamais rectiligne par construction.

La Fig. 3.57 présente les tracés des chemins dans l'environnement *open-rect*, dans lequel le mouvement des robots est contraint par un rayon de courbure minimum. On peut notamment observer ce rayon de courbure lorsque le robot effectue un tour complet sur lui-même afin d'éviter le trajet d'un obstacle. Du fait des contraintes non-holonomes des robots, tous les chemins apparaissent « lisses ». Cependant, on remarque ici aussi que la trajectoire du robot suivant la méthode *ris-path* est plus souvent rectiligne que toutes les autres méthodes.

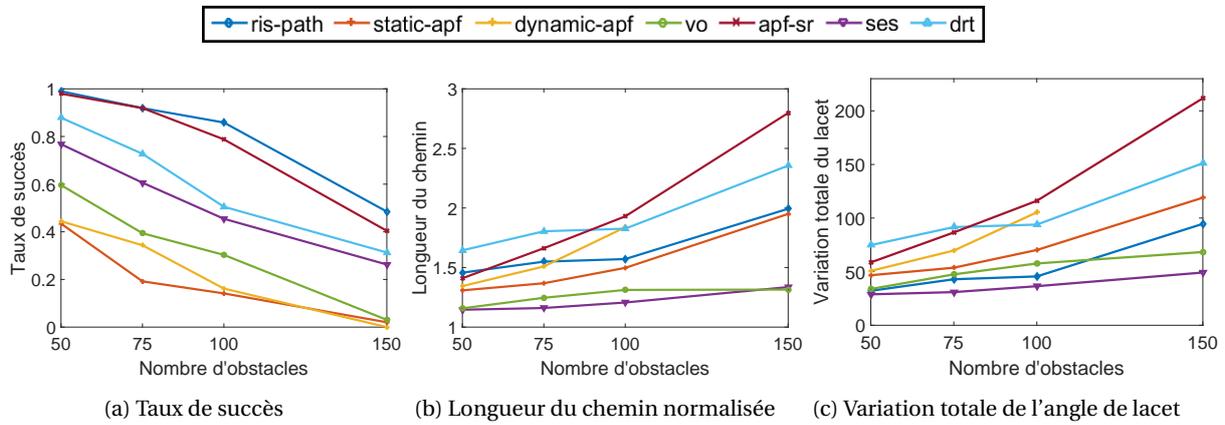


FIGURE 3.43 – Résultats dans l'environnement *open* et pour la loi de vitesse *var-faible*.

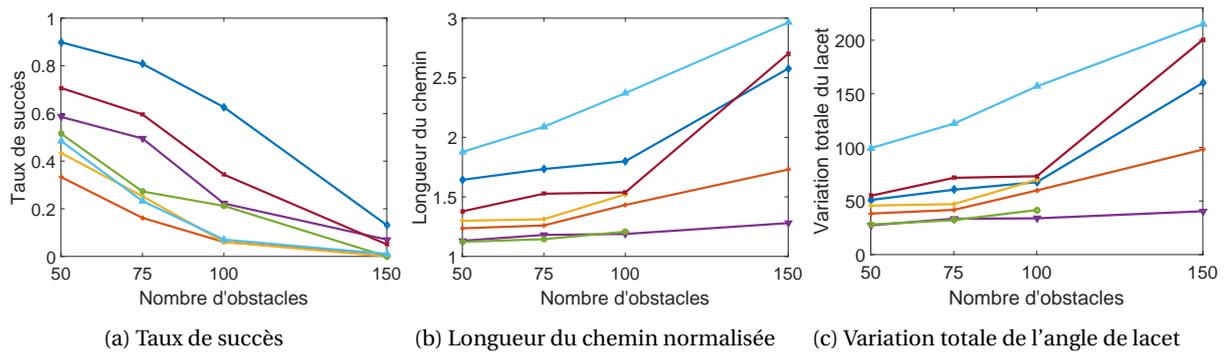


FIGURE 3.44 – Résultats dans l'environnement *open* et pour la loi de vitesse *var-élevée*.

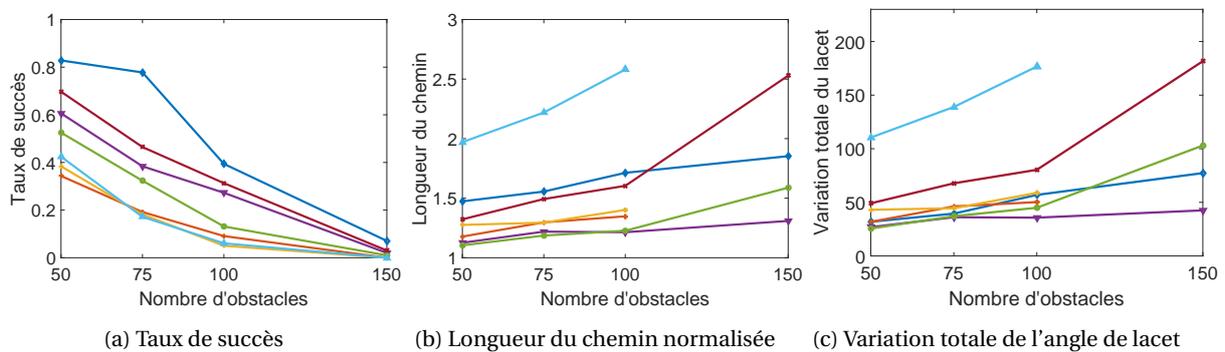


FIGURE 3.45 – Résultats dans l'environnement *open* et pour la loi de vitesse *var-très-élevée*.

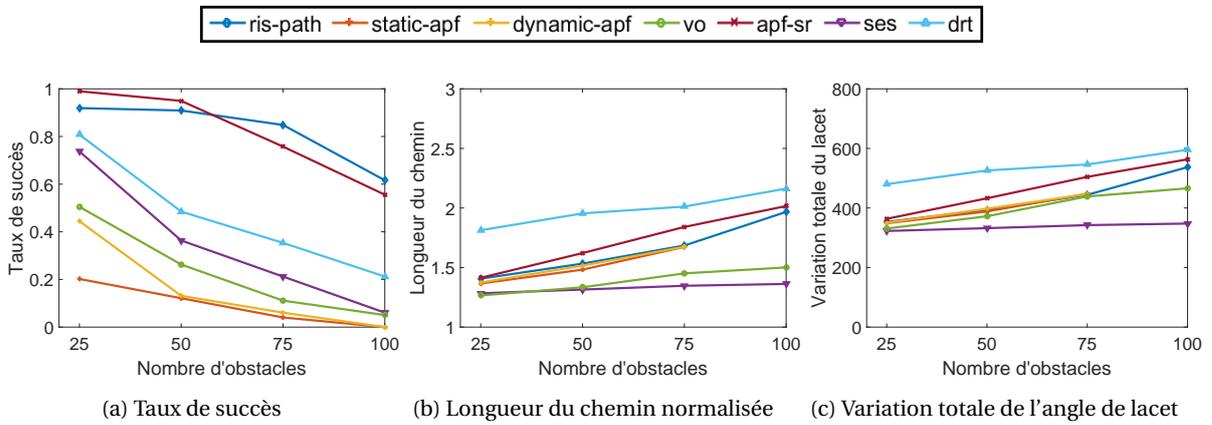


FIGURE 3.47 – Résultats dans l'environnement *gate* et pour la loi de vitesse *var-faible*.

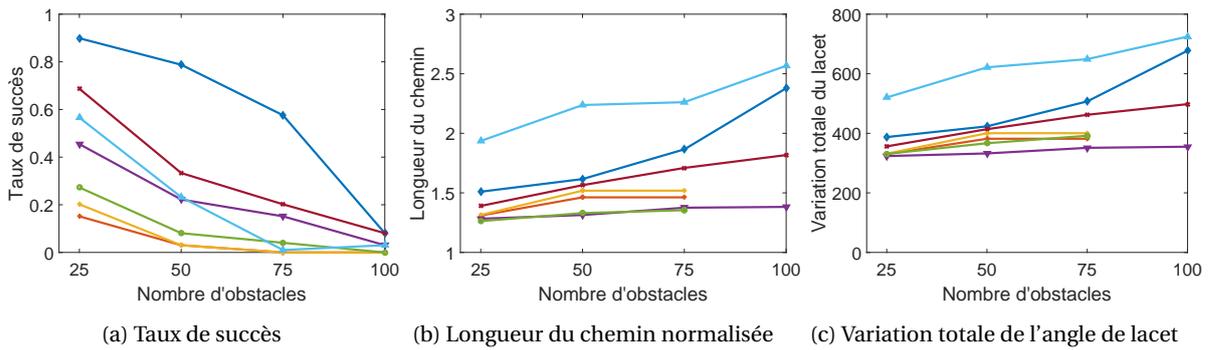


FIGURE 3.48 – Résultats dans l'environnement *gate* et pour la loi de vitesse *var-élevée*.

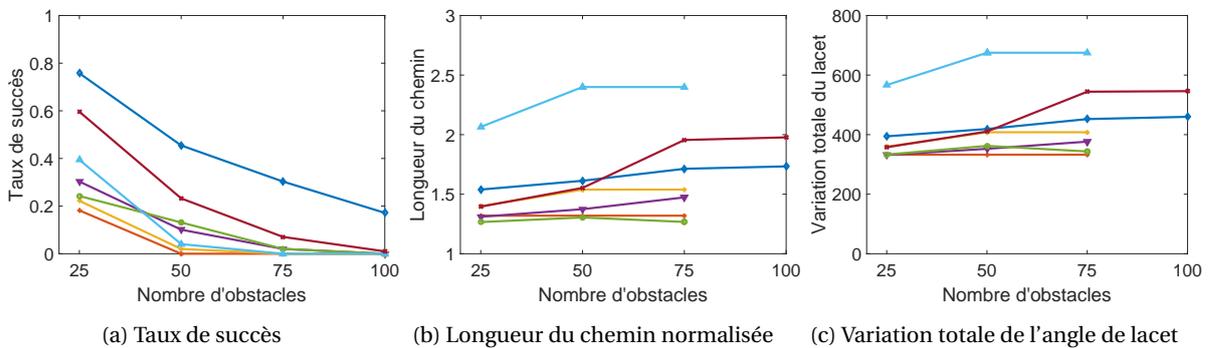


FIGURE 3.49 – Résultats dans l'environnement *gate* et pour la loi de vitesse *var-très-élevée*.

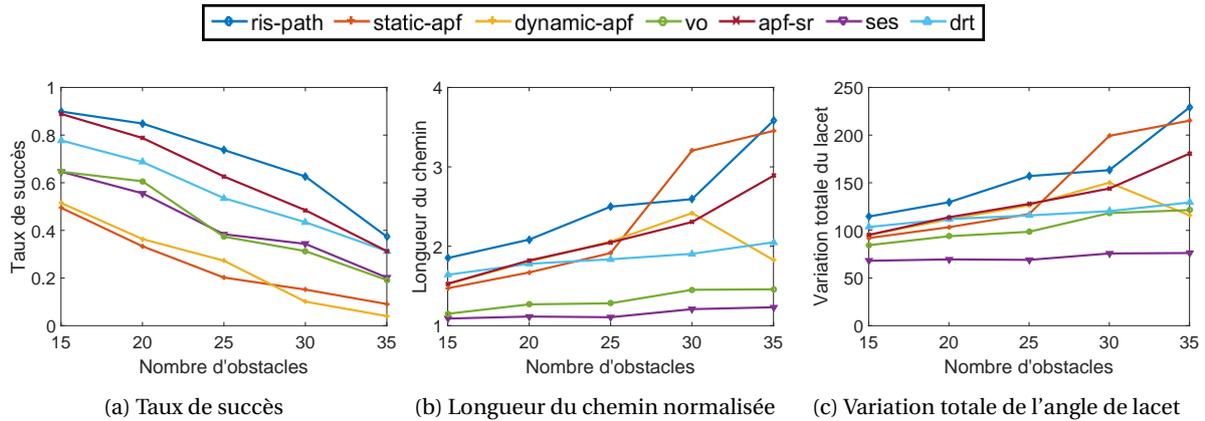


FIGURE 3.51 – Résultats dans l'environnement *lanes* et pour la loi de vitesse *var-faible*.

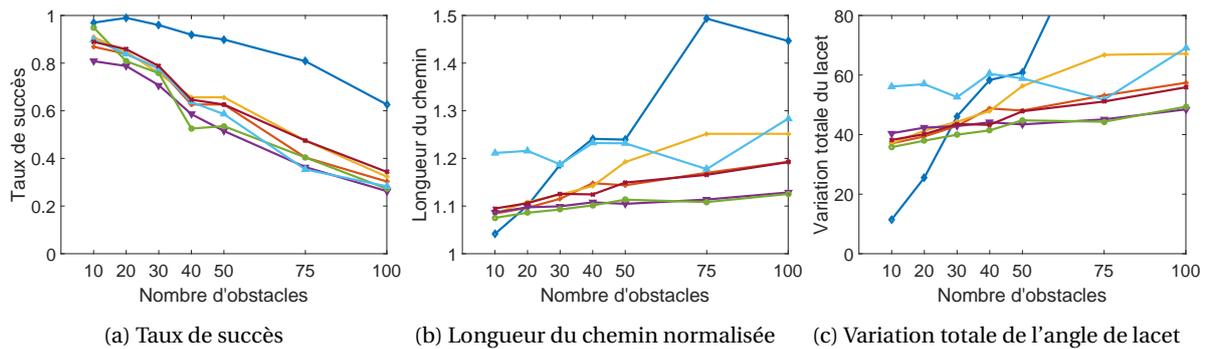


FIGURE 3.52 – Résultats dans l'environnement *open-rect* et pour la loi de vitesse *var-faible*. Dans cet environnement, le robot est de forme rectangulaire et est soumis aux contraintes non holonomes d'un véhicule de Dubins.

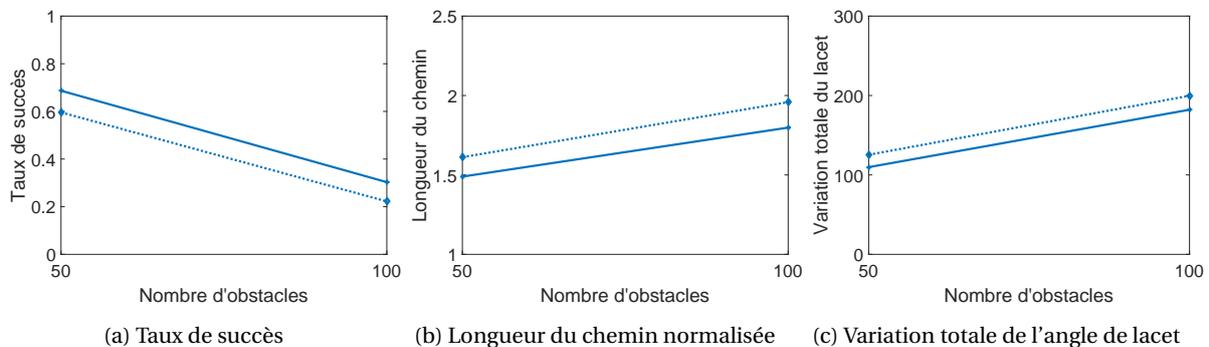


FIGURE 3.53 – Résultats dans l'environnement *open-rect* et pour la loi de vitesse *var-faible* et un robot rectangulaire cinq fois plus long que large soumis aux contraintes non holonomes d'un véhicule de Dubins.

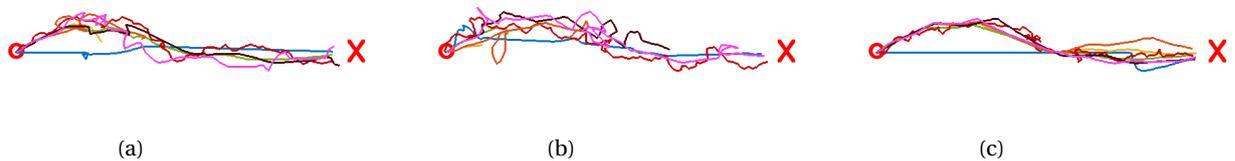


FIGURE 3.55 – Tracés de 3 chemins dans l’environnement *open*. Le robot est en présence de 50 obstacles dont les vitesses suivent la loi *var-faible*.

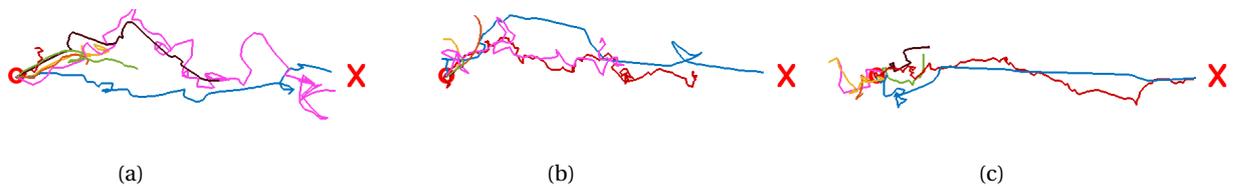


FIGURE 3.56 – Tracés de 3 chemins dans l’environnement *open*. Le robot est en présence de 150 obstacles dont les vitesses suivent la loi *var-faible*.



FIGURE 3.57 – Tracés de 3 chemins dans l’environnement *open-rect*. Le robot est en présence de 50 obstacles dont les vitesses suivent la loi *var-faible*.

3.6 Conclusion

Les « Reachable Interaction Sets » introduits dans ce chapitre proposent une nouvelle formulation simple du problème de planification de trajectoire parmi des obstacles mobiles.

L'utilisation des RIS avec différentes méthodes de planification a été proposée (une méthode réactive avec un champ de potentiel, une planification locale, une approche hybride, et une planification globale) et comparée à d'autres approches de l'état de l'art, démontrant une nette amélioration des performances pour guider le mouvement d'un robot dans un environnement dynamique. Les résultats montrent qu'en présence d'obstacles « très dynamiques », le taux de succès des approches utilisant les RIS diminue moins vite que celui des autres approches de l'état de l'art.

La représentation des RIS masque la complexité du problème en retirant la dimension temporelle de la planification. Elle rend possible l'utilisation d'une planification complète de chemin, même si dans un environnement très dynamique elle n'est valable en pratique que durant un court horizon temporel. A ce titre, un nouvel algorithme de recherche de chemin de Dubins a été présenté. Il s'appuie sur un chemin pour robot holonome trouvé par un graphe de visibilité. Ce type de planification génère un chemin de meilleure qualité, et rend les mouvements du robot plus ordonnés par rapport à une approche réactive ou de planification par échantillonnage.

L'heuristique que constitue la planification de chemin contournant les RIS repose sur deux hypothèses pouvant générer des situations dangereuses en cas d'erreur trop importante :

- La première hypothèse, commune à la plupart des approches de planification de trajectoire, est que le mouvement futur des obstacles mobiles peut faire l'objet d'une prédiction relativement précise, avec éventuellement une incertitude pouvant être quantifiée.
- La deuxième hypothèse est spécifique à la génération des RIS et considère que le chemin futur du robot sera « direct » : rectiligne dans le cas d'un robot holonome, ou un chemin de Dubins constitué d'un arc de cercle puis d'une demi-droite dans le cas d'un véhicule de Dubins. Cette hypothèse est intrinsèquement fautive à partir du moment où le robot planifie un chemin qui contourne les RIS. Cependant, ce chemin est recalculé très régulièrement afin de compenser dynamiquement ces approximations.

Chapitre 4

Conclusion

Sommaire

| | |
|--|-----|
| 4.1 Planification de vitesse | 130 |
| 4.2 Planification de chemin | 131 |
| 4.3 Perspective future : vers une planification chemin et vitesse? | 132 |

Dans cette thèse, deux stratégies de décomposition chemin-vitesse conduisant à deux méthodes de planification de trajectoire dans un environnement peu contraint et fortement dynamique ont été proposées. Nous rappellerons leurs caractéristiques dans les sections 4.1 et 4.2 et nous indiquerons des perspectives d'amélioration propres à chaque approche.

La possibilité d'une heuristique complète qui intègre chemin et vitesse dans la planification de trajectoire sera enfin discutée.

4.1 Planification de vitesse

Les premiers travaux présentés dans cette thèse ont pour contexte la conduite d'un véhicule automatisé dans des situations de conduite urbaine, sur des intersections non supervisées ainsi que dans un environnement semi-contraint. Les véhicules présents dans l'environnement, le VA inclus, sont des voitures citadines de taille moyenne soumises aux contraintes dynamiques associées. La topologie de la route ne permet pas d'identifier des contraintes strictes sur les futures trajectoires des véhicules, ce qui complexifie le problème de la planification. Notre approche a consisté à décomposer la planification de trajectoire en un calcul dynamique du profil de vitesse sur un chemin établi d'avance. En effet, dans le cadre de la conduite d'une voiture dans un environnement routier, il est nettement préférable d'ajuster plutôt la vitesse en fonction des autres usagers de la route, afin de minimiser les changements brusques de chemin.

L'algorithme introduit a les caractéristiques suivantes :

- L'algorithme adapte la vitesse du VA en combinant plusieurs hypothèses de profils de vitesse générés et adaptés en permanence selon la situation. Ces profils de référence sont inspirés de la conduite humaine afin que le VA suive un profil de vitesse qui soit qualitativement acceptable pour des passagers humains ainsi que pour les autres conducteurs.
- Le TG (Time Gap) s'est avéré être une métrique de risque pertinente dans tous les scénarios considérés. Il peut complètement remplacer le PET dans des scénarios d'intersections simples, mais généralise mieux dans des situations plus particulières.
- Une prédiction sur un horizon fuyant est utilisée avec une modélisation MPC pour évaluer le risque de chaque profil de référence en fonction de la métrique de risque.
- Les hypothèses de profils de vitesse qui dépassent un certain seuil de risque sont retirées, et le profil le plus rapide est finalement retenu pour minimiser le temps de parcours.
- Différents scénarios ont été introduits et étudiés sur des intersections et d'autres sur un environnement semi-contraint.
- Un algorithme de gestion des interactions du VA avec plusieurs autres véhicules dans un environnement semi-contraint a été présenté.

L'approche a fait l'objet d'une validation quantitative en matière de sécurité dans plusieurs scénarios en simulation. Chaque scénario testé a fait l'objet de plusieurs milliers de variantes, certaines des situations générées se situant à la limite des capacités physiques du VA lui permettant encore d'éviter qu'une collision ne se produise. Les résultats en simulation montrent globalement une bonne robustesse de l'algorithme lorsqu'il est confronté à des situations critiques variées. En simulation, l'algorithme a permis d'éviter d'entrer en collision dans toutes les situations de croisement, que ce soit sur une intersection ou dans un environnement semi-contraint.

Des expérimentations sur véhicules ont également été menées pour valider qualitativement la pertinence de l'approche en situation réelle, dans le cas d'une interaction avec deux véhicules sur une intersection simple. Elles ont notamment permis de valider le fait que le seuil de TG minimum doit être supérieur à 2.0s pour une conduite confortable.

Les scénarios de tests sur environnement semi-contraint ont pu montrer les limitations de l'approche lorsque la topologie de la route comporte une sémantique forte, qui n'est pas prise en

compte par l'approche de planification. En effet, les contraintes imposées par le code de la route ou par la topologie des voies navigables ne sont pas l'objet des présents travaux. Par exemple, lorsque les autres véhicules ont la priorité et s'insèrent rapidement devant le VA, il s'avère dangereux d'utiliser un seuil de TG minimum trop faible. Une valeur faible pousse en effet le VA à continuer sa course au moindre laps de temps durant lequel la voie est libre, et à s'insérer alors devant un autre conducteur qui s'attend à ce que le code de la route soit respecté. Pour respecter le code de la route ainsi que les conventions de conduite, il serait nécessaire de moduler le seuil du risque que le VA est capable de prendre en fonction du contexte de conduite.

Perspectives d'amélioration

- Seul le risque de la situation a été considéré dans cette approche d'adaptation dynamique de la vitesse. Cependant, d'autres critères pourraient facilement être intégrés lors du choix du profil de vitesse final, tels que le confort des passagers ou la consommation du véhicule.
- L'approche pourrait être adaptée à des environnements urbains classiques où les voies de circulation et les priorités entre les véhicules sont bien déterminées par la topologie de la route. Ceci permettrait d'utiliser une seule et même architecture de planification du profil de vitesse dans des environnements très divers, qu'ils soient contraints ou non par la topologie de la route, et d'utiliser une prédiction de mouvement pour les autres véhicules qui soit adaptée selon le contexte. Ainsi, cette approche de planification serait également applicable dans les situations délicates de « transition » pour entrer ou sortir d'une zone où l'environnement est semi-contraint.

4.2 Planification de chemin

Une stratégie moins classique a été introduite, consistant à décomposer la planification de trajectoire en émettant au préalable une hypothèse sur la vitesse du robot, puis en recherchant un chemin admissible. Cette approche a été traitée dans un environnement plus théorique, où des obstacles mobiles aux formes abstraites peuvent se superposer les uns aux autres.

Les Reachable Interaction Sets (RIS) sont une nouvelle représentation de l'environnement dynamique dans lequel se déplace le robot :

- L'hypothèse émise sur la vitesse du robot permet de supprimer l'aspect temporel du problème de planification, ce qui le ramène à une recherche de chemin duale. Il est alors possible d'utiliser une approche de recherche de chemin « classique », en considérant que la nouvelle représentation de l'environnement basée sur les RIS est quasi-statique.
- La définition des RIS a été généralisée à différents types de véhicules. L'approche a été présentée dans le cas d'un robot holonome ainsi que pour un véhicule de Dubins.
- De même, l'approche a été adaptée pour intégrer l'incertitude dans la modélisation du mouvement futur des obstacles, ce qui a pu être utilisé avec des environnements dans lesquels les obstacles mobiles suivent une loi de mouvement stochastique.
- Différents algorithmes de planification de chemin ont intégré les RIS, parmi lesquels plusieurs approches de planification locale qui s'appuient sur un chemin global statique, ainsi qu'une nouvelle approche de planification complète d'un chemin de Dubins.

Les RIS permettent d'identifier les zones à contourner pour éviter les obstacles mobiles au cours du temps, sous certaines hypothèses pour le mouvement du robot. Cette approche peut rappeler les fondements des *Velocity Obstacles* qui permettent d'identifier, dans l'espace des vitesses, les vecteurs de vitesse constante admissibles pour le robot afin d'éviter les trajectoires des autres obstacles. Les zones RIS sont par définition strictement incluses dans les cônes des VO, cependant leur surface est bien plus réduite grâce à l'hypothèse sur la vitesse du robot, ce qui facilite la planification d'un chemin en dehors de ces zones dangereuses. Les *Inevitable Collision State* (ICS)

décrivent également un espace de configuration avec des zones dangereuses à éviter. Ces zones sont celles où les capacités motrices du robot ne lui permettent plus d'éviter une collision future. Les approches de planification utilisant les ICS offrent donc une garantie supplémentaire que le robot ne se retrouve pas dans une situation critique. Cependant, la dimensionnalité du problème n'est pas réduite, c'est pourquoi les approches de planification avec ICS ont souvent recours à la construction d'un arbre de recherche pour trouver une trajectoire admissible.

Les résultats des tests comparant les planifications utilisant les RIS à d'autres approches de l'état de l'art montrent une nette amélioration du taux de succès pour guider le mouvement d'un robot dans un environnement dynamique. Les résultats montrent qu'en présence d'obstacles « très dynamiques », c'est à dire plus rapides que le robot, le taux de succès des approches utilisant les RIS diminue moins vite que celui des autres approches de l'état de l'art.

Perspectives d'amélioration

- L'aspect temps-réel de la planification de trajectoire n'a pas été considéré dans ces travaux. Plusieurs optimisations et parallélisations de l'algorithme pourraient être envisagées.
- Une perspective d'amélioration pour la définition des RIS serait d'affiner le calcul de la trajectoire de manière récursive. Le chemin obtenu en contournant les RIS donne en effet une hypothèse de trajectoire pour le robot. Ce chemin peut alors être utilisé pour calculer à nouveau les RIS de manière plus précise, en remplaçant l'hypothèse de chemin « direct » du robot. La prédiction des zones d'interaction avec les obstacles mobiles serait alors plus précise ce qui rendrait l'approche plus robuste.

4.3 Perspective future : vers une planification chemin et vitesse ?

En considérant les deux stratégies de décomposition de trajectoire proposées dans cette thèse –calculer un profil de vitesse et calculer un chemin–, il semble naturel de chercher à combiner les deux approches afin de générer une trajectoire sans se restreindre à une hypothèse formulée a priori.

En effet, puisque la pertinence d'une heuristique donnée dépend du contexte dans lequel elle est utilisée, il semble logique que les deux stratégies représentant les deux extrémités du continuum de la décomposition chemin-vitesse ne soient pas toujours les plus adaptées à la situation. Dans le contexte de la conduite d'un véhicule automatisé dans un environnement semi-contraint, il semble possible d'envisager la conception d'une heuristique intermédiaire, utilisant à la fois le calcul d'un profil de vitesse garantissant un seuil minimum de TG et le calcul de chemin qui contourne les RIS afin de permettre au robot de freiner ou bien de contourner un obstacle lorsque celui-ci se situe sur son chemin. Pour cela, une fonction de coût intégrant la longueur du chemin ainsi que le temps de parcours permettrait de combiner planification de chemin et planification de vitesse.

Cependant, l'étude de la complexité algorithmique de la planification de trajectoire en présence d'obstacles mobiles démontre qu'une telle fusion est loin d'être triviale, du fait de la complexité NP-difficile du problème.

Chapitre 5

Références

2018, «World Energy Outlook 2018», International Energy Agency. [2](#)

ABATE, A., M. PRANDINI, J. LYGEROS et S. SASTRY. 2008, «Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems», *Automatica*, vol. 44, n° 11, doi :10.1016/J.AUTOMATICA.2008.03.027, p. 2724–2734, ISSN 0005-1098. [88](#)

ALAHY, A., K. GOEL, V. RAMANATHAN, A. ROBICQUET, L. FEI-FEI et S. SAVARESE. 2016, «Social LSTM : Human Trajectory Prediction in Crowded Spaces», dans *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, ISBN 978-1-4673-8851-1, p. 961–971, doi : 10.1109/CVPR.2016.110. [25](#)

ALTCHÉ, F. et A. DE LA FORTELLE. 2017, «An LSTM network for highway trajectory prediction», dans *2017 20th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-5386-1526-3, p. 353–359, doi :10.1109/ITSC.2017.8317913. [25](#)

ALTCHÉ, F., X. QIAN et A. DE LA FORTELLE. 2016, «Time-optimal coordination of mobile robots along specified paths», dans *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-5090-3762-9, p. 5020–5026, doi :10.1109/IROS.2016.7759737. [18](#)

ALTHOFF, M., M. KOSCHI et S. MANZINGER. 2017, «CommonRoad : Composable benchmarks for motion planning on roads», dans *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5090-4804-5, p. 719–726, doi :10.1109/IVS.2017.7995802. [22](#)

AMMOUN, S. et F. NASHASHIBI. 2009, «Real time trajectory prediction for collision risk estimation between vehicles», dans *2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing*, IEEE, ISBN 978-1-4244-5007-7, p. 417–422, doi :10.1109/ICCP.2009.5284727. [23](#)

ARMAND, A., D. FILLIAT et J. IBANEZ-GUZMAN. 2014, «Ontology-based context awareness for driving assistance systems», dans *2014 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4799-3638-0, p. 227–233, doi :10.1109/IVS.2014.6856509. [14](#)

ASIMOV, I. 1942, *Runaround (I, robot)*, Doubleday, New York City, ISBN 0385423047, 192 p.. [82](#)

BÄRGMAN, J., K. SMITH et J. WERNEKE. 2015, «Quantifying drivers' comfort-zone and dread-zone boundaries in left turn across path/opposite direction (LTAP/OD) scenarios», *Transportation Research Part F : Traffic Psychology and Behaviour*, vol. 35, doi :10.1016/J.TRF.2015.10.003, p. 170–184, ISSN 1369-8478. [27](#), [29](#), [52](#), [66](#), [71](#)

BENDER, P., O. S. TAS, J. ZIEGLER et C. STILLER. 2015, «The combinatorial aspect of motion planning : Maneuver variants in structured environments», dans *2015 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4673-7266-4, p. 1386–1392, doi :10.1109/IVS.2015.7225909. [14](#), [15](#)

- VAN DEN BERG, J., S. J. GUY, J. SNAPE, M. C. LIN et D. MANOCHA. 2016, «RVO2 Library - Reciprocal Collision Avoidance for Real-Time Multi-Agent Simulation», <http://gamma.cs.unc.edu/RVO2/>, 87, 111, 118
- VAN DEN BERG, J. et M. OVERMARS. 2008, «Planning Time-Minimal Safe Paths Amidst Unpredictably Moving Obstacles», *The International Journal of Robotics Research*, vol. 27, n° 11-12, doi : 10.1177/0278364908097581, p. 1274–1294, ISSN 0278-3649. 85, 86
- BOJARSKI, M., D. DEL TESTA, D. DWORAKOWSKI, B. FIRNER, B. FLEPP, P. GOYAL, L. D. JACKEL, M. MONFORT, U. MULLER, J. ZHANG, X. ZHANG, J. ZHAO et K. ZIEBA. 2016, «End to End Learning for Self-Driving Cars», *CoRR*, vol. abs/1604.07316. 18
- BOUTON, M., A. COSGUN et M. J. KOCHENDERFER. 2017, «Belief state planning for autonomously navigating urban intersections», dans *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5090-4804-5, p. 825–830, doi :10.1109/IVS.2017.7995818. 13
- BRECHTEL, S., T. GINDELE et R. DILLMANN. 2014, «Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs», dans *2014 17th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-4799-6078-1, p. 392–399, doi :10.1109/ITSC.2014.6957722. 12
- BROADHURST, A., S. BAKER et T. KANADE. 2005, «Monte Carlo road safety reasoning», dans *2005 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 0-7803-8961-1, p. 319–324, doi :10.1109/IVS.2005.1505122. 23
- CANNY, J. et J. REIF. 1987, «New lower bound techniques for robot motion planning problems», dans *28th Annual Symposium on Foundations of Computer Science (SFCS 1987)*, IEEE, ISBN 0-8186-0807-2, p. 49–60, doi :10.1109/SFCS.1987.42. 80
- CHAKRAVARTHY, A. et D. GHOSE. 1998, «Obstacle avoidance in a dynamic environment : a collision cone approach», *IEEE Transactions on Systems, Man, and Cybernetics - Part A : Systems and Humans*, vol. 28, n° 5, doi :10.1109/3468.709600, p. 562–574, ISSN 10834427. 86
- CHAUDHARY, K. L. et D. GHOSE. 2017, «Path planning in dynamic environments with deforming obstacles using collision cones», dans *2017 Indian Control Conference (ICC)*, IEEE, ISBN 978-1-5090-1795-9, p. 87–92, doi :10.1109/INDIANCC.2017.7846457. 86
- CHEN, C., A. SEFF, A. KORNHAUSER et J. XIAO. 2015, «DeepDriving : Learning Affordance for Direct Perception in Autonomous Driving», dans *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE, ISBN 978-1-4673-8391-2, p. 2722–2730, doi :10.1109/ICCV.2015.312. 18
- CHEON, H. et B. K. KIM. 2019, «Online Bidirectional Trajectory Planning for Mobile Robots in State-Time Space», *IEEE Transactions on Industrial Electronics*, vol. 66, n° 6, doi :10.1109/TIE.2018.2866039, p. 4555–4565, ISSN 0278-0046. 79, 85
- CHEUNG, E., A. BERA, E. KUBIN, K. GRAY et D. MANOCHA. 2018, «Identifying Driver Behaviors Using Trajectory Features for Vehicle Navigation», dans *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-5386-8094-0, p. 3445–3452, doi :10.1109/IROS.2018.8594348. 19
- CHIANG, H.-T., N. MALONE, K. LESSER, M. OISHI et L. TAPIA. 2015a, «Path-guided artificial potential fields with stochastic reachable sets for motion planning in highly dynamic environments», dans *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-4799-6923-4, p. 2347–2354, doi :10.1109/ICRA.2015.7139511. 88, 89, 95, 118

- CHIANG, H.-T., N. RACKLEY et L. TAPIA. 2015b, «Stochastic Ensemble Simulation motion planning in stochastic dynamic environments», dans *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-4799-9994-1, p. 3836–3843, doi :10.1109/IROS.2015.7353916. [88](#), [89](#), [118](#)
- CHIANG, H.-T. L., B. HOMCHAUDHURI, L. SMITH et L. TAPIA. 2017a, «Safety , Challenges , and Performance of Motion Planners in Dynamic Environments», dans *2017 International Symposium of Robotics Research (ISRR)*. [119](#), [120](#)
- CHIANG, H.-T. L., B. HOMCHAUDHURI, A. P. VINOD, M. OISHI et L. TAPIA. 2017b, «Dynamic risk tolerance : Motion planning by balancing short-term and long-term stochastic dynamic predictions», dans *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-5090-4633-1, p. 3762–3769, doi :10.1109/ICRA.2017.7989434. [89](#), [96](#), [118](#)
- CHING-YAO CHAN. 2006, «Defining Safety Performance Measures of Driver-Assistance Systems for Intersection Left-Turn Conflicts», dans *2006 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 4-901122-86-X, p. 25–30, doi :10.1109/IVS.2006.1689600. [28](#)
- COLYAR, J. et J. HALKIAS. 2007, «Us highway 101 dataset.», dans *Federal Highway Administration (FHWA), Tech. Rep. FHWA-HRT07-030*. [23](#), [25](#)
- DEO, N. et M. M. TRIVEDI. 2018, «Convolutional Social Pooling for Vehicle Trajectory Prediction», dans *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, ISBN 978-1-5386-6100-0, p. 1549–15498, doi :10.1109/CVPRW.2018.00196. [25](#)
- DESARAJU, V., H. RO, M. YANG, E. TAY, S. ROTH et D. DEL VECCHIO. 2009, «Partial order techniques for vehicle collision avoidance : Application to an autonomous roundabout test-bed», dans *2009 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-4244-2788-8, p. 82–87, doi :10.1109/ROBOT.2009.5152656. [18](#)
- DUBINS, L. E. 1957, «On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents», *American Journal of Mathematics*, vol. 79, n° 3, doi :10.2307/2372560, p. 497, ISSN 00029327. [83](#)
- EIDEHALL, A. et L. PETERSSON. 2008, «Statistical Threat Assessment for General Road Scenes Using Monte Carlo Sampling», *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, n° 1, doi :10.1109/TITS.2007.909241, p. 137–147, ISSN 1524-9050. [23](#)
- ELVIK, R. 2014, «A review of game-theoretic models of road user behaviour», *Accident Analysis & Prevention*, vol. 62, doi :10.1016/J.AAP.2013.06.016, p. 388–396, ISSN 0001-4575. [17](#)
- ERDMANN, M. et T. LOZANO-PÉREZ. 1987, «On multiple moving objects», *Algorithmica*, vol. 2, n° 1-4, doi :10.1007/BF01840371, p. 477–521, ISSN 0178-4617. [84](#)
- FIORINI, P. et Z. SHILLER. 1993, «Motion planning in dynamic environments using the relative velocity paradigm», dans *1993 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE Comput. Soc. Press, ISBN 0-8186-3450-2, p. 560–565, doi :10.1109/ROBOT.1993.292038. [87](#)
- FIORINI, P. et Z. SHILLER. 1998, «Motion Planning in Dynamic Environments Using Velocity Obstacles», *The International Journal of Robotics Research*, vol. 17, n° 7, doi :10.1177/027836499801700706, p. 760–772, ISSN 0278-3649. [87](#), [88](#)
- FOX, D., W. BURGARD et S. THRUN. 1997, «The dynamic window approach to collision avoidance», *IEEE Robotics & Automation Magazine*, vol. 4, n° 1, doi :10.1109/100.580977, p. 23–33, ISSN 10709932. [90](#)
- FRAICHARD, T. 1998, «Trajectory planning in a dynamic workspace : a ‘state-time space’ approach», *Advanced Robotics*, vol. 13, n° 1, p. 75–94. [85](#)

- FRAICHARD, T. et H. ASAMA. 2003, «Inevitable collision states a step towards safer robots?», dans *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, IEEE, ISBN 0-7803-7860-1, p. 388–393, doi :10.1109/IROS.2003.1250659. [89](#)
- FRAICHARD, T. et A. SCHEUER. 2004, «From Reeds and Shepp's to Continuous-Curvature Paths», *IEEE Transactions on Robotics*, vol. 20, n° 6, doi :10.1109/TRO.2004.833789, p. 1025–1035, ISSN 1552-3098. [84](#), [106](#)
- FUJIMURA, K. et H. SAMET. 1990, «Motion planning in a dynamic domain», dans *1990 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE Comput. Soc. Press, ISBN 0-8186-9061-5, p. 324–330, doi :10.1109/ROBOT.1990.125996. [79](#), [83](#)
- GALCERAN, E., A. G. CUNNINGHAM, R. M. EUSTICE et E. OLSON. 2017, «Multipolicy decision-making for autonomous driving via changepoint-based behavior prediction : Theory and experiment», *Autonomous Robots*, vol. 41, n° 6, doi :10.1007/s10514-017-9619-z, p. 1367–1382, ISSN 0929-5593. [13](#)
- GE, S. et Y. CUI. 2002, «Dynamic Motion Planning for Mobile Robots Using Potential Field Method», *Autonomous Robots*, vol. 13, n° 3, doi :10.1023/A:1020564024509, p. 207–222, ISSN 09295593. [91](#), [111](#), [118](#)
- GERDES, J. C. et E. J. ROSSETTER. 2001, «A Unified Approach to Driver Assistance Systems Based on Artificial Potential Fields», *Journal of Dynamic Systems, Measurement, and Control*, vol. 123, n° 3, doi :10.1115/1.1386788, p. 431, ISSN 00220434. [16](#)
- GILLULA, J. H., G. M. HOFFMANN, H. HAOMIAO HUANG, M. P. VITUS et C. J. TOMLIN. 2011, «Applications of hybrid reachability analysis to robotic aerial vehicles», *The International Journal of Robotics Research*, vol. 30, n° 3, doi :10.1177/0278364910387173, p. 335–354, ISSN 0278-3649. [84](#)
- GINDELE, T., S. BRECHTEL et R. DILLMANN. 2010, «A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments», dans *2010 13th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-4244-7657-2, p. 1625–1631, doi :10.1109/ITSC.2010.5625262. [24](#)
- GONZALEZ, D., J. PEREZ, V. MILANES et F. NASHASHIBI. 2016, «A Review of Motion Planning Techniques for Automated Vehicles», *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, n° 4, doi :10.1109/TITS.2015.2498841, p. 1135–1145, ISSN 1524-9050. [12](#)
- GOTTE, C., M. KELLER, C. HASS, K.-H. GLANDER, A. SEEWALD et T. BERTRAM. 2015, «A model predictive combined planning and control approach for guidance of automated vehicles», dans *2015 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, IEEE, ISBN 978-1-4673-9114-6, p. 69–74, doi :10.1109/ICVES.2015.7396896. [16](#)
- GOVEA, A. D. V. 2010, *Incremental Learning for Motion Prediction of Pedestrians and Vehicles*, Springer Tracts in Advanced Robotics, vol. 64, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-642-13641-2, doi :10.1007/978-3-642-13642-9. [24](#)
- GRAF, R., H. DEUSCH, F. SEELIGER, M. FRITZSCHE et K. DIETMAYER. 2014, «A learning concept for behavior prediction at intersections», dans *2014 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4799-3638-0, p. 939–945, doi :10.1109/IVS.2014.6856415. [24](#)
- GRAF PLESSSEN, M., D. BERNARDINI, H. ESEN et A. BEMPORAD. 2018, «Spatial-Based Predictive Control and Geometric Corridor Planning for Adaptive Cruise Control Coupled With Obstacle Avoidance», *IEEE Transactions on Control Systems Technology*, vol. 26, n° 1, doi :10.1109/TCST.2017.2664722, p. 38–50, ISSN 1063-6536. [16](#)
- GUN, P., A. HILL et R. VUJANIC. 2018, «Multi-Vehicle Trajectory Optimisation On Road Networks», *CoRR*, vol. abs/1810.0251. [18](#)

- GUNNARSSON, J., L. SVENSSON, E. BENGTTSSON et L. DANIELSSON. 2006, «Joint Driver Intention Classification and Tracking of Vehicles», dans *2006 IEEE Nonlinear Statistical Signal Processing Workshop*, IEEE, ISBN 978-1-4244-0579-4, p. 95–98, doi :10.1109/NSSPW.2006.4378828. [24](#)
- HERMES, C., C. WOHLER, K. SCHENK et F. KUMMERT. 2009, «Long-term vehicle motion prediction», dans *2009 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4244-3503-6, p. 652–657, doi :10.1109/IVS.2009.5164354. [23](#)
- HILGERT, J., K. HIRSCH, T. BERTRAM et M. HILLER. 2003, «Emergency path planning for autonomous vehicles using elastic band theory», dans *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, vol. 2, IEEE, ISBN 0-7803-7759-1, p. 1390–1395, doi :10.1109/AIM.2003.1225546. [17](#)
- HOY, M., A. S. MATVEEV et A. V. SAVKIN. 2015, «Algorithms for collision-free navigation of mobile robots in complex cluttered environments : a survey», *Robotica*, vol. 33, n° 03, doi :10.1017/S0263574714000289, p. 463–497, ISSN 0263-5747. [80](#)
- HSU, D., R. KINDEL, J.-C. LATOMBE et S. ROCK. 2002, «Randomized Kinodynamic Motion Planning with Moving Obstacles», *The International Journal of Robotics Research*, vol. 21, n° 3, doi :10.1177/027836402320556421, p. 233–255, ISSN 0278-3649. [85](#)
- HUANG, L. 2009, «Velocity planning for a mobile robot to track a moving target — a potential field approach», *Robotics and Autonomous Systems*, vol. 57, n° 1, doi :10.1016/J.ROBOT.2008.02.005, p. 55–63, ISSN 0921-8890. [91](#)
- HUBMANN, C., M. AEBERHARD et C. STILLER. 2016, «A generic driving strategy for urban environments», dans *2016 19th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-5090-1889-5, p. 1010–1016, doi :10.1109/ITSC.2016.7795679. [15](#)
- HUBMANN, C., M. BECKER, D. ALTHOFF, D. LENZ et C. STILLER. 2017, «Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles», dans *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5090-4804-5, p. 1671–1678, doi :10.1109/IVS.2017.7995949. [13](#)
- HULSEN, M., J. M. ZOLLNER, N. HAEBERLEN et C. WEISS. 2011a, «Asynchronous real-time framework for knowledge-based intersection assistance», dans *2011 14th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-4577-2197-7, p. 1680–1685, doi :10.1109/ITSC.2011.6082810. [14](#)
- HULSEN, M., J. M. ZOLLNER et C. WEISS. 2011b, «Traffic intersection situation description ontology for advanced driver assistance», dans *2011 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4577-0890-9, p. 993–999, doi :10.1109/IVS.2011.5940415. [14](#)
- INTERGOVERNMENTAL PANEL ON CLIMATE CHANGE. 2018, «Summary for Policymakers - Global Warming of 1.5 °C», SR15 Report. [2](#)
- ISELE, D., A. COSGUN, K. SUBRAMANIAN et K. FUJIMURA. 2018, «Navigating Intersections with Autonomous Vehicles using Deep Reinforcement Learning», dans *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, p. 2034–2039. [18](#)
- JEON, H.-S., D.-S. KUM et W.-Y. JEONG. 2018, «Traffic Scene Prediction via Deep Learning : Introduction of Multi-Channel Occupancy Grid Map as a Scene Representation», dans *2018 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5386-4452-2, p. 1496–1501, doi : 10.1109/IVS.2018.8500567. [25](#)
- JEVONS, W. S. 1866, «The coal question; an inquiry concerning the progress of the nation, and the probable exhaustion of our coal-mines.», *Fortnightly*, vol. 6, n° 34, p. 505–507. [3](#)

- JIMÉNEZ, F., J. E. NARANJO et F. GARCÍA. 2013, «An Improved Method to Calculate the Time-to-Collision of Two Vehicles», *International Journal of Intelligent Transportation Systems Research*, vol. 11, n° 1, doi :10.1007/s13177-012-0054-4, p. 34–42, ISSN 1348-8503. 28
- JOHNSON, J. et K. HAUSER. 2012, «Optimal acceleration-bounded trajectory planning in dynamic environments along a specified path», dans *2012 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-4673-1405-3, p. 2035–2041, doi :10.1109/ICRA.2012.6225233. 14, 15
- JOHNSON, J. et K. HAUSER. 2013, «Optimal longitudinal control planning with moving obstacles», dans *2013 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4673-2755-8, p. 605–611, doi :10.1109/IVS.2013.6629533. 14, 15
- KANT, K. et S. W. ZUCKER. 1986, «Toward Efficient Trajectory Planning : The Path-Velocity Decomposition», *The International Journal of Robotics Research*, vol. 5, n° 3, doi :10.1177/027836498600500304, p. 72–89, ISSN 0278-3649. 6, 14
- KARAMAN, S. et E. FRAZZOLI. 2011, «Sampling-based algorithms for optimal motion planning», *The International Journal of Robotics Research*, vol. 30, n° 7, doi :10.1177/0278364911406761, p. 846–894, ISSN 0278-3649. 82, 98
- KELLER, M., F. HOFFMANN, C. HASS, T. BERTRAM et A. SEEWALD. 2014, «Planning of Optimal Collision Avoidance Trajectories with Timed Elastic Bands», *IFAC Proceedings Volumes*, vol. 47, n° 3, doi :10.3182/20140824-6-ZA-1003.01143, p. 9822–9827, ISSN 1474-6670. 17
- KHATIB, O. 1985, «Real-time obstacle avoidance for manipulators and mobile robots», dans *1985 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, Institute of Electrical and Electronics Engineers, p. 500–505, doi :10.1109/ROBOT.1985.1087247. 80
- KIM, B., C. M. KANG, J. KIM, S. H. LEE, C. C. CHUNG et J. W. CHOI. 2017, «Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network», dans *2017 20th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-5386-1526-3, p. 399–404, doi :10.1109/ITSC.2017.8317943. 25
- KITA, H. 1999, «A merging-giveway interaction model of cars in a merging section : a game theoretic analysis», *Transportation Research Part A : Policy and Practice*, vol. 33, n° 3-4, p. 305–312. 17
- KLUGE, B. 2003, «Recursive agent modeling with probabilistic velocity obstacles for mobile robot navigation among humans», dans *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, IEEE, ISBN 0-7803-7860-1, p. 376–381, doi :10.1109/IROS.2003.1250657. 87
- KRAJEWSKI, R., J. BOCK, L. KLOEKER et L. ECKSTEIN. 2018, «The highD dataset : A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems», dans *2018 21th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, ISSN 2153-0017, p. 2118–2125, doi :10.1109/ITSC.2018.8569552. 25
- KUEFLER, A., J. MORTON, T. WHEELER et M. KOCHENDERFER. 2017, «Imitating driver behavior with generative adversarial networks», dans *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5090-4804-5, p. 204–211, doi :10.1109/IVS.2017.7995721. 18
- KUMAR, P., M. PERROLLAZ, S. LEFÈVRE et C. LAUGIER. 2013, «Learning-based approach for online lane change intention prediction», dans *2013 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4673-2755-8, p. 797–802, doi :10.1109/IVS.2013.6629564. 24

- KYE, D.-K., S.-W. KIM et S.-W. SEO. 2015, «Decision making for automated driving at unsignalized intersection», dans *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, IEEE, ISBN 978-8-9932-1508-3, p. 522–525, doi :10.1109/ICCAS.2015.7364974. [13](#)
- LARGE, F., C. LAUGIER et Z. SHILLER. 2005, «Navigation Among Moving Obstacles Using the NLVO : Principles and Applications to Intelligent Vehicles», *Autonomous Robots*, vol. 19, n° 2, doi :10.1007/s10514-005-0610-8, p. 159–171, ISSN 0929-5593. [88](#)
- LARGE, F., S. SEKHAVAT, Z. SHILLER et C. LAUGIER. 2002, «Towards real-time global motion planning in a dynamic environment using the NLVO concept», dans *2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, IEEE, ISBN 0-7803-7398-7, p. 607–612, doi :10.1109/IRDS.2002.1041458. [88](#)
- LATOMBE, J.-C. 1991, *Robot motion planning*, Kluwer Academic Publishers, ISBN 9780792392064, 651 p.. [80](#), [111](#), [118](#)
- LAVALLE, S. M. 1998, «Rapidly-Exploring Random Trees : A New Tool for Path Planning», Report No. TR 98-11, Computer Science Department, Iowa State University. [82](#)
- LAVALLE, S. M. 2006, *Planning algorithms*, Cambridge University Press, ISBN 0521862051, 826 p.. [80](#), [83](#), [84](#), [85](#), [94](#)
- LAVALLE, S. M. et J. J. KUFFNER. 2001, «Randomized Kinodynamic Planning», *The International Journal of Robotics Research*, vol. 20, n° 5, doi :10.1177/02783640122067453, p. 378–400, ISSN 0278-3649. [82](#)
- LAWITZKY, A., D. ALTHOFF, C. F. PASSENBERG, G. TANZMEISTER, D. WOLLHERR et M. BUSS. 2013, «Interactive scene prediction for automotive applications», dans *2013 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4673-2755-8, p. 1028–1033, doi :10.1109/IVS.2013.6629601. [24](#)
- LEE, J. et K. JANG. 2017, «A framework for evaluating aggressive driving behaviors based on in-vehicle driving records», *Transportation Research Part F: Traffic Psychology and Behaviour*, doi :10.1016/J.TRF.2017.11.021, ISSN 1369-8478. [19](#)
- LEFÈVRE, S., C. LAUGIER et J. IBAÑEZ-GUZMÁN. 2012, «Evaluating Risk at Road Intersections by Detecting Conflicting Intentions», dans *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, p. 4841–4846. [24](#)
- LEFÈVRE, S., D. VASQUEZ et C. LAUGIER. 2014, «A survey on motion prediction and risk assessment for intelligent vehicles», *ROBOMECH Journal*, vol. 1, n° 1, doi :10.1186/s40648-014-0001-z, p. 1, ISSN 2197-4225. [22](#), [23](#), [25](#)
- LENZ, D., F. DIEHL, M. T. LE et A. KNOLL. 2017, «Deep neural networks for Markovian interactive scene prediction in highway scenarios», dans *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5090-4804-5, p. 685–692, doi :10.1109/IVS.2017.7995797. [25](#)
- LIEBNER, M., F. KLANNER, M. BAUMANN, C. RUHHAMMER et C. STILLER. 2013, «Velocity-Based Driver Intent Inference at Urban Intersections in the Presence of Preceding Vehicles», *IEEE Intelligent Transportation Systems Magazine*, vol. 5, n° 2, doi :10.1109/MITS.2013.2246291, p. 10–21, ISSN 1939-1390. [23](#)
- LIKHACHEV, M. et D. FERGUSON. 2009, «Planning Long Dynamically Feasible Maneuvers for Autonomous Vehicles», *The International Journal of Robotics Research*, vol. 28, n° 8, doi :10.1177/0278364909340445, p. 933–945, ISSN 0278-3649. [86](#)
- LOZANO-PEREZ, T. et M. A. WESLEY. 1979, «An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles», *Communications of the ACM*, vol. 22. [83](#)

- MAC, T. T., C. COPOT, D. T. TRAN et R. DE KEYSER. 2016, «Heuristic approaches in robot path planning : A survey», *Robotics and Autonomous Systems*, vol. 86, doi :10.1016/J.ROBOT.2016.08.001, p. 13–28, ISSN 0921-8890. [80](#)
- MALONE, N., H.-T. CHIANG, K. LESSER, M. OISHI et L. TAPIA. 2017, «Hybrid Dynamic Moving Obstacle Avoidance Using a Stochastic Reachable Set-Based Potential Field», *IEEE Transactions on Robotics*, vol. 33, n° 5, doi :10.1109/TRO.2017.2705034, p. 1124–1138, ISSN 1552-3098. [88](#)
- MAO, X., Y. XU, S. MITA, H. CHIN et H. TEHRANI. 2018, «Navigating Automated Vehicle through Expressway Toll Gate», dans *2018 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5386-4452-2, p. 814–819, doi :10.1109/IVS.2018.8500478. [15](#), [30](#)
- MARGELLOS, K. et J. LYGEROS. 2010, «Hamilton-Jacobi formulation for reach-avoid problems with an application to air traffic management», dans *Proceedings of the 2010 American Control Conference*, IEEE, ISBN 978-1-4244-7427-1, p. 3045–3050, doi :10.1109/ACC.2010.5530514. [84](#)
- MARTIN, A. 2013, «Interactive Motion Prediction using Game Theory», Master Thesis, University of Padua. [17](#)
- MARTINEZ-GOMEZ, L. et T. FRAICHARD. 2008, «An efficient and generic 2D Inevitable Collision State-checker», dans *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-4244-2057-5, p. 234–241, doi :10.1109/IROS.2008.4650640. [90](#)
- MARTINEZ-GOMEZ, L. et T. FRAICHARD. 2009, «Collision avoidance in dynamic environments : An ICS-based solution and its comparative evaluation», dans *2009 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-4244-2788-8, p. 100–105, doi :10.1109/ROBOT.2009.5152536. [79](#), [90](#)
- MEIRING, G., H. MYBURGH, G. A. M. MEIRING et H. C. MYBURGH. 2015, «A Review of Intelligent Driving Style Analysis Systems and Related Artificial Intelligence Algorithms», *Sensors*, vol. 15, n° 12, doi :10.3390/s151229822, p. 30 653–30 682, ISSN 1424-8220. [19](#)
- MESSAOUD, K., I. YAHIAOUI, A. VERROUST-BLONDET et F. NASHASHIBI. 2019, «Non-local social pooling for vehicle trajectory prediction», dans *2019 IEEE Intelligent Vehicles Symposium (IV)*. [25](#)
- MINDERHOUD, M. M. et P. H. BOVY. 2001, «Extended time-to-collision measures for road traffic safety assessment», *Accident Analysis & Prevention*, vol. 33, n° 1, doi :10.1016/S0001-4575(00)00019-1, p. 89–97, ISSN 0001-4575. [28](#)
- MINGUEZ, J., F. LAMIRAUX et J.-P. LAUMOND. 2008, «Motion Planning and Obstacle Avoidance», dans *Springer Handbook of Robotics*, Springer, Berlin, Heidelberg, p. 827–852, doi :10.1007/978-3-540-30301-5_36. [80](#), [99](#)
- MITCHELL, I., A. BAYEN et C. TOMLIN. 2005, «A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games», *IEEE Transactions on Automatic Control*, vol. 50, n° 7, doi :10.1109/TAC.2005.851439, p. 947–957, ISSN 0018-9286. [84](#)
- MORTENSON, M. E. 1997, *Geometric modeling*, John Wiley & Sons, Inc., ISBN 0471129577, 523 p.. [100](#)
- MORTON, J. et M. J. KOCHENDERFER. 2017, «Simultaneous policy learning and latent state inference for imitating driver behavior», dans *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-5386-1526-3, p. 1–6, doi :10.1109/ITSC.2017.8317738. [18](#)

- NAK YONG KO et BUM HEE LEE. 1996, «Avoidability measure in moving obstacle avoidance problem and its use for robot motion planning», dans *1996 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, IEEE, ISBN 0-7803-3213-X, p. 1296–1303, doi : 10.1109/IROS.1996.568984. 91
- NILSSON, J. et J. SJOBERG. 2013, «Strategic decision making for automated driving on two-lane, one way roads using model predictive control», dans *2013 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4673-2755-8, p. 1253–1258, doi :10.1109/IVS.2013.6629638. 16
- NILSSON, N. J. 1969, «A Mobile Automaton : An Application of Artificial Intelligence Techniques», 1st International Joint Conference on Artificial Intelligence (IJCAI). 80
- PADEN, B., M. CAP, S. Z. YONG, D. YERSHOV et E. FRAZZOLI. 2016, «A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles», *IEEE Transactions on Intelligent Vehicles*, vol. 1, n° 1, doi :10.1109/TIV.2016.2578706, p. 33–55, ISSN 2379-8904. 80
- PETRICH, D., T. DANG, D. KASPER, G. BREUEL et C. STILLER. 2013, «Map-based long term motion prediction for vehicles in traffic environments», dans *2013 16th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-4799-2914-6, p. 2166–2172, doi : 10.1109/ITSC.2013.6728549. 24
- PETTI, S. et T. FRAICHARD. 2005, «Safe motion planning in dynamic environments», dans *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 0-7803-8912-3, p. 2210–2215, doi :10.1109/IROS.2005.1545549. 90
- PITRON, G. 2018, *La guerre des métaux rares : la face cachée de la transition énergétique et numérique*, Les liens qui libèrent, ISBN 9791020905789, 294 p.. 3
- PIVTORAIKO, M. et A. KELLY. 2005, «Efficient constrained path planning via search in state lattices», dans *International Symposium on Artificial Intelligence, Robotics, and Automation in Space*. 86
- POLYCHRONOPOULOS, A., M. TSOGAS, A. AMDITIS, U. SCHEUNERT, L. ANDREONE et F. TANGO. 2004, «Dynamic situation and threat assessment for collision warning systems : the Euclidean approach», dans *2004 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 0-7803-8310-9, p. 636–641, doi :10.1109/IVS.2004.1336458. 27
- QUEHL, J., H. HU, S. WIRGES et M. LAUER. 2018, «An Approach to Vehicle Trajectory Prediction Using Automatically Generated Traffic Maps», dans *2018 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-5386-4452-2, p. 544–549, doi :10.1109/IVS.2018.8500535. 24
- QUINLAN, S. et O. KHATIB. 1993, «Elastic bands : connecting path planning and control», dans *[1993] Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, IEEE Comput. Soc. Press, ISBN 0-8186-3450-2, p. 802–807, doi :10.1109/ROBOT.1993.291936. 17
- QUINTERO M., C. G., J. O. LOPEZ et A. C. CUERVO PINILLA. 2012, «Driver behavior classification model based on an intelligent driving diagnosis system», dans *2012 15th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-4673-3063-3, p. 894–899, doi :10.1109/ITSC.2012.6338727. 19
- RASEKHIPOUR, Y., A. KHAJEPOUR, S.-K. CHEN et B. LITKOUHI. 2017, «A Potential Field-Based Model Predictive Path-Planning Controller for Autonomous Road Vehicles», *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, n° 5, doi :10.1109/TITS.2016.2604240, p. 1255–1267, ISSN 1524-9050. 16
- REEDS, J. A. et L. A. SHEPP. 1990, «Optimal paths for a car that goes both forwards and backwards», *Pacific Journal of Mathematics*, vol. 145, n° 2, p. 367–393. 83

- REGELE, R. 2008, «Using Ontology-Based Traffic Models for More Efficient Decision Making of Autonomous Vehicles», dans *Fourth International Conference on Autonomic and Autonomous Systems (ICAS'08)*, IEEE, ISBN 978-0-7695-3093-2, p. 94–99, doi :10.1109/ICAS.2008.10. 14
- REICHARDT, D. et J. SHICK. 1994, «Collision avoidance in dynamic environments applied to autonomous vehicle guidance on the motorway», dans *Proceedings of the Intelligent Vehicles '94 Symposium*, IEEE, ISBN 0-7803-2135-9, p. 74–78, doi :10.1109/IVS.1994.639475. 16
- ROESMANN, C., W. FEITEN, T. WOESCH, F. HOFFMANN et T. BERTRAM. 2012, «Trajectory modification considering dynamic constraints of autonomous robots», dans *ROBOTIK 2012; 7th German Conference on Robotics*, [VDE Verlag], ISBN 9783800734184. 17
- ROSSETTER, E. J. 2003, «A potential field framework for active vehicle lanekeeping assistance», PhD thesis. department of Mechanical engineering, Stanford University. 16
- SADIGH, D., S. SASTRY, S. A. SESHIA et A. D. DRAGAN. 2016, «Planning for Autonomous Cars that Leverage Effects on Human Actions», dans *Robotics : Science and Systems XII*, Robotics : Science and Systems Foundation, ISBN 9780992374723, doi :10.15607/RSS.2016.XII.029. 18
- SAMA, K., Y. MORALES, N. AKAI, H. LIU, E. TAKEUCHI et K. TAKEDA. 2018, «Driving Feature Extraction and Behavior Classification Using an Autoencoder to Reproduce the Velocity Styles of Experts», dans *2018 21st IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-7281-0321-1, p. 1337–1343, doi :10.1109/ITSC.2018.8569245. 19
- SATTEL, T. et T. BRANDT. 2005, «Ground vehicle guidance along collision-free trajectories using elastic bands», dans *2005 Annual American Control Conference (ACC)*, IEEE, ISBN 0-7803-9098-9, p. 4991–4996, doi :10.1109/ACC.2005.1470798. 17
- SCHEUER, A. et T. FRAICHARD. 1997, «Continuous-curvature path planning for car-like vehicles», dans *1997 IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS)*, vol. 2, IEEE, ISBN 0-7803-4119-8, p. 997–1003, doi :10.1109/IROS.1997.655130. 83
- SCHUBERT, R., E. RICHTER et G. WANIELIK. 2008, «Comparison and evaluation of advanced motion models for vehicle tracking», *Information Fusion, 2008 11th International Conference on*, , n° 1, doi :10.1109/ICIF.2008.4632283, p. 1–6. 23, 50
- SEDER, M. et I. PETROVIC. 2007, «Dynamic window based approach to mobile robot motion control in the presence of moving obstacles», dans *2007 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 1-4244-0602-1, ISSN 1050-4729, p. 1986–1991, doi : 10.1109/ROBOT.2007.363613. 90
- SEZER, V., T. BANDYOPADHYAY, D. RUS, E. FRAZZOLI et D. HSU. 2015, «Towards autonomous navigation of unsignalized intersections under uncertainty of human driver intent», dans *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-4799-9994-1, p. 3578–3585, doi :10.1109/IROS.2015.7353877. 13
- SHALEV-SHWARTZ, S., S. SHAMMAH et A. SHASHUA. 2016, «Safe, Multi-Agent, Reinforcement Learning for Autonomous Driving», *CoRR*, vol. abs/1610.03295. 18
- SONG, W., G. XIONG et H. CHEN. 2016, «Intention-Aware Autonomous Driving Decision-Making in an Uncontrolled Intersection», *Mathematical Problems in Engineering*, vol. 2016, doi :10.1155/2016/1025349, p. 1–15, ISSN 1024-123X. 13
- SORSTEDT, J., L. SVENSSON, F. SANDBLOM et L. HAMMARSTRAND. 2011, «A New Vehicle Motion Model for Improved Predictions and Situation Assessment», *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, n° 4, doi :10.1109/TITS.2011.2160342, p. 1209–1219, ISSN 1524-9050. 24

- STREUBEL, T. et K. H. HOFFMANN. 2014, «Prediction of driver intended path at intersections», dans *2014 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4799-3638-0, p. 134–139, doi : 10.1109/IVS.2014.6856508. [23](#)
- SUMMERS, S. et J. LYGEROS. 2010, «Verification of discrete time stochastic hybrid systems : A stochastic reach-avoid decision problem», *Automatica*, vol. 46, n° 12, doi :10.1016/J.AUTOMATICA.2010.08.006, p. 1951–1961, ISSN 0005-1098. [88](#)
- SUNKARA, V., A. CHAKRAVARTHY et D. GHOSE. 2019, «Collision Avoidance of Arbitrarily Shaped Deforming Objects using Collision Cones», *IEEE Robotics and Automation Letters*, doi :10.1109/LRA.2019.2900535, p. 1–1, ISSN 2377-3766. [86](#)
- TREIBER, M., A. HENNECKE et D. HELBING. 2000, «Congested Traffic States in Empirical Observations and Microscopic Simulations», *Physical Review E*, vol. 62, n° 2, doi :10.1103/PhysRevE.62.1805, ISSN 1063-651X. [51](#)
- TSOURVELOUDIS, N., K. VALAVANIS et T. HEBERT. 2001, «Autonomous vehicle navigation utilizing electrostatic potential fields and fuzzy logic», *IEEE Transactions on Robotics and Automation*, vol. 17, n° 4, doi :10.1109/70.954761, p. 490–497, ISSN 1042296X. [91](#)
- VAN DEN BERG, J., MING LIN et D. MANOCHA. 2008, «Reciprocal Velocity Obstacles for real-time multi-agent navigation», dans *2008 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-4244-1646-2, p. 1928–1935, doi :10.1109/ROBOT.2008.4543489. [87](#)
- VASQUEZ, D. et T. FRAICHARD. 2004, «Motion prediction for moving objects : a statistical approach», dans *2004 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 0-7803-8232-3, p. 3931–3936 Vol.4, doi :10.1109/ROBOT.2004.1308883. [23](#)
- VASQUEZ, D., T. FRAICHARD et C. LAUGIER. 2009, «Incremental Learning of Statistical Motion Patterns With Growing Hidden Markov Models», *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, n° 3, doi :10.1109/TITS.2009.2020208, p. 403–416, ISSN 1524-9050. [24](#)
- VASQUEZ, D., F. LARGE, T. FRAICHARD et C. LAUGIER. 2004, «High-speed autonomous navigation with motion prediction for unknown moving obstacles», dans *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, IEEE, ISBN 0-7803-8463-6, p. 82–87, doi :10.1109/IROS.2004.1389333. [23](#), [88](#)
- VOGEL, K. 2003, «A comparison of headway and time to collision as safety indicators», *Accident Analysis & Prevention*, vol. 35, n° 3, doi :10.1016/S0001-4575(02)00022-2, p. 427–433, ISSN 0001-4575. [27](#), [28](#), [29](#)
- WANG, J. et R. RAJAMANI. 2004, «Should Adaptive Cruise-Control Systems be Designed to Maintain a Constant Time Gap Between Vehicles?», *IEEE Transactions on Vehicular Technology*, vol. 53, n° 5, doi :10.1109/TVT.2004.832386, p. 1480–1490, ISSN 0018-9545. [27](#)
- WERLING, M., S. KAMMEL, J. ZIEGLER et L. GRÖLL. 2012, «Optimal trajectories for time-critical street scenarios using discretized terminal manifolds», *The International Journal of Robotics Research*, vol. 31, n° 3, doi :10.1177/0278364911423042, p. 346–359, ISSN 0278-3649. [86](#)
- WILKIE, D., J. VAN DEN BERG et D. MANOCHA. 2009, «Generalized velocity obstacles», dans *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-4244-3803-7, p. 5573–5578, doi :10.1109/IROS.2009.5354175. [79](#), [87](#)
- WOLF, M. T. et J. W. BURDICK. 2008, «Artificial potential functions for highway driving with collision avoidance», dans *2008 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, ISBN 978-1-4244-1646-2, p. 3731–3736, doi :10.1109/ROBOT.2008.4543783. [16](#)

- XIN, L., P. WANG, C.-Y. CHAN, J. CHEN, S. E. LI et B. CHENG. 2018, «Intention-aware Long Horizon Trajectory Prediction of Surrounding Vehicles using Dual LSTM Networks», dans *2018 21st IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-7281-0321-1, p. 1441–1446, doi :10.1109/ITSC.2018.8569595. 25
- YAN, Y., X. ZHU, M. LIAO et Z. MA. 2018, «Danger Degree Evaluation Method of Crossroad Scenario Based on Naturalistic Driving Analysis», dans *2018 21st IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-7281-0321-1, p. 1435–1440, doi : 10.1109/ITSC.2018.8569328. 29
- YOO, J. H. et R. LANGARI. 2013, «A Stackelberg Game Theoretic Driver Model for Merging», dans *Intelligent Transportation Systems*, ASME, doi :10.1115/DSCC2013-3882. 17
- ZHANG, Q., D. FILEV, H. E. TSENG, S. SZWABOWSKI et R. LANGARI. 2018, «Addressing Mandatory Lane Change Problem with Game Theoretic Model Predictive Control and Fuzzy Markov Chain», dans *2018 Annual American Control Conference (ACC)*, IEEE, ISBN 978-1-5386-5428-6, p. 4764–4771, doi :10.23919/ACC.2018.8431530. 16
- ZHAO, L., R. ICHISE, S. MITA et Y. SASAKI. 2015a, «An Ontology-Based Intelligent Speed Adaptation System for Autonomous Cars», dans *Semantic Technology*, Springer International Publishing, Cham, p. 397–413, doi :10.1007/978-3-319-15615-6_30. 14
- ZHAO, L., R. ICHISE, T. YOSHIKAWA, T. NAITO, T. KAKINAMI et Y. SASAKI. 2015b, «Ontology-based decision making on uncontrolled intersections and narrow roads», dans *2015 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, ISBN 978-1-4673-7266-4, p. 83–88, doi :10.1109/IVS.2015.7225667. 14
- ZHOU, Y. et J. S. BARAS. 2015, «Reachable set approach to collision avoidance for UAVs», dans *2015 54th IEEE Conference on Decision and Control (CDC)*, IEEE, ISBN 978-1-4799-7886-1, p. 5947–5952, doi :10.1109/CDC.2015.7403154. 84
- ZHU, T., J. XU, Y. BAI et X. YANG. 2009, «A research on risk assessment and warning strategy for intersection collision avoidance system», dans *2009 12th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, ISBN 978-1-4244-5519-5, p. 1–6, doi : 10.1109/ITSC.2009.5309864. 28
- ZIEGLER, J. et C. STILLER. 2009, «Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios», dans *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, ISBN 978-1-4244-3803-7, p. 1879–1884, doi :10.1109/IROS.2009.5354448. 86

Annexe A

Liste des acronymes

ADAS système d'aide à la conduite automobile (Advanced Driver-Assistance Systems). Ce terme est notamment employé pour des fonctionnalités avancées des voitures équipées telles que la détection de piétons, l'alerte de franchissement involontaire de ligne, ou l'arrêt d'urgence en présence d'un obstacle. Cependant sa définition inclut également les fonctionnalités plus simples comme l'ABS ou le régulateur de vitesse. [14](#)

MPC commande prédictive (*Model Predictive Control*), technique de commande à modèle interne consistant à utiliser une modélisation dynamique du processus en sortie à l'intérieur du module qui prend la décision en amont. Dans le cadre de la planification de trajectoire, cela consiste à intégrer une modélisation du contrôleur bas niveau dans la prédiction de mouvement du véhicule. [11](#), [44](#), [54](#)

PET délai après croisement, ou *Post-Encroachment Time*. [27](#), [30](#), [37](#), [51](#)

TG intervalle de temps entre les trajectoires, ou *Time Gap*. Il est défini pour toute paire de trajectoires dont les chemins se croisent. [27](#), [30](#), [51](#), [69](#)

TTC laps de temps avant qu'une collision n'advienne, ou *Time To Collision*. Il est défini pour une paire de trajectoires qui mène à une collision. [13](#), [27](#), [89](#)

VA véhicule automatisé, qui exécute l'algorithme principal de planification. [11](#), [44](#), [48](#)

Annexe B

Glossaire

carrefour giratoire carrefour particulier formé d'un anneau central qui permet aux usagers de prendre n'importe quelle direction, y compris de faire un demi-tour. La priorité s'exerce à gauche, elle est donnée aux véhicules circulant sur l'anneau. Bien que l'abus de langage soit courant, il ne faut pas confondre un carrefour giratoire avec un rond-point. [37](#)

continuité G1 la continuité G1 entre deux courbes, ou tangence géométrique, est leur propriété de partager une même tangente en leur point de contact. Cette propriété est équivalente à une continuité C1 de la courbe en ajoutant comme condition l'absence de point de rebroussement. [33](#), [37](#), [50](#), [102](#)

environnement semi-contraint type d'environnement de conduite particulier dans lequel l'espace navigable est très large et le marquage au sol quasi inexistant, de sorte qu'il n'est pas possible de spécifier une topologie de route vraiment stricte. On peut par exemple trouver ce type d'environnement sur certaines larges intersections parisiennes (place de la Bastille, place de l'Étoile), au niveau des péages d'autoroutes, sur certains parkings où le marquage au sol est absent ou encore lors de travaux. [11](#), [47](#)

holonome Un robot holonome, ou robot omni-directionnel, doit être capable de se déplacer dans n'importe quelle direction, quelle que soit son orientation (i.e. capable de se déplacer « en crabe »). En effet, un robot est qualifié d'holonome lorsqu'il possède autant de degrés de liberté de déplacement qu'il est possible d'en obtenir dans l'espace dans lequel il se déplace. Un robot se déplaçant sur un plan doit donc posséder trois degrés de liberté : les deux axes du plan ainsi que son orientation (ou angle de lacet). À l'inverse, une voiture ne possède que deux degrés de liberté : son déplacement latéral est lié à sa rotation. [81](#)

rond-point place circulaire (ronde, ovale ou polygonale) qui permet aux usagers de prendre n'importe quelle direction, y compris de faire un demi-tour. La priorité s'exerce à droite, elle est donnée aux véhicules qui arrivent sur l'anneau. En France, les ronds-points sont en réalité assez rares sauf dans Paris (pour n'en citer qu'un : celui de la place de l'Étoile). Bien que l'abus de langage soit courant, il ne faut pas confondre un rond-point avec un carrefour giratoire. [48](#), [59](#)