

N° Ordre..... /DISM/FT/UMBB/2023

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE M'HAMED BOUGARA - BOUMERDES



Faculté de Technologie

Département Ingénierie des Systèmes Electriques en collaboration avec

Le Centre de Développement des Technologies Avancées

Mémoire de Master

Réalisé par :

KEBIR Anis & KEBIR Kahina

THEME:

Aide à la conduite pour un système de transport

Filière: Télécommunications

Spécialité: Réseaux et Télécommunications

Soutenu le 13/ 07/ 2023 devant le jury composé de:

AKLIOUAT	Hacene	UMBB	Président
BOUMERIDJA	Hafida	UMBB	Examinatrice
BENMOUSSA	Yahia	UMBB	Promoteur
BOURAINÉ	Sara	CDTA	Encadreur

Année Universitaire : 2022/2023



Remerciement

Tout d'abord, nous souhaitons exprimer notre profonde gratitude à Dieu pour nous avoir donné la force, la vie et la santé, qui nous ont permis de mener à bien ce travail.

Nous tenons à adresser nos sincères remerciements à toute l'équipe du CDTA pour avoir accepté de collaborer avec nous dans le cadre de notre projet de fin d'études. Votre accueil chaleureux et votre précieux soutien ont grandement contribué à notre réussite.

*Nous aimerions également exprimer notre reconnaissance envers Monsieur le Directeur du Centre de Développement et de Technologies Avancées, ainsi que toute l'équipe du Département Robotique (en particulier **Mme (Dr). BOURAINE**) du CDTA, pour leur encadrement de qualité, leur rigueur, leur disponibilité, leurs remarques fructueuses et leurs directives précieuses, qui ont efficacement contribué à l'avancement de ce travail. Nous adressons également nos remerciements à **Monsieur (Dr). BENMOUSSA** de l'Université de Technologie M'HAMED BOUGARA - BOUMERDES -, pour avoir accepté de nous encadrer et pour ses conseils précieux tout au long de la réalisation de ce projet.*

*Nous saisissons également cette occasion pour exprimer notre gratitude envers tous les membres du jury pour avoir honoré notre travail de leur présence et pour avoir accepté de l'évaluer. Nous n'oublions pas **Monsieur (Dr). MESSAOUDI** pour ses efforts au sein du Département Génie Électrique durant notre cursus, ainsi que tous les enseignants du département Réseaux et Télécommunications (RT).*

Encore une fois, merci du fond du cœur à tous ceux qui ont contribué de près ou de loin à la réalisation de ce projet. Votre soutien a été essentiel et nous en sommes extrêmement reconnaissants.

Dédicace

Avec l'expression de ma reconnaissance,

J'ai l'immense Plaisir de dédier ce travail à

*A mon cher **père**,*

Qu'Allah t'offre une vie longue pour être mon soutien dans cette vie.

*A ma très chère **mère**,*

Je veux te dire que je t'aime tellement et qu'Allah te protège pour nous.

*Mon **papa** et ma **maman**, ce travail est le fruit de vos sacrifices que vous avez*

Consentis pour mon éducation et mon avenir,

*À ma chère tante **Fatiha**, tu es une source de lumière qui éclaire nos vies avec ta
bonté et ta bienveillance.*

*À mon cher oncle **Amer**, qui a été un pilier solide dans ma vie. Malgré son
absence physique, son esprit m'accompagne aujourd'hui et chaque jour. Merci à
toi, mon oncle, pour tout l'amour et l'inspiration que tu m'as donnés. Ton souvenir
restera vivant dans mon cœur pour toujours.*

A mes chères sœurs et chers frère, avec tous mes souhaits

de succès dans leur vie.

*Sans oublier mon binôme, ma sœur **KAHINA**, pour son soutien moral, sa*

Patience et sa compréhension tout au long de ce projet.

Merci !

ANIS

Dédicace

Avec l'expression de ma reconnaissance,

J'ai l'immense Plaisir de dédier ce travail à

*A mon cher **père**,*

Qu'Allah t'offre une vie longue pour être mon soutien dans cette vie.

*A ma très chère **mère**,*

Je veux te dire que je t'aime tellement et qu'Allah te protège pour nous.

*Mon **papa** et ma **maman**, ce travail est le fruit de vos sacrifices que vous avez*

Consentis pour mon éducation et mon avenir,

*À ma chère tante **Fatiha**, tu es une source de lumière qui éclaire nos vies avec ta bonté et ta bienveillance.*

*À mon cher oncle **Amer**, qui a été un pilier solide dans ma vie. Malgré son absence physique, son esprit m'accompagne aujourd'hui et chaque jour. Merci à toi, mon oncle, pour tout l'amour et l'inspiration que tu m'as donnés. Ton souvenir restera vivant dans mon cœur pour toujours.*

A mes chères sœurs et chers frère, avec tous mes souhaits

de succès dans leur vie.

*Sans oublier mon binôme, mon frère **ANIS**, pour son soutien moral, sa*

Patience et sa compréhension tout au long de ce projet.

Merci !

KAHINA

Table des matières

Remerciement.....	2
Dédicace	3
Liste des figures.....	9
Liste des tableaux.....	11
Résumé	12
Introduction générale	14
Chapitre I : Systèmes anti-collision	
I. 1 Introduction	16
I. 2 Définition d'un système anti-collision.....	16
I. 3 Objectifs des systèmes anti-collision	16
I. 4 Principe de fonctionnement des systèmes anti-collision :.....	17
I. 4. 1 Capteurs et détection des obstacles :	17
I. 4. 1. 1 Radar (Radio Detection and Ranging).....	18
I. 4. 1. 2 Lidar (Light Detection and Ranging).....	18
I. 4. 1. 3 Caméras.....	18
I. 4. 1. 4 Capteurs ultrasoniques	18
I. 4. 2 Traitement des données et prise de décision.....	18
I. 4. 2. 1 Acquisition des données.....	18
I. 4. 2. 2 Prétraitement des données.....	19
I. 4. 2. 3 Fusion des données	19
I. 4. 2. 4 Analyse des données	19
I. 4. 2. 5 Prise de décision.....	19
I. 4. 2. 6 Mise à jour en temps réel	19
I. 4. 3 Systèmes de contrôle et d'alerte.....	20
I. 4. 3. 1 Contrôle préventif	20
I. 4. 3. 2 Alertes visuelles	20
I. 4. 3. 3 Alertes sonores	20
I. 5 Applications des systèmes anti-collision	20
I. 5. 1 Systèmes anti-collision dans les véhicules autonomes.....	21
I. 5. 1. 1 Freinage automatique d'urgence (AEB - Automatic Emergency Braking).....	21
I. 5. 1. 2 Le régulateur de vitesse adaptatif (ACC - Adaptive Cruise Control)	22
I. 5. 1. 3 Avertissement de sortie de voie (LDW - Lane Departure Warning)	22
I. 5. 1. 4 Avertissement de collision frontale (FCW - Forward Collision Warning).....	22
I. 5. 1. 5 Surveillance des angles morts (BSM - Blind Spot Monitoring)	22
I. 5. 2 Systèmes anti-collision dans l'aviation	22
I. 5. 2. 1 Système d'alerte de trafic et d'évitement de collision (TCAS - Traffic Alert and Collision Avoidance System)	23

I. 5. 2. 2 Système d'alerte de proximité du sol (GPWS - Ground Proximity Warning System).....	23
I. 5. 3 Systèmes anti-collision dans le transport ferroviaire.....	23
I. 5. 3. 1 Système de contrôle des trains (CBTC - Communications-Based Train Control) :	24
I. 5. 3. 2 Système d'arrêt automatique des trains (ATS - Automatic Train Stop) :	24
I. 5. 3. 3 Système d'alerte de collision imminente (ICA - Imminent Collision Alert)...	24
I. 5. 4 Systèmes anti-collision dans le transport maritime :	24
I. 5. 4. 1 Système d'identification automatique (AIS - Automatic Identification System) :.....	25
I. 5. 4. 2 Système d'alerte de proximité (CPA - Closest Point Approach) :	25
I. 5. 5 Systèmes anti-collision dans le domaine de la robotique :	25
I. 5. 5. 1 Evitement basé sur les capteurs :	26
I. 5. 5. 2 Systèmes de sécurité intégrés :	26
I. 6 Etude des systèmes anti-collision dans le transport ferroviaire	26
I. 6. 1 Les facteurs contribuant aux accidents de tramway	26
I. 6. 2 Système avancé d'assistance à la conduite de tramway :.....	27
I. 6. 3 La possibilité de transfert de la technologie ADAS - De la voiture au tramway- ..	27
I. 6. 4 Technologies et capteurs pour l'amélioration de l'automatisation dans les transports en commun : De niveau d'automatisation actuel (GoA 0) à un niveau supérieur (GoA 1)31	
I. 6. 5 Les types de Capteurs dans les transports en commun.....	31
I. 6. 6 Intégration des systèmes avancés d'aide à la conduite dans les tramways :.....	33
I. 6. 7 Les caractéristiques principales de tramway :	33
I. 7 Conclusion :.....	35

Chapitre II : Système d'aide à la conduite : méthodologie et conception

II. 1 Introduction.....	36
II. 2 Approches de navigation réactive	36
II. 3 L'étude des approches de navigation réactive	36
II. 3. 1 Les approches réactives.....	37
II. 3. 1. 1 La fenêtre dynamique (DW)	37
II. 3. 1. 2 L'approche des champs de potentiel (PF)	37
II. 3. 1. 3 Obstacle vitesse (VO)	38
II. 3. 1. 4 Algorithmes de Bugs.....	38
II. 3. 1. 5 Navigation basée sur les états de collision inévitable (ICS).....	39
II. 3. 2 Planification réactive	39
II. 3. 2. 1 Les techniques par échantillonnage	39
II.3.2.1.1 Échantillonnage de l'espace d'entrées (ISS)	40
II.3.2.1.2 L'échantillonnage de l'espace d'états (SSS)	40
II. 4 La Solution proposée : Safe ISS	41
II. 4. 1 Perception et modélisation de l'environnement	41
II. 4. 1. 1 Espace de configuration.....	41
II. 4. 1. 2 Modélisation des données des capteurs	42
II. 4. 2 Echantillonnage de l'espace local du robot.....	42

II. 4. 2. 1 La génération d'un ensemble de trajectoires dans l'espace de recherche.....	44
II. 4. 3 Profil des vitesses	45
II. 4. 3. 1 Profil de vitesse Linéaire	46
II. 4. 3. 2 Profil de vitesse Angulaire.....	46
II. 4. 4 Modèle odométrique.....	47
II. 5 La sûreté du mouvement	48
II. 6 Les critères de la sûreté du mouvement :.....	49
II. 6. 1 Critère 1 : Contraintes liées au mouvement du robot.....	49
II. 6. 2 Critère 2 : Raisonnement sur le futur	50
II. 6. 3 Critère 3 : Un horizon temporel approprié	51
II. 7 Vérifier la sûreté de mouvement	52
II. 8 Le système d'aide à la conduite	53
II. 9 Conclusion :	54
 Chapitre III : Simulation et résultats	
III. 1 Introduction.....	55
III. 2 ROS (Robot Operating System)	55
III. 2. 1 Historique de ROS	55
III. 2. 2 Architecture de ROS	55
III. 2. 3 Composants clés de ROS	56
III. 2. 4 Communication et gestion des nœuds.....	56
III. 2. 5 Bibliothèques et outils intégrés à ROS	56
III. 2. 6 Avantages de l'utilisation de ROS dans la robotique.....	56
III. 3 Gazebo (simulateur).....	57
III. 3. 1 Fonctionnalités de Gazebo	57
III. 3. 2 Modélisation de robots dans Gazebo	58
III. 3. 3 Environnements de simulation réalistes.....	58
III. 3. 4 Intégration de ROS avec Gazebo	58
III. 3. 5 Avantages de l'utilisation de Gazebo pour la simulation robotique	58
III. 4 Résultat de l'implémentation du système d'aide à la conduite.....	59
III. 4. 1 Cas d'un robot mobile différentiel	59
III. 4. 1. 1 Le model cinématique du robot.....	59
III. 4. 1. 2 Modélisation	60
III. 4. 1. 3 La génération de trajectoire	62
III. 4. 2 Scénario de test	64
III. 4. 2. 1 Simulation des zones de perception du capteur laser	64
III. 4. 3 Navigation sans obstacle	66
III. 4. 3. 1 Détection d'un obstacle grâce au capteur laser.....	66
III. 4. 4 Détection et évitement de plusieurs obstacles.....	67
a. Détection.....	67
b. Evitement.....	68

III. 5 Le système d'aide à la conduite	71
III. 6 Cas d'un tramway	73
III. 6. 1. 1 Le modèle cinématique du tramway	73
III. 6. 1. 2 Description	73
III. 6. 1. 3 Modélisation	74
III. 6. 1. 4 Navigation sans obstacle:	74
III. 6. 1. 5 Détection et évitement des obstacles	75
III. 6. 1. 6 Navigation avec obstacle	75
III. 7 Conclusion	78
Conclusion générale.....	79
Bibliographie	80

Liste des figures

Figure I-1 Principe de fonctionnement des systèmes anti-collision.....	17
Figure I-2 Applications des systèmes anti-collision.....	21
Figure I-3 Systèmes anti-collision dans les véhicules automobiles	21
Figure I-4 Systèmes anti-collision dans l'aviation.....	22
Figure I-5 Systèmes anti-collision dans Transport ferroviaire	23
Figure I-6 Systèmes anti-collision dans le transport maritime.....	25
Figure I-7 Systèmes anti-collision dans le domaine des robots	26
Figure I-8 La détection des piétons et la surveillance de la voie ferrée	27
Figure I-9 Répartition technique et configuration des capteurs du tramway"	33
Figure II-1 L'étude des approches de navigation	36
Figure II-2 L'approche de la fenêtre dynamique.	37
Figure II-3 Champs de potentiel pour un environnement contenant trois obstacles.	38
Figure II-4 L'approche VO.....	38
Figure II-5 Calcul du chemin du robot par les algorithmes Bug: (a) Bug1, (b) Bug2	39
Figure II-6 Échantillonnage de trajectoire entre espace de contrôle et espace d'état pour un véhicule sur une route.....	40
Figure II-7 Espace de configuration	42
Figure II-8 Echantillonnage de l'espace local du robot(Le robot contient 15 zones de détection)	43
Figure II-9 Ensemble de trajectoires – Affichage sur RVIZ (sous ROS).....	44
Figure II-10 Principe de fonctionnement de SafeISS.....	48
Figure II-11 Les critères de la sûreté du mouvement	49
Figure II-12 Illustration de l'impact de la dynamique d'un robot (masse ponctuelle) sur sa sécurité.....	50
Figure II-13 L'influence de la dynamique d'un robot et celle des obstacles mobiles sur la sûreté de mouvement d'un robot.	51
Figure II-14 Sélection trajectoire de freinage.....	53
Figure III-1 Architecture de ROS	55
Figure III-2 Robot mobile différentiel.....	59
Figure III-3 Le robot mobile dingo	59
Figure III-4 centre instantanée de rotation CIR d'un robot mobile différentiel	61
Figure III-5 Simulation de génération des trajectoires sur RVIZ avec des sphères List points	63

Liste des figures

Figure III-6 Scénario de test dans le simulateur Gazebo.....	64
Figure III-7 Résultat de la simulation des zones de perception sur Gazebo.	65
Figure III-8 Résultat de la simulation des zones de perception sur Gazebo (un zoom sur le robot).	65
Figure III-9 Résultat de la simulation des zones de perception sur Gazebo(avec déplacement de robot).	66
Figure III-10 Détection d'un obstacle grâce au capteur laser	67
Figure III-11 Détection de deux obstacles par le robot	67
Figure III-12 Détection de deux obstacles par le robot (un zoom sur le robot).	68
Figure III-13 Le processus d'évitement des obstacles par le robot.....	71
Figure III-14 Simulation des trajectoires générées par la commande avant sur Rviz	72
Figure III-15 Simulation des trajectoires générées par la commande gauche sur Rviz	72
Figure III-16 Simulation des trajectoires générées par la commande droite sur Rviz	73
Figure III-17 Robot mobile omnidirectionnel	73
Figure III-18 le déplacement du robot d'un point initial vers un autre point à une vitesse constante.....	74
Figure III-19 un "zoom" sur le robot lorsqu'il est complètement arrêté.....	77
Figure III-20 Augmentation de la vitesse de robot et continue à se déplacer.....	78

Liste des tableaux

Tableau 1-1 La transférabilité de la technologie ADAS - De la voiture au tramway- 30

Tableau 1-2 L'intégration de capteurs sur un tramway..... 31

Tableau 1-3 Une étude des différents types de capteurs..... 32

Tableau 1-4 Les caractéristiques principales de tramway 35

Liste des abréviations

Liste des abréviations

ICS - Inevitable Collision State
ISS - Input Space Sampling
ROS - Robot Operating System
SSS - State Space Sampling
VO - Velocity Obstacle
DW - La fenêtre dynamique
PF - L'approche des champs de potentiel
Radar - Radio Detection and Ranging
Lidar - Light Detection and Ranging
AEB - Automatic Emergency Braking
ACC - Adaptive Cruise Control
LDW - Lane Departure Warning
FCW - Forward Collision Warning
BSM - Blind Spot Monitoring
TCAS - Traffic Alert and Collision Avoidance System
GPWS - Ground Proximity Warning System
CBTC - Communications-Based Train Control
ATS - Automatic Train Stop
ICA - Imminent Collision Alert
AIS - Automatic Identification System
CPA - Closest Point Approach
T-ADAS - Tramway Advanced Driver Assistance System
LKS - Lane Keeping System
TSR - Traffic Sign Recognition
BLIS - Blind Spot Information System
BSD - Blind Spot Detection
ALC - Adaptive Light Control
RCW - Rear Collision Warning
DMS - Driver Monitoring System
CTA - Cross Traffic Alert
PA - Park Assist

Résumé

Les systèmes de transport jouent un rôle crucial dans notre société moderne et revêtent une grande importance à plusieurs niveaux. Ce mémoire se concentre sur l'aide à la conduite dans les systèmes de transport, en mettant particulièrement l'accent sur les risques de collisions et la sécurité. L'objectif principal de ce projet est de concevoir un système d'aide à la conduite efficace et fiable, reposant sur des techniques avancées de perception et de navigation en robotique mobile. Il vise à développer un système d'aide à l'autonomie de déplacement qui garantit la sécurité tout en respectant les choix de l'utilisateur, en utilisant une méthode d'échantillonnage de l'espace de contrôles.

Mots clés : Sûreté de mouvement, Évitement d'obstacles, Aide à la conduite, ROS, Gazebo.

Abstract

Transportation systems play a crucial role in our modern society and hold significant importance at various levels. This work focuses on driver assistance in transportation systems, with a particular emphasis on collision risks and safety. The primary objective of this project is to design an efficient and reliable driver assistance system, based on advanced perception and navigation techniques in mobile robotics. The aim is to develop a system that aids in autonomous mobility while ensuring safety and respecting user choices, control space sampling method.

Keywords: Motion safety, Obstacle avoidance, Driver assistance, ROS, Gazebo.

الملخص

تلعب أنظمة النقل دورًا حاسمًا في مجتمعنا الحديث ولها أهمية كبيرة على عدة مستويات. تركز هذه الأطروحة على مساعدة السائق في أنظمة النقل، مع التركيز بشكل خاص على مخاطر الاصطدام والسلامة. الهدف الرئيسي من هذا المشروع هو تصميم نظام فعال وموثوق لمساعدة السائق، يعتمد على الإدراك المتقدم وتقنيات الملاحة في الروبوتات المتنقلة. ويهدف إلى تطوير نظام مساعدة في استقلالية الحركة يضمن السلامة مع احترام اختيارات المستخدم، وذلك بفضل عنصر تحكم باستخدام طريقة أخذ عينات من مساحة التحكم.

الكلمات المفتاحية: سلامة الحركة، تجنب العوائق، مساعدة القيادة، ROS، Gazebo.

Introduction générale

Ce n'est pas que je suis si intelligent, c'est que je reste plus longtemps avec les problèmes.

'Albert Einstein'

Introduction générale

Les avancées technologiques dans le domaine des transports ont révolutionné notre manière de nous déplacer, offrant des possibilités sans précédent en termes d'efficacité, de rapidité et de commodité. Cependant, ces progrès ont également posé de nouveaux défis en matière de sécurité et de fiabilité. L'évolution rapide des systèmes de transport a ouvert de nouvelles perspectives mais a également engendré de nombreux défis. Parmi ces défis figurent les risques de collisions, les défaillances techniques et les problèmes de communication qui peuvent compromettre la sécurité des systèmes de transport. Dans ce mémoire, nous nous concentrons plus particulièrement sur les risques de collisions et l'importance de garantir une conduite sûre et fiable.

Le développement de techniques avancées de perception et de navigation en robotique mobile offre des solutions potentielles pour résoudre ce problème critique. En exploitant ces techniques, il devient possible de concevoir un système d'aide à la conduite capable de détecter les obstacles, de prédire leur trajectoire et de prendre des décisions appropriées pour éviter les collisions.

Le travail se concentre sur le développement d'un système d'aide à la conduite utilisant des techniques avancées de perception et de navigation en robotique mobile.

Ce mémoire est organisé en plusieurs chapitres. Le chapitre 1 introduit les généralités sur le système anti-collision en mettant en évidence les principaux défis et en présentant les différentes approches existantes.

Le chapitre 2 porte sur le système d'aide à la conduite : conception et méthodologie. Nous examinons diverses techniques utilisées pour éviter les obstacles, en mettant l'accent sur leur impact sur la sécurité du système. Nous étudions les différentes composantes nécessaires à la mise en place d'un tel système, en soulignant les considérations essentielles liées à la détection des obstacles, et la prise de décision en temps réel.

Enfin, le chapitre 3 explore les outils de simulations basées sur ROS (Robot Operating System), une plateforme populaire dans le domaine de la robotique. Nous discutons sur les détails de l'implémentation des algorithmes développés et l'évaluation des performances du système d'aide à la conduite proposé.

En conclusion, ce mémoire vise à apporter une contribution significative dans le domaine de l'aide à la conduite pour les systèmes de transport. En combinant des techniques de perception et de navigation avancées, nous espérons améliorer la sécurité des systèmes de transport et réduire les risques de collisions.

Chapitre I : Systèmes anti-collision

I. 1 Introduction

Dans le domaine des transports, la sécurité occupe une place primordiale. En effet, les collisions potentielles véhicules, avions, trains ou autres moyens de transport, représentent un risque majeur pour la vie des passagers et la protection des biens. C'est pourquoi, les systèmes anti-collision jouent un rôle essentiel dans la prévention des accidents.

Un système anti-collision, comme son nom l'indique, est conçu pour détecter les obstacles obstruant la trajectoire d'un véhicule et prendre des mesures pour éviter les collisions. Cela est possible grâce à des capteurs adaptés, des algorithmes de traitement de données, des systèmes de contrôle, et surtout des algorithmes de raisonnement et de prise de décision.

I. 2 Définition d'un système anti-collision

Un système anti-collision, également connu sous le nom de système de prévention des collisions est un dispositif technologique conçu pour détecter la présence d'obstacles ou de dangers potentiels sur la trajectoire d'un véhicule et prendre des mesures pour éviter les collisions. Ces systèmes sont largement utilisés dans divers domaines tels que l'automobile, l'aviation, le ferroviaire, et d'autres moyens de transport, avec pour objectif principal d'assurer la sécurité des passagers et des biens [1].

I. 3 Objectifs des systèmes anti-collision

Les systèmes anti-collision ont plusieurs objectifs clés contribuant à assurer la sécurité des passagers, réduire les accidents et protéger l'environnement. Voici les principaux objectifs de ces systèmes :

1. **Sécurité des passagers** : l'objectif principal des systèmes anti-collision est d'assurer la sécurité des passagers. En détectant les obstacles sur la trajectoire du véhicule, cela réduit considérablement les risques de blessures ou de décès pour les personnes à bord ou à l'extérieur.
2. **Réduction des accidents** : les systèmes anti-collision visent à réduire le nombre d'accidents en prévenant le conducteur des situations dangereuses. La détection des obstacles et leur identification (véhicules, piétons, animaux ou d'autres objets) aident les conducteurs ou les opérateurs à prendre des bonnes décisions à temps pour éviter les collisions. Cela contribue à diminuer les risques d'accidents et à améliorer la sécurité routière, aérienne, ferroviaire, etc. (selon le système en question).

3. **Protection des biens** : les systèmes anti-collision ne sont pas seulement axés sur la sécurité des personnes, mais aussi sur la protection des biens. En évitant les collisions, ces systèmes contribuent à prévenir les dommages matériels aux véhicules, aux infrastructures et à d'autres biens impliqués dans les transports. En conséquence, cela réduit les coûts associés aux accidents, les réparations et les pertes de biens.

I. 4 Principe de fonctionnement des systèmes anti-collision :

Les systèmes anti-collision sont conçus pour garantir la sécurité en prévenant les collisions potentielles. Dans ce contexte, la figure I.1 présente les principes de fonctionnement de ces systèmes.

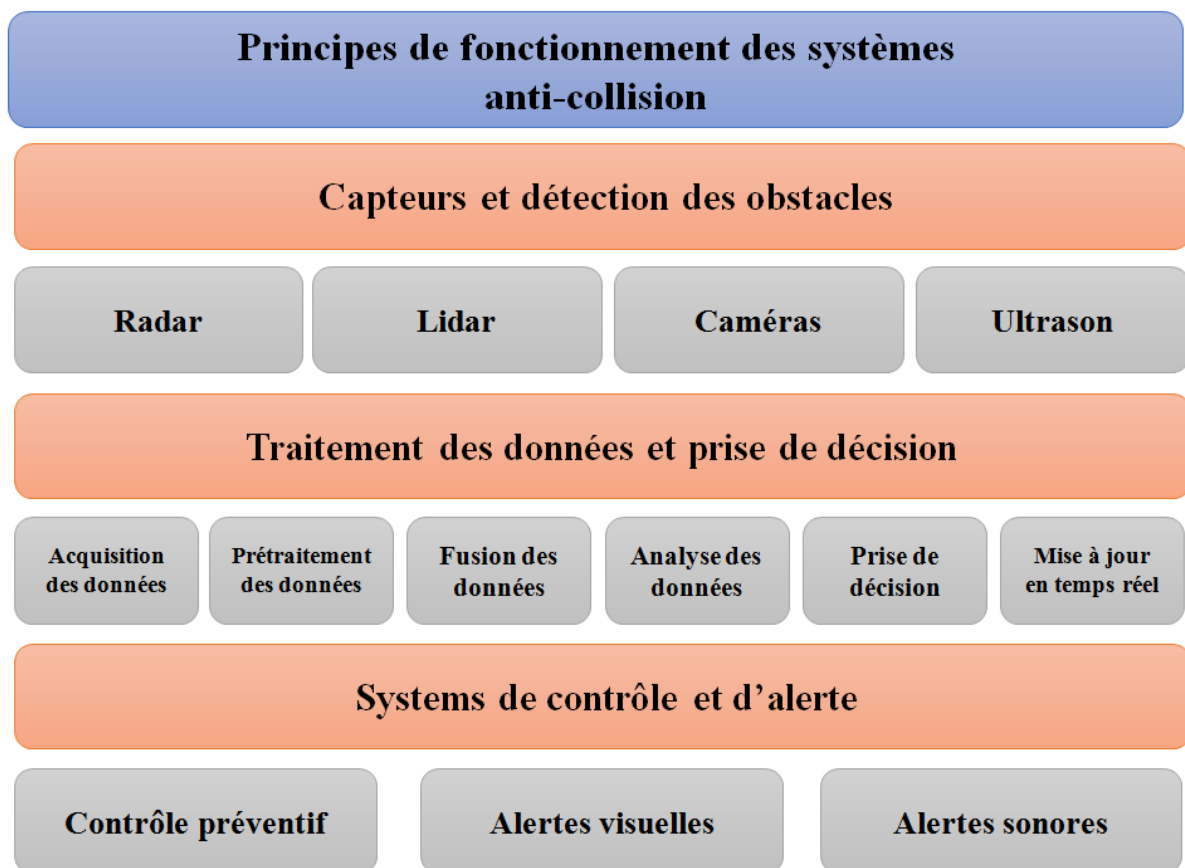


Figure I-1 Principe de fonctionnement des systèmes anti-collision

I. 4. 1 Capteurs et détection des obstacles :

Les systèmes anti-collision utilisent une variété de capteurs pour détecter les obstacles et collecter des données permettant la prédiction de leur position, leur vitesse et d'autres caractéristiques. Voici un aperçu des différents types de capteurs couramment utilisés pour ces systèmes :

I. 4. 1. 1 Radar (Radio Detection and Ranging)

Les capteurs radar émettent des ondes radio et mesurent le temps qu'il faut pour que ces ondes rebondissent sur les objets et reviennent au capteur. En analysant les changements dans les ondes réfléchies, le radar est utilisé pour mesurer la vitesse du système de transport [2].

I. 4. 1. 2 Lidar (Light Detection and Ranging)

Les capteurs lidar émettent des ondes lumineuses et mesurent le temps qu'il faut pour que ces ondes rebondissent sur les objets et reviennent au capteur. En mesurant la différence de temps et en analysant les propriétés de la lumière réfléchie, le lidar peut fournir une information de distance, à partir de laquelle peuvent être déduites la forme et la nature des obstacles [3].

I. 4. 1. 3 Caméras

Les caméras utilisent des systèmes d'imagerie pour capturer des images visuelles de l'environnement du système. Les algorithmes de traitement d'image peuvent ensuite être utilisés pour identifier et suivre les objets en se basant sur leur forme, leur couleur et d'autres caractéristiques visuelles [3].

I. 4. 1. 4 Capteurs ultrasoniques

Les capteurs ultrasoniques émettent des ondes sonores à haute fréquence et détectent les échos provenant des objets environnants. En mesurant le temps mis par les ondes pour revenir au capteur, les capteurs ultrasoniques peuvent déterminer la distance et la proximité des obstacles. Ils disposent d'un cône de détection entre 20 et 30 degrés ou 45 degrés selon le modèle [4].

Ces capteurs sont souvent utilisés en combinaison pour obtenir des informations complètes sur l'environnement du système.

I. 4. 2 Traitement des données et prise de décision

Une fois les données collectées par les capteurs, les systèmes anti-collision les traitent de manière à fournir des informations précieuses pour la prise de décision en temps réel. Voici les principales étapes du processus de traitement des données et de prise de décision [5]

I. 4. 2. 1 Acquisition des données

Les données collectées par les capteurs, telles que les informations sur la distance, la vitesse et la position des obstacles, sont transmises au système de traitement. Chaque type

de capteur fournit des données spécifiques qui sont utilisées conjointement pour une vue complète de l'environnement du système.

I. 4. 2. 2 Prétraitement des données

Avant d'être analysées, les données peuvent nécessiter un prétraitement pour éliminer les valeurs aberrantes, réduire le bruit ou corriger les erreurs. Cette étape garantit la qualité et la fiabilité des données utilisées pour la prise de décision.

I. 4. 2. 3 Fusion des données

Les données provenant de différents capteurs sont fusionnées pour créer une représentation globale de l'environnement du système. La fusion des données permet de combiner les informations provenant de sources multiples et d'améliorer la précision et la fiabilité des résultats.

I. 4. 2. 4 Analyse des données

Les algorithmes d'analyse des données sont utilisés pour extraire des informations significatives à partir des données brutes. Ces algorithmes peuvent inclure des techniques de vision par ordinateur, de traitement du signal, d'apprentissage automatique et d'intelligence artificielle. Ils permettent de détecter les objets, de suivre leur mouvement, d'estimer leur trajectoire dans l'objectif d'évaluer les risques potentiels de collision.

I. 4. 2. 5 Prise de décision

Une fois les données analysées, le système anti-collision utilise des algorithmes de prise de décision pour déterminer les actions à entreprendre. En fonction des informations fournies par les capteurs et des modèles prédictifs, le système peut déclencher des alertes visuelles ou sonores, activer le freinage automatique d'urgence, ajuster la vitesse du système ou effectuer d'autres actions visant à éviter les collisions potentielles activement ou passivement.

I. 4. 2. 6 Mise à jour en temps réel

Les systèmes anti-collision fonctionnent en temps réel, ce qui signifie que le traitement des données et la prise de décision sont continuellement mis à jour à mesure que de nouvelles informations sont collectées. Cela permet une réponse rapide et adaptative aux changements de l'environnement du système.

I. 4. 3 Systèmes de contrôle et d'alerte

Les systèmes anti-collision utilisent des systèmes de contrôle et d'alerte pour interagir avec l'environnement, le système et l'utilisateur via des algorithmes, afin de prendre des mesures préventives et d'avertir les conducteurs ou les opérateurs de la présence d'obstacles. Le fonctionnement de ces systèmes peut se présenter comme suit [5]

I. 4. 3. 1 Contrôle préventif

Lorsqu'un système anti-collision détecte un risque de collision imminente, il peut prendre des mesures préventives pour éviter l'accident. Par exemple, si le système détecte un obstacle à l'avant du véhicule et estime que la distance de freinage est insuffisante, il peut activer le freinage automatique d'urgence pour réduire la vitesse du véhicule et éviter la collision.

I. 4. 3. 2 Alertes visuelles

Les systèmes anti-collision peuvent utiliser des alertes visuelles pour attirer l'attention du conducteur ou de l'opérateur sur la présence d'obstacles. Cela peut être réalisé en affichant des icônes ou des symboles sur le tableau de bord ou le pare-brise d'un véhicule par exemple. Les alertes visuelles sont souvent accompagnées d'un système de couleur (par exemple, rouge pour indiquer un danger imminent) pour fournir une indication claire de la situation.

I. 4. 3. 3 Alertes sonores

Les alertes sonores sont un moyen efficace de signaler la présence d'obstacles. Les systèmes anti-collision peuvent émettre des avertissements sonores tels que des bips, des sons d'alarme ou des messages vocaux pour alerter le conducteur ou l'opérateur de la nécessité de prendre des mesures préventives. Les alertes sonores sont particulièrement utiles lorsque l'attention visuelle du conducteur est occupée par d'autres tâches.

I. 5 Applications des systèmes anti-collision

Les systèmes anti-collision trouvent de nombreuses applications dans divers domaines. Leur utilisation permet de prévenir les accidents et d'améliorer la sécurité dans différentes situations. La Figure I.2 illustre quelques-unes de ces applications

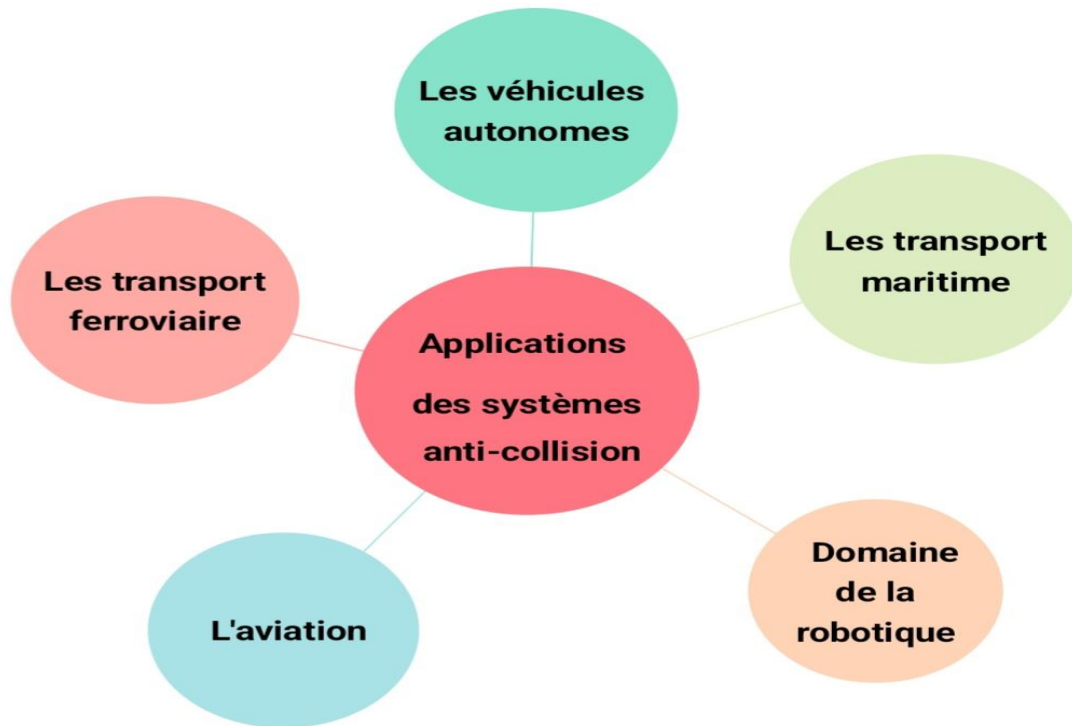


Figure I-2 Applications des systèmes anti-collision

I. 5. 1 Systèmes anti-collision dans les véhicules autonomes

Les systèmes anti-collision ont une application importante dans les véhicules automobiles modernes, contribuant à améliorer la sécurité routière (Figure I.3). Voici quelques exemples de systèmes anti-collision intégrés dans les voitures :

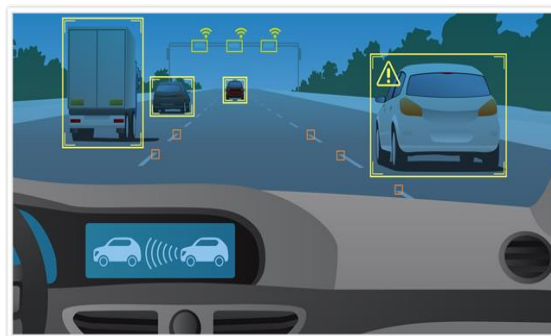


Figure I-3 Systèmes anti-collision dans les véhicules automobiles

I. 5. 1. 1 Freinage automatique d'urgence (AEB - Automatic Emergency Braking)

Ce système utilise les capteurs pour détecter les obstacles ou les véhicules présents à l'avant et active automatiquement le freinage si le conducteur ne réagit pas suffisamment rapidement. Il contribue à réduire le risque de collisions arrière en cas de situation d'urgence [6].

I. 5. 1. 2 Le régulateur de vitesse adaptatif (ACC - Adaptive Cruise Control)

Est une technologie de conduite avancée conçue pour répondre aux limites des systèmes de régulateur de vitesse traditionnels, en particulier dans les embouteillages. Le système de régulateur de vitesse traditionnel ne fournit qu'un seul mode de contrôle pour le véhicule - le contrôle de la vitesse - le système ACC fournit deux modes de contrôle : le contrôle de la vitesse et de la distance [7].

I. 5. 1. 3 Avertissement de sortie de voie (LDW - Lane Departure Warning)

Ce système avertit le conducteur lorsqu'il commence à dévier de sa voie de circulation sans utiliser le clignotant. Il peut émettre des alertes sonores ou visuelles pour rappeler au conducteur de corriger sa trajectoire [8].

I. 5. 1. 4 Avertissement de collision frontale (FCW - Forward Collision Warning)

Ce système utilise les capteurs pour détecter la proximité d'un véhicule ou d'un obstacle à l'avant. Il avertit le conducteur par des alertes visuelles ou sonores en cas de risque de collision imminente [9].

I. 5. 1. 5 Surveillance des angles morts (BSM - Blind Spot Monitoring)

Ce système utilise des capteurs pour détecter la présence de véhicules dans les angles morts du conducteur. Il émet des alertes visuelles ou sonores lorsque le conducteur tente de changer de voie alors qu'un véhicule est détecté [10].

Ces systèmes anti-collision dans les voitures modernes offrent de nombreux avantages en termes de sécurité routière. Ils aident à prévenir les collisions, réduisent les risques d'accidents causés par des erreurs humaines et augmentent la confiance et le confort des conducteurs

I. 5. 2 Systèmes anti-collision dans l'aviation

Les systèmes anti-collision jouent un rôle crucial dans l'aviation pour éviter les collisions en vol (Figure I.4). Voici quelques exemples de systèmes utilisés dans ce domaine :



Figure I-4 Systèmes anti-collision dans l'aviation

I. 5. 2. 1 Système d'alerte de trafic et d'évitement de collision (TCAS - Traffic Alert and Collision Avoidance System)

Le TCAS est largement utilisé dans l'aviation commerciale. Il utilise des transpondeurs embarqués pour échanger des informations sur la position et la trajectoire des autres avions à proximité. En cas de risque de collision, le système émet des alertes sonores et visuelles pour informer les pilotes et leur donne des directives spécifiques pour éviter la collision, telles que monter ou descendre [11].

I. 5. 2. 2 Système d'alerte de proximité du sol (GPWS - Ground Proximity Warning System)

Le GPWS est conçu pour avertir les pilotes lorsque l'avion se rapproche dangereusement du sol, notamment en cas de descente trop rapide ou d'approche trop basse. Il émet des alertes sonores pour inciter les pilotes à prendre des mesures correctives et éviter ainsi les accidents liés à la proximité du sol [12].

Ces systèmes anti-collision sont essentiels dans l'aviation pour éviter les collisions en vol. Ils permettent de surveiller en temps réel l'environnement aérien, d'alerter les pilotes sur les risques potentiels de collision et de leur fournir des directives précises pour éviter les situations dangereuses. Ils sont conçus pour fonctionner de manière autonome et indépendante, offrant ainsi une couche supplémentaire de sécurité pour les vols.

I. 5. 3 Systèmes anti-collision dans le transport ferroviaire

Dans le transport ferroviaire, les systèmes anti-collision jouent un rôle essentiel pour assurer la sécurité des trains, des passagers et des infrastructures (Figure I.5). Voici quelques exemples de systèmes anti-collision utilisés dans ce domaine :

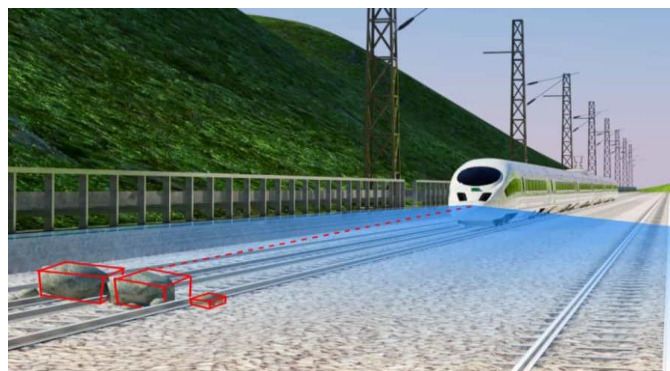


Figure I-5 Systèmes anti-collision dans Transport ferroviaire

I. 5. 3. 1 Système de contrôle des trains (CBTC - Communications-Based Train Control) :

Le CBTC est un système avancé qui utilise des communications sans fil pour surveiller et contrôler les mouvements des trains. Il permet de suivre en temps réel la position des trains, de gérer les distances de sécurité entre eux et d'appliquer automatiquement des mesures de freinage en cas de risque de collision. Cela permet d'optimiser l'utilisation des voies et de prévenir les accidents [13].

I. 5. 3. 2 Système d'arrêt automatique des trains (ATS - Automatic Train Stop) :

L'ATS est un système de sécurité ferroviaire qui utilise des balises et des aimants placés le long des voies pour transmettre des signaux aux trains. Ces signaux peuvent indiquer aux conducteurs de train de réduire leur vitesse, de s'arrêter ou de prendre d'autres mesures de sécurité en fonction de la situation. L'ATS est conçu pour empêcher les trains de franchir des signaux rouges ou d'entrer dans des sections de voie occupées [14].

I. 5. 3. 3 Système d'alerte de collision imminente (ICA - Imminent Collision Alert)

Les systèmes d'ICA sont conçus pour détecter les situations de collision imminente entre les trains. Ils utilisent des capteurs et des algorithmes pour évaluer les trajectoires des trains et identifier les situations de risque. Lorsqu'une collision imminente est détectée, des alertes sont générées pour avertir les conducteurs de train et leur permettre de prendre des mesures d'évitement [15].

Ces systèmes anti-collision dans le transport ferroviaire sont conçus pour garantir la sécurité des passagers, prévenir les collisions entre les trains et assurer une exploitation efficace et fiable des réseaux ferroviaires. Ils contribuent à réduire les risques d'accidents, à optimiser la capacité des voies et à améliorer la ponctualité des services ferroviaires.

I. 5. 4 Systèmes anti-collision dans le transport maritime :

Dans le transport maritime, les systèmes anti-collision sont utilisés pour éviter les collisions entre les navires et garantir la sécurité des voies navigables (Figure I.6). Voici quelques exemples de systèmes anti-collision utilisés dans ce domaine :

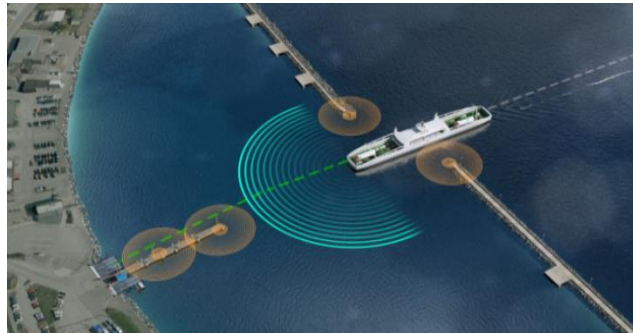


Figure I-6 Systèmes anti-collision dans le transport maritime

I. 5. 4. 1 **Système d'identification automatique (AIS - Automatic Identification System) :**

L'AIS est un système obligatoire à bord des navires qui permet le suivi et l'échange d'informations entre les navires et les stations côtières. Il utilise des transpondeurs pour émettre des signaux contenant des données telles que la position, la vitesse, la direction et l'identité du navire. Les informations AIS sont utilisées pour détecter les navires à proximité et évaluer les risques de collision [16].

I. 5. 4. 2 **Système d'alerte de proximité (CPA - Closest Point Approach) :**

Les systèmes CPA sont utilisés pour calculer et afficher la distance minimale attendue entre deux navires, ainsi que le moment où cette distance sera atteinte. Ces systèmes avertissent les opérateurs si la distance minimale est inférieure à un seuil critique, leur permettant ainsi de prendre des mesures d'évitement appropriées pour éviter la collision [17].

Ces systèmes anti-collision dans le transport maritime sont conçus pour améliorer la sécurité et la gestion du trafic. Ils permettent de détecter les navires à proximité, d'évaluer les risques de collision et de prendre des mesures d'évitement appropriées. Grâce à ces systèmes, les opérateurs maritimes peuvent naviguer en toute sécurité et prévenir les accidents sur les voies navigables.

I. 5. 5 Systèmes anti-collision dans le domaine de la robotique :

Les systèmes anti-collision jouent un rôle crucial pour assurer la sécurité des opérations et prévenir les accidents (Figure I.7). Voici quelques exemples de systèmes anti-collision utilisés dans les robots :



Figure I-7 Systèmes anti-collision dans le domaine des robots

I. 5. 5. 1 **Evitement basé sur les capteurs :**

Les robots sont équipés de capteurs de proximité (tels que des capteurs infrarouges, des capteurs ultrasoniques ou des capteurs de distance laser) ou des systèmes de vision. Ces capteurs permettent de détecter les obstacles et les objets environnants dans le champ de perception du robot. Ils fournissent des informations sur la distance, la position et la nature des obstacles, ce qui permet au robot d'éviter les collisions en ajustant sa trajectoire ou en s'arrêtant.

I. 5. 5. 2 **Systèmes de sécurité intégrés :**

Certains robots sont équipés de systèmes de sécurité intégrés qui surveillent en permanence les paramètres opérationnels du robot. Ces systèmes peuvent détecter les situations anormales, telles que des mouvements brusques, des chocs ou une surcharge, et déclencher des actions d'arrêt d'urgence pour éviter les accidents.

Ces systèmes anti-collision dans les robots sont essentiels pour garantir la sécurité des opérations, protéger les personnes et les biens, et éviter les collisions avec les obstacles environnants. Ils permettent aux robots de naviguer de manière autonome dans des environnements complexes, tout en minimisant les risques d'accidents et de dommages [18].

I. 6 **Etude des systèmes anti-collision dans le transport ferroviaire**

I. 6. 1 **Les facteurs contribuant aux accidents de tramway**

Les piétons et les conducteurs de voitures sont les principaux responsables des accidents de tramway en raison de leur mauvaise estimation du temps d'arrêt du tramway. Les tramways ont un coefficient d'adhérence inférieur d'environ $\sigma_{tram} = 0,3$ par temps sec, tandis que les voitures ont un coefficient d'adhérence de $\sigma_{car} = 0,8$, soit presque trois fois plus élevé. Cela affecte les distances de freinage, le freinage du tramway étant inversement proportionnel au rapport entre le coefficient d'adhérence rail-route et celui de la gomme-asphalte.

$$\Delta x_{\text{tram}}^{\text{stop}} = \frac{\sigma_{\text{car}}}{\sigma_{\text{tram}}} \times \Delta x_{\text{car}}^{\text{stop}}$$

$\Delta x_{\text{car}}^{\text{stop}}$: L'espace d'arrêt de la voiture

$\Delta x_{\text{tram}}^{\text{stop}}$: L'espace d'arrêt du tramway

Une autre cause d'accidents de tramway est la traversée des piétons sans faire attention au passage du tramway. Ces facteurs affectent le comportement des conducteurs de tramway, qui doivent faire face à des difficultés telles que la sélection de vitesses appropriées et l'adaptation à l'environnement urbain chaotique. Il est donc évident qu'un T-ADAS est nécessaire pour aider le conducteur dans sa tâche de conduite. Le développement de T-ADAS nécessite de partir du monde ferroviaire, auquel appartiennent les tramways, mais également de comparer les services et les technologies avec le monde automobile en raison de l'environnement d'exploitation unique des tramways sur la route [19].

I. 6. 2 Système avancé d'assistance à la conduite de tramway :

(En anglais T-ADAS signifie Tramway Advanced Driver Assistance System), C'est un système de sécurité embarqué destiné à aider le conducteur de tramway dans sa tâche de conduite, en fournissant des informations et des alertes sur les conditions de circulation et les obstacles potentiels. Le T-ADAS peut inclure des fonctionnalités telles que la détection des piétons, la surveillance de l'état du tramway, la surveillance de la voie ferrée, la signalisation des feux de circulation et des panneaux de signalisation, ainsi que la gestion de la vitesse et de la distance de sécurité. Son objectif est d'améliorer la sécurité et la fiabilité du transport en commun par tramway [19].



Figure I-8 La détection des piétons et la surveillance de la voie ferrée

I. 6. 3 La possibilité de transfert de la technologie ADAS - De la voiture au tramway-

Le transfert de la technologie ADAS, qui se réfère aux systèmes avancés d'aide à la conduite, de la voiture au tramway est une perspective prometteuse pour améliorer la sécurité et l'efficacité des transports en commun. Cette évolution est soutenue par des études récentes,

dont celle présentée dans le tableau 1.1, qui mettent en évidence les avantages potentiels de l'application de ces technologies dans le contexte des tramways.

Type ADAS	Description	Transférabilité de la voiture au tramway
régulateur de vitesse adaptatif (AdaptiveCruiseControl(ACC))	Le système ajuste automatiquement la vitesse du véhicule pour maintenir une distance sécuritaire par rapport aux véhicules qui se trouvent devant.	Le transfert est possible, même s'il n'est pas très important en milieu urbain. Compte tenu des paramètres de circulation du tramway, l'ACC devrait être utilisé pour maintenir la même distance de sécurité avec le véhicule à l'avant en ajustant continuellement la vitesse, minimisant ainsi le risque de collision
Avertissement de collision avant (Forward Collision Warning (FCW)) et Coupure d'urgence autonome (Autonomous Emergency Breaking (AEB))	Le FCW avertit le conducteur par un avertissement sans activation du freinage automatique. Cette dernière fonction est mise en œuvre par AEB. Les systèmes FCW et AEB fonctionnent généralement ensemble, mesurant le temps de collision avec le véhicule vers l'avant; FCW avertit le conducteur lorsque ce temps est devenu critique et que le freinage peut ne pas être suffisant pour éviter la collision. Si le conducteur ne réagit pas, l'AEB déclenche un freinage d'urgence.	Le transfert est possible. Le tramway équipé de ce système peut alerter le conducteur lorsqu'un obstacle (par ex. piétons, vélos ou voitures) se trouve le long de la trajectoire du tramway. Si le conducteur ne réagit pas à temps, compte tenu de la vitesse du tramway et de la distance par rapport à la cible, le système peut déclencher un freinage atténuant le risque de collision.
Park Assist (PA)	Cette technologie aide les conducteurs à stationner leur véhicule dans des espaces clos, lorsqu'ils	Le transfert n'est pas possible.

	s'approchent trop d'un obstacle, le système émet un signal acoustique avec une fréquence croissante à mesure que la distance à l'obstacle diminue.	
Alerte trafic croisé (Cross Traffic Alert (CTA))	Le système avertit le conducteur si un autre objet (véhicule, piéton ou cycliste) entre dans sa zone d'intérêt lorsque le véhicule se déplace en marche arrière. Une application typique pour les voitures se trouve à l'intérieur des stationnements.	Le transfert n'est pas possible. Le tramway ne fait pas marche arrière sur le trajet normal. Cependant, les tramways ont normalement deux cabines, une pour chaque sens de déplacement, de sorte que la fonction peut être utile dans les phases de manœuvre à l'intérieur des dépôts.
Système de surveillance des conducteurs (Driver Monitoring System (DMS))	L'attention du conducteur est continuellement surveillée. Si le conducteur ne fait pas attention à la route qui le précède et qu'une situation dangereuse est détectée, le système avertira le conducteur en allumant des feux clignotants ou des signaux sonores	Le transfert est possible. Ce système pourrait réduire de nombreux accidents de distraction au volant en surveillant l'attention du conducteur. Par exemple, le dernier accident de tramway à Milan (Italie) est dû à une distraction du conducteur qui en conséquence heurté une voiture.
Avertissement de collision arrière (Rear Collision Warning (RCW))	RCW scanne la zone derrière le véhicule et vous avertit d'une collision potentielle.	Le transfert est possible. Le conducteur peut être averti lorsqu'un véhicule arrivant à grande vitesse par l'arrière n'a pas des distances de freinage suffisantes.
Contrôle adaptatif de la lumière (Adaptive Light Control (ALC))	Ce système régule l'intensité du faisceau lumineux et l'orientation des phares. Si le capteur	Le transfert est possible, même s'il n'est pas très important en milieu urbain. Ce système automobile de

	détecte une obscurité totale à l'avant (par exemple près d'une courbe), il active le faisceau de route. A l'arrivée d'un véhicule dans la direction opposée, des ombres seront créées qui suivront ce véhicule afin de ne pas éblouir le conducteur.	réglage du faisceau lumineux dans les coins sombres peut être transféré au tramway afin d'offrir une meilleure visibilité dans les courbes rendant les bords de route plus visibles.
Détection des angles morts (Blind Spot Detection (BSD)) /Système d'information sur les angles morts (Blind Spot Information System (BLIS))	Ce système détecte un véhicule adjacent dans l'angle mort arrière et avertit le conducteur de changer de voie pour éviter l'accident avec le véhicule qui arrive.	Le transfert est possible. Un avertissement peut être produit pour alerter le conducteur lorsqu'un autre véhicule est à côté du tramway, ce qui rend un angle mort évident.
Reconnaissance des panneaux de signalisation (Traffic Sign Recognition (TSR))	Le TSR est en mesure de détecter, de suivre et de reconnaître les panneaux de signalisation, p. ex., « limite de vitesse », « enfants » ou « tourner en avant ». Le système avertit le conducteur avec des signaux spécifiques dans le tableau de bord et, éventuellement, ajuste la vitesse du véhicule à la valeur limite	Le transfert est possible. Le système pourrait être programmé et formé pour reconnaître non seulement les panneaux routiers ordinaires, mais aussi les panneaux de tramway
Système de maintien des voies (Lane Keeping System (LKS))	Ce système avertit le conducteur et prend automatiquement des mesures pour s'assurer que le véhicule reste dans sa voie	Le transfert n'est pas possible.

Tableau 1-1 La transférabilité de la technologie ADAS - De la voiture au tramway-

I. 6. 4 Technologies et capteurs pour l'amélioration de l'automatisation dans les transports en commun : De niveau d'automatisation actuel (GoA 0) à un niveau supérieur (GoA 1)

Les technologies et les capteurs nécessaires pour passer du niveau d'automatisation actuel (Grade of Automation GoA 0) à un niveau supérieur de GoA 1 dans les systèmes de transport en commun. Les tramways ne peuvent pas recueillir d'informations de l'infrastructure ferroviaire, et doivent être considérés similaires aux voitures en termes de leur capacité à utiliser les informations de l'environnement pour prendre des décisions de conduite [19].

T-ADAS	les Capteurs					GoA 1
	Lidar	Radar	IR	Caméra	Ultrason	
ACC		X				X
FCW		X				X
AEB	X	X	X	X		X
TSR	X			X		X
BLIS		X	X		X	X
ALC			X			
RCW		X				X
DMS				X		X

■ équitable ■ bon

Tableau 1-2 L'intégration de capteurs sur un tramway.

I. 6. 5 Les types de Capteurs dans les transports en commun

Les types de Capteurs	Capteurs spécifiques	Distance	Coût	Avantages	Désavantage
Acoustique [21]	- SONY ECM-77B	-Dépend de l'amplitude des ondes sonores et de la sensibilité du micro	≈ 350 \$	- Microphone omnidirectionnel - Une solution économique - Temps réel	- Problème d'interférence - Sensible au bruit - Courte portée

Radars [22, 23, 24, 25]	- Delphi adaptive Cruise control (Régulateur de vitesse adaptatif Delphi)	175 mètres	≈2 000 \$	- Mesurer la distance directement avec moins de ressources informatiques - Portée de détection plus longue que les capteurs acoustiques et optiques - Robuste par temps de brouillard ou de pluie, et pendant la nuit.	- Problème d'interférence - Coût plus élevé que l'acoustique - Problème de classement - Plus de consommation d'énergie que le capteur acoustique et optique
Laser/Lidar [26, 27,28]	Télémètre laser Velodyne HDL-64E (Lidar 3D)	120 mètres	≈ 70.000 \$	- Portée de détection plus longue que les capteurs acoustiques et optiques - Indépendant des conditions météorologiques - Les scanners laser/lidar modém acquièrent des informations hautes résolution et 3D	- Dépendance aux infrastructures routières - Plus de consommation d'énergie que les autres capteurs - Les scanners 3D à grande vitesse sont chers
	SICK LMS511-10100 (2D)	80 mètres	≈7 000 \$		
caméra [29-30]	SV-625B	100 m pour le jour 12 m pour la nuit (profondeur de mise au point)	≈ 160 \$	- Faible coût, plus facile à installer et à entretenir - Résolution plus élevée et angle de vue plus large - Information détaillées dans les images	- La qualité de l'image dépend de l'éclairage et des conditions météorologiques

Tableau 1-3 Une étude des différents types de capteurs

I. 6. 6 Intégration des systèmes avancés d'aide à la conduite dans les tramways :

Différents capteurs sont nécessaires pour les opérations ferroviaires automatisées, car ils offrent diverses capacités d'observation et de détection. Les trams ne peuvent pas changer de sens de marche, donc les capteurs doivent être installés dans les deux cabines. Les entreprises de transport public peuvent envisager d'adopter cette approche pour moderniser leurs systèmes de tramway, compte tenu des avancées des systèmes d'aide à la conduite dans l'industrie automobile. L'intégration des tramways dans l'écosystème des voitures autonomes connectées pourrait être une progression logique.

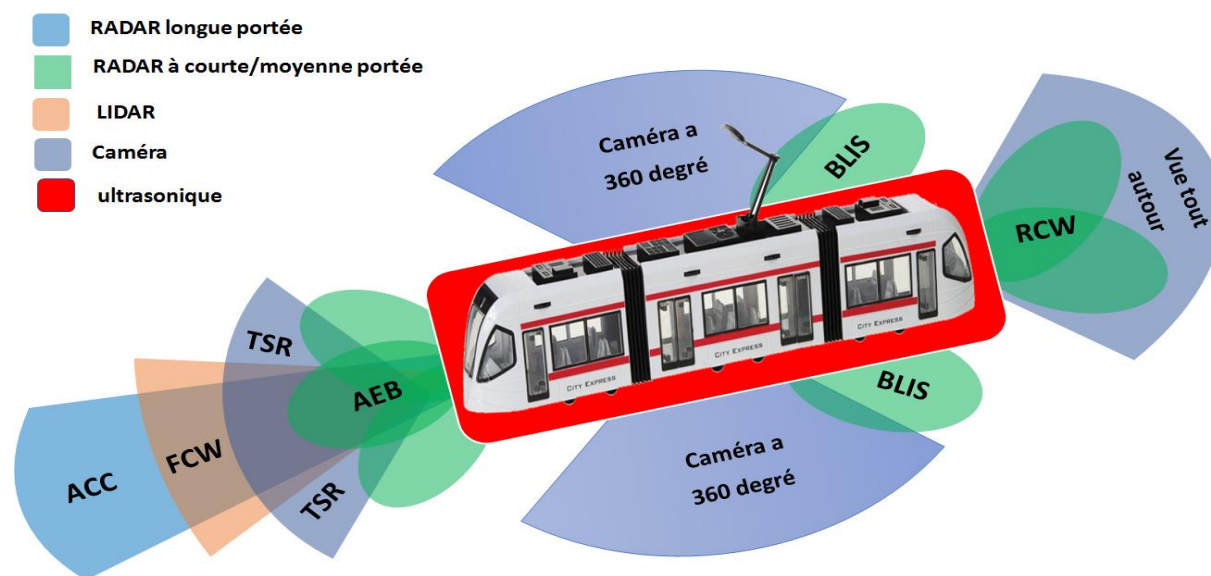


Figure I-9 Répartition technique et configuration des capteurs du tramway"

I. 6. 7 Les caractéristiques principales de tramway :

Voici le tableau qui représente les principales caractéristiques d'une ligne de tramway [20].

Les caractéristiques principales	
Longueur hors-tout	43 760 mm
Largeur de caisse	2 400 mm
Hauteur maxi au dessus du rail	3 321 mm
Hauteur du plancher au dessus du rail	350 mm

Distance entre pivots des bogies	11 142 mm
Empattement des bogies	1 600 mm
Masse à vide en ordre de marche	54,92 t
Masse en charge normale	75,92 t
Nombre de bogies moteurs	3
Nombre de bogies porteurs	1
Nombre de places assises	71
Charge normale	287 passagers
Charge maximale	395 passagers
Vitesse maximale	70 km/h
Vitesse minimale	10 km/h
L'accélération maximale	1,5 m/s ²
L'accélération minimale	0,8 m/s ²
Puissance maximale à la jante (traction)	880 kW

Tension d'alimentation	750 V Courant Continu
Rayon de courbure minimal	25

Tableau 1-4 Les caractéristiques principales de tramway

I. 7 Conclusion :

En conclusion, nous avons exploré dans ce chapitre les systèmes anti-collision, leur définition, leurs objectifs et leurs principes de fonctionnement dans différents domaines de transport tels que l'automobile, l'aviation, le ferroviaire, le maritime, ainsi que dans le domaine de la robotique. Par la suite, nous avons traité le cas du transport ferroviaire dont il fait l'objet de notre travail.

Afin de doter ces systèmes de la capacité de raisonnement et de prise de décision, il est nécessaire d'exploiter des techniques avancées de la navigation dans le domaine de la robotique mobile, ce qui va être présenté dans le chapitre suivant.

Chapitre II : **Système
d'aide à la conduite :
méthodologie et
conception**

II. 1 Introduction

Dans ce deuxième chapitre, nous aborderons le système d'aide au conduit proposé dans ce travail. L'objectif est de garantir la sécurité tout en respectant les choix de l'utilisateur, ainsi que le modèle cinématique du robot. Pour ce faire, ce système doit faire recours à la technique de navigation réactive pour assurer la sûreté. Ce chapitre est ainsi structuré comme suit : nous commençons par présenter les techniques de navigation réactive les plus pertinentes et justifier notre choix relativement à ce point, à savoir la technique d'échantillonnage de l'espace de contrôle (ISS). Par la suite, nous illustrons en détail la solution développée étant une version améliorée de la technique ISS, où nous expliquons la méthodologie, le concept et l'algorithme.

II. 2 Approches de navigation réactive

Pour permettre à un système mobile de naviguer en toute sécurité, il faut le doter de la capacité d'éviter les obstacles. En effet, dans la littérature, il existe plusieurs méthodes de navigation réactive (ou méthodes d'évitement d'obstacles) que nous allons présenter dans ce qui suit.

II. 3 L'étude des approches de navigation réactive

L'étude des approches de navigation a été largement abordée dans la littérature afin de résoudre le problème de déplacement des robots mobiles dans des environnements statiques ou dynamiques [31]. Nous examinerons différentes approches de navigation. Comme illustre dans la figure II.1, notamment les méthodes réactives qui consistent en des stratégies d'évitement d'obstacles, ainsi que les méthodes d'échantillonnage largement utilisées dans le domaine de la robotique.

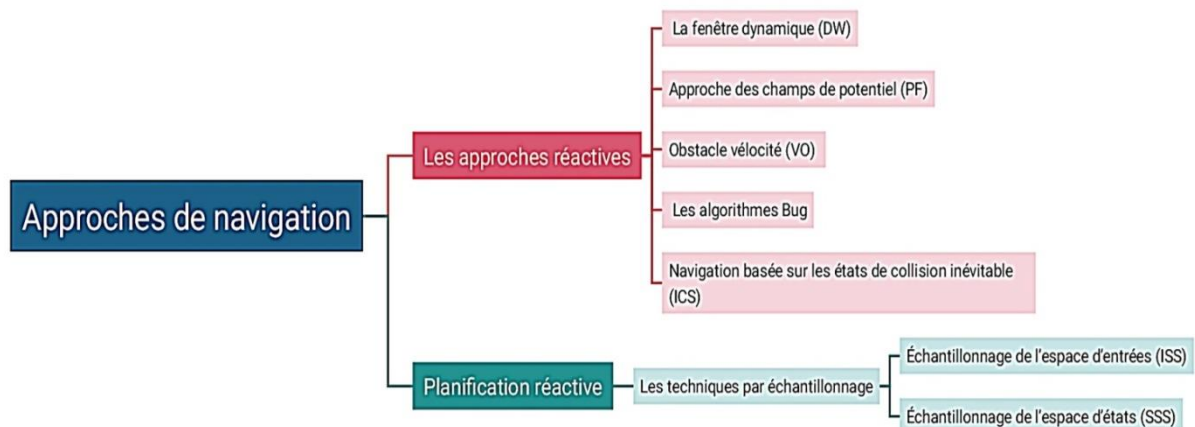


Figure II-1 L'étude des approches de navigation

II. 3. 1 Les approches réactives :

Les méthodes réactives (appelées méthodes d'évitement d'obstacles) utilisent les informations des capteurs embarqués pour chaque pas de temps est de être appliquée à la commande de l'actionneur. Cela les rend idéales pour les méthodes qui fonctionnent en temps réel. Une description non exhaustive de ces méthodes est donnée dans les paragraphes suivants [32].

II. 3. 1. 1 La fenêtre dynamique (DW)

La méthode de la fenêtre dynamique (DW) est une approche qui opère dans l'espace des vitesses du robot, prenant en compte les contraintes cinématiques et dynamiques. Elle vise à sélectionner un couple de vitesses (linéaire et angulaire) permettant d'éviter les obstacles perçus. En utilisant une perception locale de l'environnement, DW génère un chemin circulaire et évalue les contraintes kinodynamiques. Elle sélectionne ensuite le couple de vitesses le plus pertinent parmi les vitesses admissibles (Voir figure II.2), limitées par une fenêtre dynamique qui représente les limitations dynamiques du robot [31].

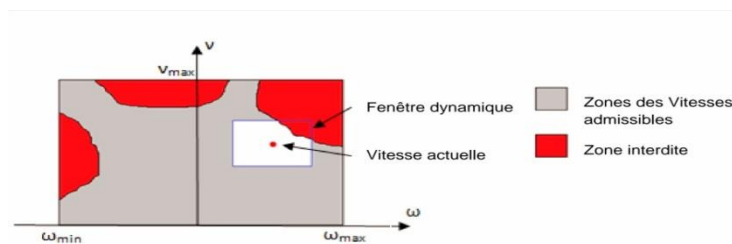


Figure II-2 L'approche de la fenêtre dynamique.

II. 3. 1. 2 L'approche des champs de potentiel (PF)

L'approche des champs de potentiel est une méthode utilisée pour le calcul du mouvement des bras manipulateurs. Elle repose sur la construction d'une fonction de potentiel artificiel dans l'espace de configuration, où le gradient de cette fonction définit un champ de vecteur. Ce champ de vecteur guide le robot vers sa destination en utilisant une force attractive pour l'objectif et des forces répulsives pour les obstacles. Ainsi, le robot est dirigé de sa position initiale vers sa position cible tout en évitant les obstacles (Voir figure II.3). Les forces répulsives sont plus intenses lorsque le robot est proche d'un obstacle, mais elles n'affectent pas son mouvement lorsqu'il est éloigné de l'obstacle [33].

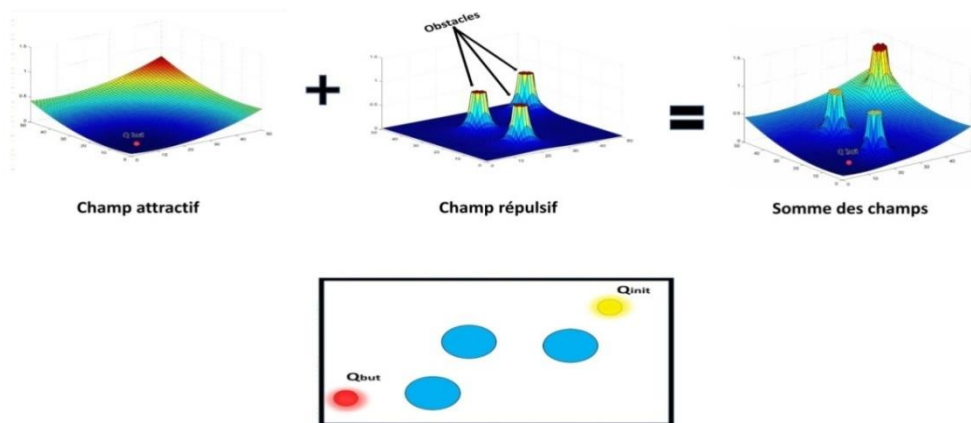


Figure II-3 Champs de potentiel pour un environnement contenant trois obstacles.

II. 3. 1. 3 Obstacle vélocité (VO)

L'obstacle vélocité (VO) vise à générer une commande de vitesse pour le robot en prenant en compte les vecteurs de vitesse du robot et des obstacles mobile illustrée dans la figure II.4. Cette approche permet de calculer une trajectoire sûre en considérant les contraintes d'accélération des moteurs et les dynamiques du robot [34].

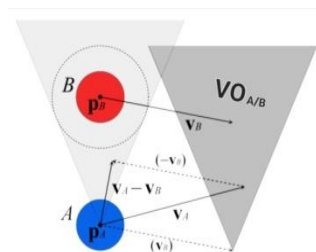


Figure II-4 L'approche VO

II. 3. 1. 4 Algorithmes de Bugs

Les algorithmes de contournement des obstacles, tels que Bug, Bug1 et Bug2, permettent aux robots d'éviter les obstacles sur leur chemin. Bug1 oblige le robot à passer par toutes les extrémités de l'obstacle, tandis que Bug2 ne gère que partiellement ces extrémités. Comme illustre dans la figure II.5. Ces méthodes nécessitent des capteurs pour détecter les obstacles à proximité immédiate. Des variantes ont été développées pour améliorer ces algorithmes en réduisant la distance parcourue par le robot [35].

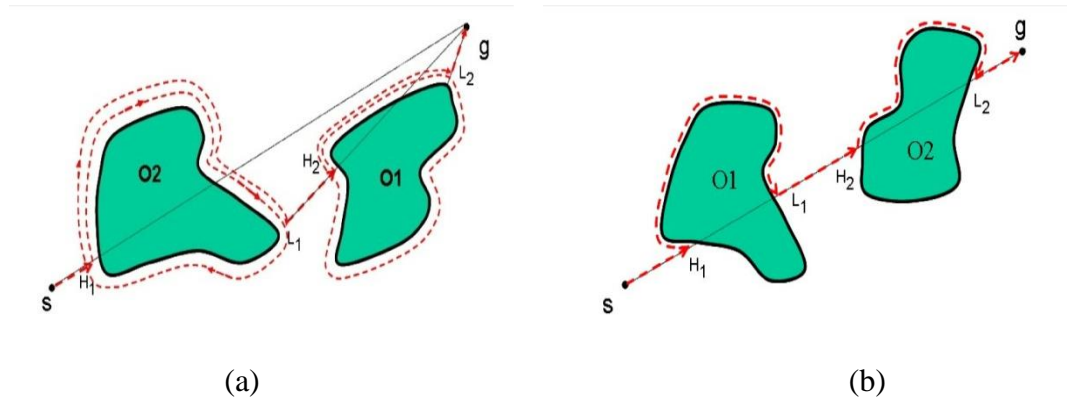


Figure II-5 Calcul du chemin du robot par les algorithmes Bug: (a) Bug1, (b) Bug2

II. 3. 1. 5 Navigation basée sur les états de collision inévitable (ICS)

Bien que la majorité des approches présentées précédemment puissent être appliquées en environnement dynamique, dans un milieu urbain encombré de piétons, voitures ou autres robots, il ne peut pas être garanti qu'aucune collision n'aura jamais lieu. Hors, dans de tels milieux, la sûreté du système robotique ainsi que ce qui l'entoure est primordiale. C'est pour faire face à ce genre de problèmes que le concept d'états de collision inévitable a été proposé dans [5]. Son principe est de déterminer les états pour lesquels, quelle que soit la trajectoire du robot, à un instant « t » une collision aura lieu entre le robot et les obstacles. Ainsi, la sûreté est garantie tout simplement en évitant ces états. En effet, il faut aller au-delà d'éviter les états de collision (le cas de la majorité des approches d'évitement d'obstacles), il faut éviter les ICS. Le concept ICS prend en compte non seulement la dynamique de l'environnement, mais également le comportement futur des obstacles mobiles.

II. 3. 2 Planification réactive :

Les méthodes de planification réactive ont été développées pour combiner les avantages des méthodes réactives tout en atténuant leurs inconvénients respectifs. Elles sont particulièrement adaptées à la navigation dans des environnements dynamiques et inconnus.

II. 3. 2. 1 Les techniques par échantillonnage :

Les techniques d'échantillonnage sont largement utilisées dans la navigation hiérarchique pour permettre une planification en temps réel dans des environnements partiellement ou totalement inconnus. Ces techniques consistent à générer et à trier des mouvements candidats. L'objectif principal est de trouver la meilleure façon d'échantillonner efficacement l'espace.

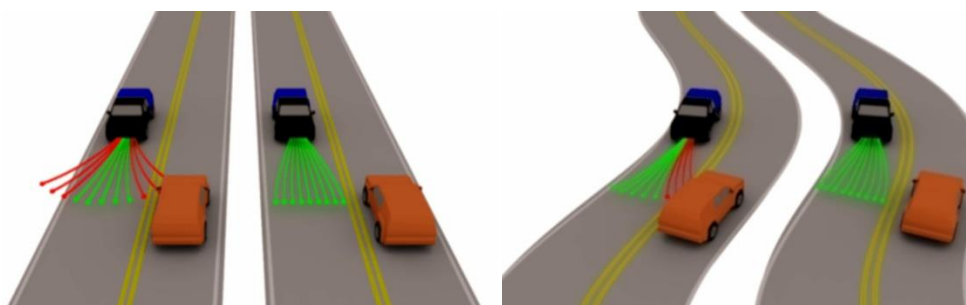
des mouvements réalisables. Deux types de techniques sont présentés : l'échantillonnage de l'espace des entrées (ISS) et l'échantillonnage de l'espace des états (SSS) [36].

II.3.2.1.1 Échantillonnage de l'espace d'entrées (ISS) :

Échantillonnage de l'espace d'entrées (ISS) Cette technique échantillonne dans l'espace d'entrées (l'espace des contrôles) et génère des mouvements faisables. Tous ces mouvements sont fondamentalement exécutables, car chaque entrée est simulée par le modèle du système robotique. Le principe de l'échantillonnage de l'espace d'entrées est simple ; à partir d'un état initial, le modèle prédictif du système est utilisé pour générer un ensemble de trajectoires de contrôle (leur forme dépend donc du modèle du système et de son état initial (courbures, vitesses)). Ces trajectoires peuvent être triées par rapport à une fonction de coût donnée (Voir figure II 6 (a)) De plus, elles peuvent être testées par rapport à non collision avec les obstacles en utilisant par exemple l'une des techniques de navigation locale

II.3.2.1.2 L'échantillonnage de l'espace d'états (SSS) :

L'échantillonnage de l'espace d'états (SSS) est une technique qui consiste à échantillonner l'espace d'états plutôt que l'espace des contrôles. Cela permet de spécifier les conditions aux limites pour chaque mouvement parmi les trajectoires générées. Cependant, cette technique ne garantit pas la faisabilité du mouvement. Une solution consiste à utiliser une méthode de génération de trajectoire inverse (Voir figure II.6 (b)), où l'on cherche un contrôle qui satisfait à la fois le modèle du mouvement et les contraintes limites de l'état pour un robot mobile donné.



(a) Espace de contrôle (b) Espace d'états (a) Espace de contrôle (b) Espace d'états

Figure II-6 Échantillonnage de trajectoire entre espace de contrôle et espace d'état pour un véhicule sur une route

II. 4 La Solution proposée : Safe ISS

Parmi les méthodes de navigation citées ci-dessus, la méthode d'échantillonnage de l'espace d'entrées (ISS) est un choix judicieux car en vue d'une application d'aide à la conduite dans le transport, ISS d'une part prend en compte les contraintes Kino dynamiques du système et d'autre part elle peut s'adapter à la dynamique de l'environnement et aux situations imprévues. Toutefois, dans une telle application le critère sécurité est primordial. Par contre, cette technique permet uniquement l'évitement d'obstacles sans apporter aucune garantie de non collision, ce qui est le cas de la majorité des méthodes d'évitement d'obstacles. Pour ce faire, afin de garantir la sûreté du système d'aide à la conduite, la méthode ISS a été améliorée dans ce sens. La variante proposée est appelée Safe ISS.

L'algorithme Pass ISS comprend plusieurs étapes savoir, la modélisation de l'environnement et l'échantillonnage de l'espace de contrôles. Chacun de ces points sera détaillé dans ce qui suit.

II. 4. 1 Perception et modélisation de l'environnement

Afin de garantir une navigation sans contraintes dans l'environnement, il est essentiel de prendre en compte la géométrie du robot, en veillant à ce qu'aucun point du robot ne touche les obstacles environnants. Pour ce faire, une représentation dans l'espace de configuration est nécessaire, où le robot est symbolisé par un point et les obstacles sont étendus en fonction des dimensions du robot.

II. 4. 1. 1 Espace de configuration

Pour générer des plans de mouvement pour un système robotique, il est crucial de définir sa position ainsi que la position de chaque point de son corps. La configuration d'un système de robot mobile est déterminée par les paramètres de sa position (x, y) et de son orientation φ , avec φ dans la plage de $[-\pi, \pi]$. À partir de là, nous pouvons introduire le concept d'espace de configuration (noté "espace-C" ou CS), qui représente l'ensemble des configurations accessibles au robot. La dimension de cet espace C correspond au nombre de degrés de liberté du système [37].

En général, l'espace de configuration est construit en utilisant des informations sur la géométrie du robot et de l'environnement dans lequel il se déplace, appelé "espace de travail". Le robot lui-même peut être décrit comme un sous-ensemble de l'espace de travail occupé par A dans la configuration q, et est représenté par $A(q)$. Les différents obstacles de

l'environnement, notés B_i avec i allant de 1 à n (où n est le nombre d'obstacles), sont décrits par des sous-ensembles fermés de l'espace de travail.

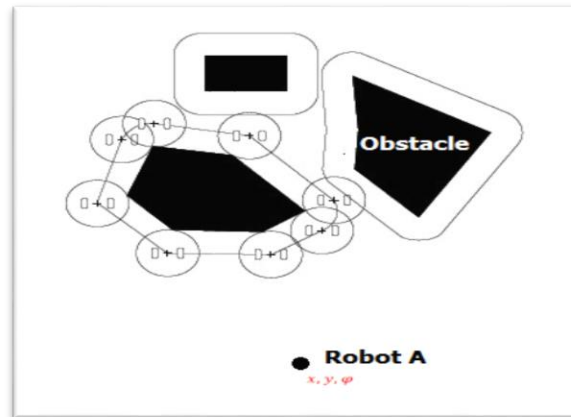


Figure II-7Espace de configuration

II. 4. 1. 2 Modélisation des données des capteurs

La première étape de l'algorithme consiste à transférer les données des capteurs dans le nouveau repère local, tout en augmentant la taille de chaque point en une forme circulaire. Cette transformation est réalisée à l'aide des fonctions "Change Reference" et "GrowPoint". L'algorithme itère sur chaque point de l'ensemble de points d'entrée, applique la transformation et stocke les points agrandis dans un nouveau tableau. Le processus est expliqué par l'Algorithme 1, où chaque point est transformé, agrandi et ensuite stocké dans un tableau appelé "PCercle" (selon algorithme 1).

Algorithm 1: Point Growth and Circle Construction

```

i ← 0 ;
for each point ∈ points do
    Change Reference (point);
    PCercle [i] ← GrowPoint (point) ; //PCercle [i] est le point agrandi
    i = i + 1 ;
end

```

II. 4. 2 Echantillonnage de l'espace local du robot

Afin de modéliser l'environnement du robot, nous avons réalisé un échantillonnage de l'espace de configuration en plusieurs zones de détection. Chaque zone est définie par un angle d'ouverture α_{zones} , comme illustré dans la Figure II.8.

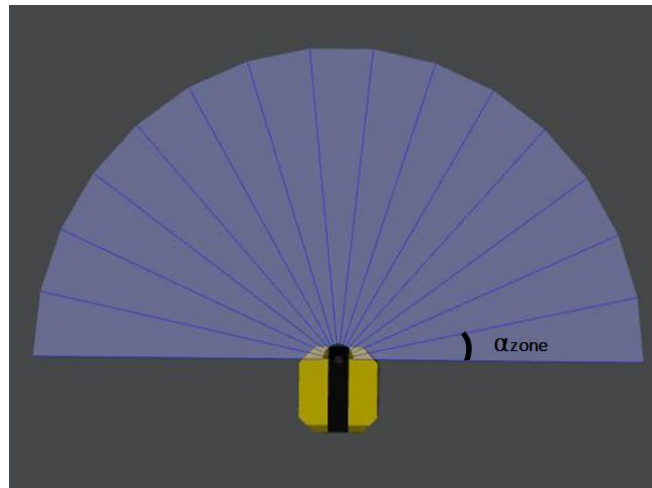


Figure II-8 Echantillonnage de l'espace local du robot (Le robot contient 15 zones de détection)

Ces zones sont construites dans le repère local RL lié au robot, comme illustré dans l'algorithme 2. Chaque point de l'environnement est associé à une zone spécifique. Par conséquent, chaque zone se voit attribuer une valeur indiquant si elle est occupée ou libre, en fonction de la présence ou de l'absence d'un obstacle. Ainsi, la position d'un point détecté par le système de perception est d'abord définie dans le repère local RL. Ensuite, en utilisant une distance minimale de détection, chaque zone est testée pour déterminer si elle est occupée ou non (c'est-à-dire si elle représente un espace interdit ou admissible).

Algorithm 2 : Determine Admissible Zone

Input: Data (Données laser), nbrZone (Nombre de zones), DistanceMax

Output: Zone

```

1: for (i=0 to nbrZone) do
2:   Zone[i].pnt = Data
3: end for

4: for (i=0 to nbrZone) do
5:   if (Zone[i].pnt.D > DistanceMax) then
6:     Zone[i].access = 0; // Free Zone
7:   else
8:     Zone[i].access = 1; // Occupied Zone
9:   end if
10: end for

11: return Zone
  
```

II. 4. 2. 1 La génération d'un ensemble de trajectoires dans l'espace de recherche

Pour notre application, nous avons choisi d'utiliser la méthode d'échantillonnage de l'espace des entrées (ISS). L'approche ISS consiste à générer un ensemble d'actions réalisables en échantillonnant l'espace de contrôles. Cela signifie que nous générons des déplacements vers d'autres états finaux en fonction des directives globales. Cette méthode se distingue des autres travaux mentionnés dans le chapitre précédent car elle est capable de créer des espaces de recherche pour la planification des mouvements. Nous utilisons des méthodes d'échantillonnage de l'espace de contrôle lors de l'application de contraintes cinématiques ou dynamiques arbitraires, régis par l'équation du système suivante :

$$\dot{s} = f(s, u)(1)$$

Le vecteur d'entrée ou de contrôle « u » et le vecteur d'état « s » sont les deux des points variant dans le temps dans l'espace d'entrée et d'état, respectivement, avec $u = (v, \omega)$, où v est la vitesse linéaire et ω est la vitesse angulaire.

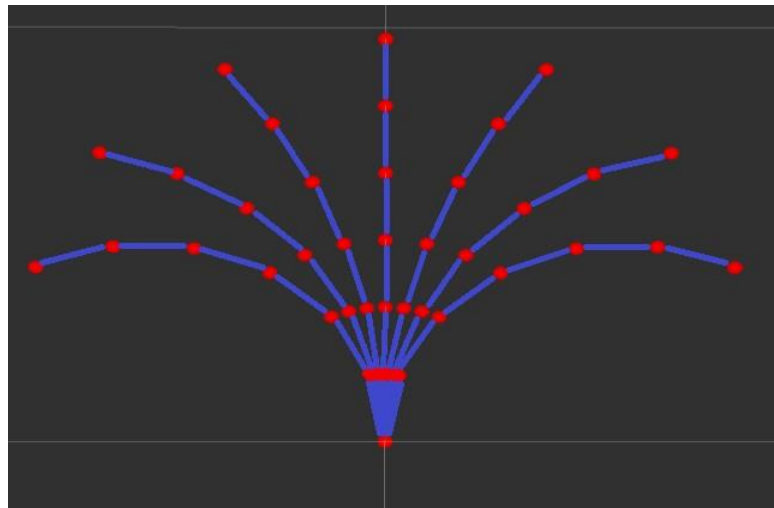


Figure II-9 Ensemble de trajectoires – Affichage sur RVIZ (sous ROS)

Le Principe de la technique d'échantillonnage d'espace de contrôle est de générer un ensemble d'actions en résolvant des trajectoires entre des paires d'états limites « π_N ». Le premier état de chaque couple est l'état " S_I ," actuel du robot et le deuxième état est l'état du terminal cible " S_F ," qui doit être atteint en fin de trajectoire.

$$\pi_N = \begin{bmatrix} S_{I,0} & S_{I,1} & S_{I,2} & \dots & S_{I,F} \\ S_{F,0} & S_{F,1} & S_{F,2} & \dots & S_{F,F} \end{bmatrix} \quad (2)$$

Où

$$S = [x, y, \varphi]^T$$

Avec : x, y, φ , les coordonnées cartésiennes, et l'orientation.

Nous devons donc déterminer l'entrée ou le contrôle approprié $u = (v, \omega)$ qui satisfait à la fois la paire de contraintes d'état aux limites " S_I, F, i " et la dynamique du système $f(S, u)$ pour qu'on puisse générer l'ensemble des trajectoires " u_n " :

$$u_n(v, \omega) = \begin{bmatrix} u_0(v_0, \omega_0) \\ u_1(v_1, \omega_1) \\ \vdots \\ u_n(v_n, \omega_n) \end{bmatrix} \quad (3)$$

Sur le plan algorithmique, comme illustré dans l'algorithme 3, nous allons concevoir un ensemble de trajectoires dans l'espace de recherche à l'aide de la fonction **GENERATESEARCHSPACE** (). Cette fonction consiste à générer des paires d'états limites en utilisant l'équation (1) avec la fonction **GENERATEUNIFORMBOUNDARYSTATES** (). Une fois que l'ensemble des paires d'états limites est établi, la fonction **GENERATETRAJECTORY** () est appelée pour déterminer l'entrée de commande et la trajectoire qui satisfont la paire de contraintes de frontière d'état.

Algorithm 3: GENERATESEARCHSPACE

Input: $S_{ii}, \Delta t, m$ (Nombre de trajectoires)

Output: U_N

- 1: $\pi_N = \text{GENERATEBOUNDARYSTATE}(\Delta t, S_{ii})$
 - 2: **for** (i to m) **do**
 - 3: $U_N = \text{GENERATETRAJECTORY}(S_N, \Delta t)$
 - 4: **end for**
 - 5: **return** U_N
-

II. 4. 3 Profil des vitesses

Afin de générer les trajectoires nécessaires pour PassISS, nous présentons ici est une transformation mathématique dont les entrées sont la vitesse linéaire et la vitesse angulaire des roues et la sortie est la pose (position et orientation) du robot mobile dans son espace de travail.

II. 4. 3. 1 Profil de vitesse Linéaire

La vitesse initiale, la vitesse de déplacement et la vitesse terminale sont souvent déterminées à l'aide du profil de vitesse linéaire souhaité. La vitesse linéaire du robot peut être exprimée en fonction de la variation de ses coordonnées dans le repère universel, tel que :

tel que :

$$v = \frac{\sqrt{(\Delta x)^2 + (\Delta y)^2}}{\Delta t} \quad (4)$$

Tel que :

$$\Delta x = x_f - x_i$$

$$\Delta y = y_f - y_i$$

Avec : x_i, y_i comme points initiaux et x_f, y_f comme points finaux

II. 4. 3. 2 Profil de vitesse Angulaire

Les formes des trajectoires sont principalement déterminées par le profil de vitesse angulaire qui est définie par l'angle de rotation $\Delta\phi$ parcouru pendant le temps t . A partir de ces deux paramètres, la vitesse angulaire " ω " est calculée :

$$\omega = \frac{\Delta\phi}{\Delta t} \quad (5)$$

La génération des vitesses angulaires et linéaires (couples de contrôles) pour la génération des trajectoires est illustrée dans l'algorithme 3 selon la fonction **GENERATETRAJECTORY ()** (ligne 3 de l'algorithme 3).

Algorithm 4: GENERATETRAJECTORY

Input: $\Delta\phi, \Delta x, \Delta y, \Delta t, n$ (nombre de trajectoires)

Output: ω, v

- 1: for ($i=0$ to n) do
 - 2: $\omega = \Delta\phi / \Delta t$
 - 3: $v = \sqrt{\Delta x^2 + \Delta y^2} / \Delta t$
 - 4: return ω, v
-

II. 4. 4 Modèle odométrique

L'équation 1 est utilisée pour prédire les états des trajectoires. Pour éviter le calcul des intégrales, l'équation 1 peut être remplacée par l'équation du modèle odométrique.

L'odométrie est une technique de localisation qui permet de déterminer la position (x, φ) du robot par rapport au repère univers en intégrant des translations et rotations élémentaires. Nous appelons "odomètre" un dispositif qui permet de mesurer grâce à des rotations de roues, ses translations et rotations élémentaires [31].

Pour un déplacement entre deux instants d'échantillonnages k et $k+1$, qui leurs correspond respectivement les mesures odométrique $(x_{k+1}, y_{k+1}, \varphi_{k+1})$ et (x_k, y_k, φ_k)

Une première approche consiste à discrétiser le modèle cinématique (où v et ω sont les vitesses linéaire et angulaire) suivant :

$$\begin{aligned}\dot{x} &= -v \sin \varphi \\ \dot{y} &= v \cos \varphi \\ \dot{\varphi} &= \omega\end{aligned}\quad (6)$$

La discrétisation de ce modèle est alors :

$$\begin{aligned}x_{k+1} &= x_k - v \Delta t \sin \varphi_k \\ y_{k+1} &= y_k + v \Delta t \cos \varphi_k \\ \varphi_{k+1} &= \varphi_k + \omega \Delta t\end{aligned}\quad (7)$$

Avec :

$$\begin{aligned}\delta d &= v \times \Delta t \quad (\text{la distance} = \text{la vitesse} \times \text{le temps}) \\ \delta \varphi &= \omega \times \Delta t\end{aligned}$$

Nous obtenons donc en faisant apparaître les mesures odométrique δd et $\delta \varphi$, le modèle :

$$\begin{aligned}x_{k+1} &= x_k - \delta d \sin (\varphi_k + \delta \varphi / 2) \\ y_{k+1} &= y_k + \delta d \cos (\varphi_k + \delta \varphi / 2) \\ \varphi_{k+1} &= \varphi_k + \delta \varphi\end{aligned}\quad (8)$$

La génération des points de déplacement est montrée dans l'algorithme de la fonction **GENERATEUNIFORMBOUNDARYSTATES ()** (ligne 1 de l'algorithme 3) :

Algorithm 5: GENERATEUNIFORMBOUNDARYSTATES

Input: $x_{k,i}, y_{k,i}, \varphi_{k,i}, \Delta y, \Delta \varphi, n$ (nombre de trajectoires)

Output: $x_{k+1,n}, y_{k+1,n}, \varphi_{k+1,n}$

```

1: for ( i to n ) do
2:    $x_{k+1,i} = x_{k,i} - \delta d * \sin ( \varphi_{k,i} + \Delta \varphi / 2 )$ 
3:    $y_{k+1,i} = x_{k,i} + \delta d * \cos ( \varphi_{k,i} + \Delta \varphi / 2 )$ 
4:    $\varphi_{k+1,i} = \varphi_{k,i} + \Delta \varphi$ 
5: end for
6: return:  $x_{k+1,n}, y_{k+1,n}, \varphi_{k+1,n}$ 

```

Pour résumer, PassISS est basée sur la génération d'un ensemble de trajectoires, où chaque trajectoire est un ensemble d'états. Ces trajectoires sont testées par rapport à la sûreté de mouvement. Enfin, la trajectoire sélectionnée est celle qui est sûre. Dans la Figure II.10, on peut observer que l'état initial est "S0", le contrôle est "u", et l'état suivant est "S1". La sûreté de mouvement et sa vérification seront expliqués dans ce qui suit.

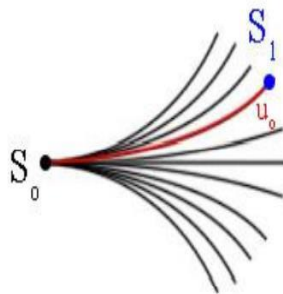


Figure II-10 Principe de fonctionnement de SafeISS

II. 5 La sûreté du mouvement

Lorsque des robots mobiles sont déployés à grande échelle en présence d'êtres humains, il devient essentiel de concevoir des approches d'évitement d'obstacles ou de navigation qui garantissent la sûreté du mouvement.

La garantie absolue de la sûreté du mouvement peut être obtenue dans le sens où, pour une position donnée du robot, il existe une trajectoire sans collision d'une durée infinie. Cependant, cette garantie n'est plus valable dans des situations où l'environnement est

dynamique et inconnu, et où la perception du robot est limitée à son champ de vision. Résoudre ce problème constitue un défi, car il est impossible de trouver une trajectoire sans collision d'une durée infinie. Une solution est de garantir une sûreté passive.

Définition 1 (sûreté passive) : Étant donné un modèle de l'environnement, un état s est considéré comme étant passivement sûr pour l'agent A s'il existe une trajectoire de freinage dont l'état initial est s et qui ne présente aucune collision jusqu'à l'arrêt complet de A [38].

II. 6 Les critères de la sûreté du mouvement :

Trois critères importants liés à la sûreté de mouvements dans les systèmes robotiques. Ces critères, liés au modèle du système robotique, au modèle de l'environnement et au processus de prise de décision, doivent être respectés pour éviter tout danger potentiel et prévenir les collisions futures [37.38]

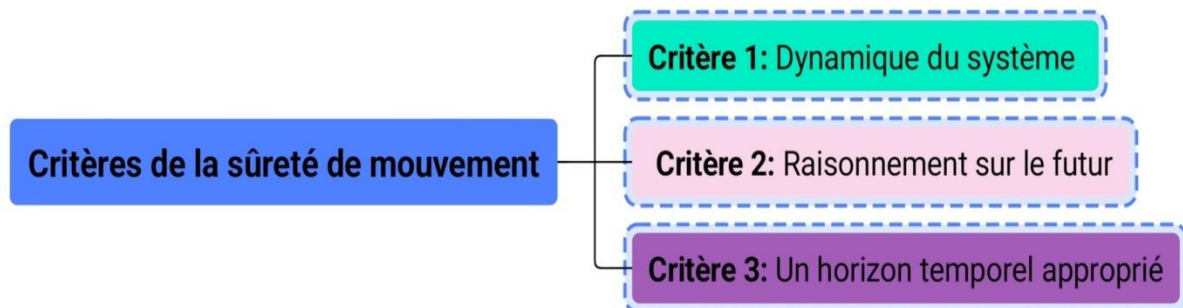


Figure II-11 Les critères de la sûreté du mouvement

II. 6. 1 Critère 1 : Contraintes liées au mouvement du robot

Le premier critère de sûreté concerne la prise en compte des limitations du mouvement du robot, telles que sa dynamique (accélération/décélération maximale, vitesse maximale, etc.). Prenons un exemple illustratif avec notre robot A , considéré comme une masse ponctuelle dans l'espace de configuration, caractérisée par sa position \mathbf{p}_A . Sa vitesse $|V_A| \leq V_{A_{\max}}$ et son accélération $|\alpha| \leq \alpha_{\max}$ et un obstacle statique B , caractérisé par sa position \mathbf{p}_B et l'espace C -obstacle (dans l'espace de configuration). A cause de sa dynamique,

A prend un temps minimal V_A/α_{\max} pour freiner et s'arrêter. La distance d correspondante est en fonction de l'accélération maximale α_{\max} , où :

$$d(V_A) = \frac{V_A^2}{2\alpha_{\max}} \quad (9)$$

Si la distance entre A et B est inférieure à une certaine valeur, peu importe ce qui se passe, une collision se produira dans le futur (figure II 12.a). Par conséquent, pour garantir la sûreté du mouvement, la condition suivante doit être vérifiée :

$$d_{A/B} > d(V_A) \quad (10)$$

Où la distance (V_A) correspond à la région interdite (la zone grise dans la figure II 12.a), alors que, $d_{A/B}$ représente la distance qui sépare le robot de la limite géométrique de l'obstacle. Par exemple, si l'obstacle B est de dimension $\rho \times \rho$, alors, $d_{A/B} = |P_B - P_A| - \frac{\rho}{2}$

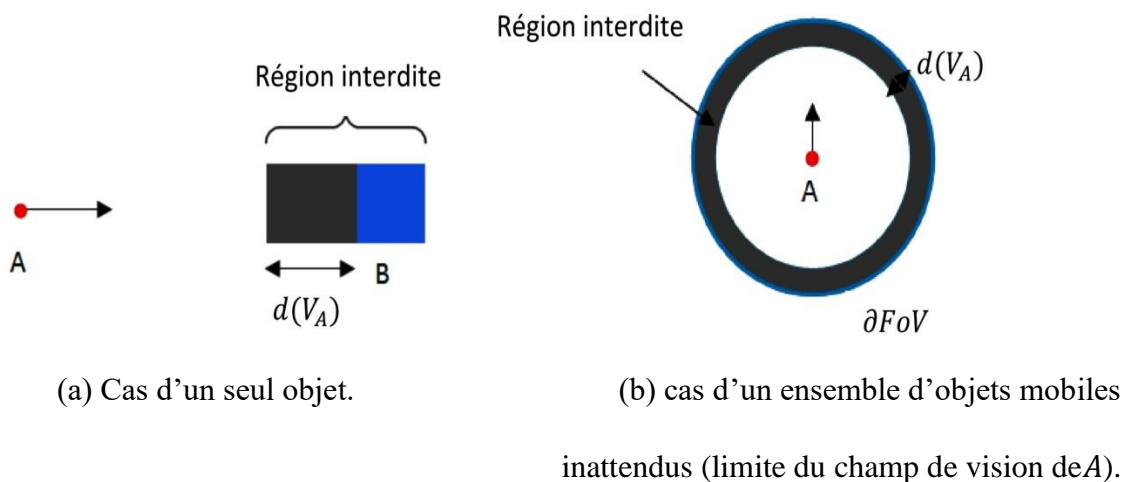


Figure II-12 Illustration de l'impact de la dynamique d'un robot (masse ponctuelle) sur sa sécurité

II. 6. 2 Critère 2 : Raisonnement sur le futur

Dans l'exemple de la Figure II 12.a, l'obstacle B est fixe. Cependant, lorsque l'obstacle B est mobile, il n'est pas suffisant de s'assurer qu'A ne se trouve pas dans la zone interdite (distance nécessaire pour permettre à A de freiner) pour garantir l'absence de collision. En d'autres termes, la condition (10) n'est plus vérifiée. Il devient alors nécessaire d'avoir une représentation de l'évolution future de l'environnement (un modèle du futur) pour garantir la sûreté. Cela constitue le deuxième critère de sûreté du mouvement.

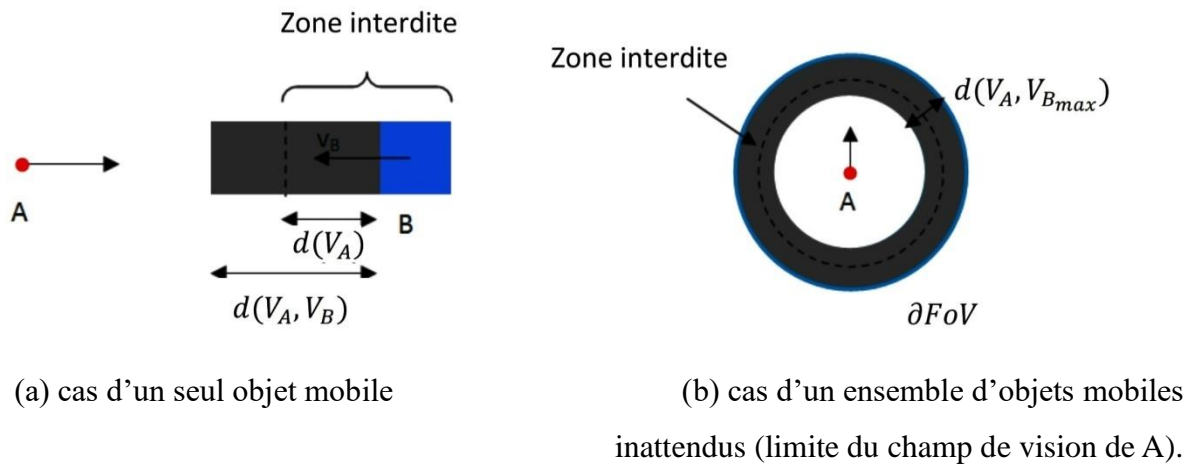


Figure II-13 L'influence de la dynamique d'un robot et celle des obstacles mobiles sur la sûreté de mouvement d'un robot.

Prenons l'exemple précédent mais cette fois-ci avec B un obstacle mobile de vitesse constante $V_B \leq V_{B_{max}}$. Dans ce cas, la dynamique de A et celle de B doivent être toutes les deux prises en compte. En effet, la distance séparant A et B doit être suffisante pour que A freine et s'arrête avant de rencontrer l'obstacle mobile. La sûreté est alors garantie en vérifiant la condition suivante (Figure II 13.a) :

$$d_{A/B} > d(V_A, V_B) \quad (11)$$

Avec

$$d(V_A, V_B) = \frac{V_A^2}{2\alpha_{max}} + V_B \frac{V_A}{\alpha_{max}}$$

Cette distance est la somme de la distance minimale nécessaire pour que A freine et la distance parcourue par B pendant le temps de freinage (V_A/α_{max}). Ce qui correspond à la région interdite.

Notons que dans le cas où aucune information sur les obstacles mobiles n'est disponible, la seule information pouvant être prise en compte est leur vitesse maximale. Par conséquent

$$V_B = V_{B_{max}}$$

II. 6. 3 Critère 3 : Un horizon temporel approprié

Pour prendre des décisions sur son mouvement futur, un système robotique doit raisonner sur un horizon temporel infini. Dans un environnement statique, il est sûr d'utiliser un horizon temporel fini correspondant au temps d'arrêt du système robotique.

Cependant, il est important de souligner que, de manière générale (notamment pour un système robotique en mouvement rapide et dans un environnement dynamique), les trois critères

s'appliquent et que la violation de l'un d'entre eux peut mettre le système robotique en danger, entraînant ainsi une collision à un moment donné dans le futur.

Pour garantir la sûreté de mouvement d'un robot mobile, il ne suffit pas d'utiliser uniquement des trajectoires sans collision. Le robot doit également maintenir un état pour lequel une trajectoire d'évitement est disponible.

II. 7 Vérifier la sûreté de mouvement

Rappelons le principe de PassISS, nous commencerons par échantillonner l'espace local du robot en utilisant la fonction *Determine Zone Admissible* () (algorithme 2). Ensuite, nous échantillonons l'espace des entrées. Par la suite, nous effectuons le test de sûreté grâce à la fonction *CheckSafety* (l'algorithme 5) pour vérifier la sûreté des mouvements du robot.

La sûreté peut être vérifiée grâce à la définition suivante :

Définition (contrôle p-sûr) : pour un état " S_0 " le contrôle " u " est p-sûr si " u " est sans collision et que l'état " S_1 " atteint à partir de " S_0 " en appliquant " u " est p-sûr. " S_1 " Est p-sûr alors il existe une trajectoire de freinage " \tilde{U}_b " dont l'état initial est " S_0 " et qui est sans collision jusqu'à l'arrêt [40]. L'algorithme de cette fonction est le suivant :

Algorithm5 : CHECKSAFETY

Input: $u_i, S_{\delta t}, Zone$

```

1: if (Zone [i] = 1) then // Occupied zone
2:   return false
3:else
4:   //Free Zone
5:   if ( $\exists \tilde{U}_b$  from  $S_{\delta t}$ ) then //  $\tilde{U}_b$  est une trajectoire de freinage sans collision
        //  $S_{\delta t}$  est l'état au quel  $u_i$  nous a conduit a partir de  $S_0$ 
6:     return true
7:   else
8:     return false
9:   end if
10:end if

```

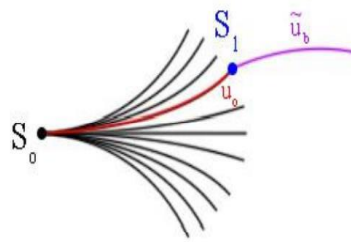


Figure II-14 Sélection trajectoire de freinage

II. 8 Le système d'aide à la conduite

Le système d'aide à la conduite permet au conducteur ou l'utilisateur de garder le contrôle tout en assurant un déplacement sûr. Cela signifie que le conducteur ne détecte pas un obstacle, estime mal sa position, ou ne le voit pas, un contrôle sûr est généré grâce à l'approche PassISS. Les étapes de l'algorithme restent les mêmes en ce qui concerne PassISS. Cependant, dans ce cas, les entrées de l'algorithme sont récupérées à partir de la commande du conducteur (ex. commande du joystick), notée $u_j = (v_j, \omega_j)$. Ainsi, ω est défini pour chaque commande en spécifiant leurs angles $\Delta\theta$. voir algorithme 6

Algorithm6: CHECK TRAGECTORY

Input: $u_d, u_N, Zone$

Output: u^*

- 1: Zone **Determine Admissible Zone** ();
 - 2: Check Safety ($u_d, Zone$); //return safe or not
 - 3: **if** u_d safe **then**
 $u^* = u_d$
 - 4: **else** { $u^* \leftarrow$ SELECT-NEARSEST-CTRL ($u_d u_N$);
 $u^* = u'^*$
 }
 - 5: **return** u^*
-

Qui veut dire que chaque commande génère des trajectoires n dans son angle $\Delta\theta$ (en le divisant sur le nombre des trajectoires $(\Delta\theta/n)$) où l'oméga Ω de chaque commande dans notre algorithme devient comme suit :

- 1) pour la commande avant :

$$\Omega_{avant} = ((\Delta\theta_{avant}/n)/dt) * q$$

(Avec $\Delta\theta$ avant l'angle de la commande avant)

- 2) pour la commande gauche :

$$\Omega_{agauche} = (\Delta\theta_{avant}/dt) + (((\Delta\theta_{agauche}/n) / dt) * q)$$

(Avec $\Delta\theta$ gauche l'angle de la commande à gauche)

- 3) Pour la commande droite :

Pour la commande du mouvement à droite, il suffit juste d'inverser la commande gauche

$$\Omega_{adroite} = -(\Delta\theta_{avant}/dt) + (((\Delta\theta_{adroite}/n) / dt) * -q)$$

(Avec $\Delta\theta$ droite l'angle de la commande à droite)

Après avoir sélectionné la commande, le reste de notre algorithme fonctionne de manière similaire pour générer les trajectoires et éviter les obstacles.

II. 9 Conclusion :

Dans ce chapitre nous avons présenté le système d'aide à la conduite proposée basé sur la méthode PassISS qui permet d'échantillonner l'espace de contrôles et de choisir une trajectoire sûre. Cette trajectoire sera appliquée dans le cas où le conducteur choisit un contrôle qui n'est pas sûr.

***Chapitre III* : Simulation et résultats**

III. 1 Introduction

Dans ce chapitre, nous présentons les simulations réalisées ainsi que les résultats obtenus en utilisant les méthodes et algorithmes décrits précédemment dans notre projet. Tout au long de notre processus, nous avons fait appel à divers outils, logiciels et plateformes, notamment ROS (Robot Operating System) et Gazebo.

III. 2 ROS (Robot Operating System) :

ROS (Robot Operating System) est un cadre de développement open source pour la robotique. Il fournit une infrastructure distribuée et flexible pour la communication entre les composants d'un système robotique. Les développeurs peuvent créer des applications robotiques modulaires en réutilisant des composants existants. ROS facilite également la gestion des tâches courantes de la robotique et dispose d'une vaste bibliothèque de packages et d'une communauté active [39].

III. 2. 1 Historique de ROS :

ROS, le système d'exploitation robotique (Robot Operating System), a été initialement développé en 2007 par le laboratoire de robotique de l'université de Stanford en réponse au besoin d'une plateforme de développement robuste et flexible pour les robots. Depuis lors, ROS a connu une adoption croissante dans la communauté de la robotique, devenant un cadre standard largement utilisé.

III. 2. 2 Architecture de ROS :

ROS adopte une architecture distribuée basée sur un système de communication asynchrone. Il utilise le concept de graphes de nœuds (nodes) pour représenter les différentes parties d'un système robotique. Les nœuds sont des entités autonomes qui communiquent entre eux en échangeant des messages. La communication entre les nœuds se fait via des sujets (topics), des services (services) et des actions (actions).

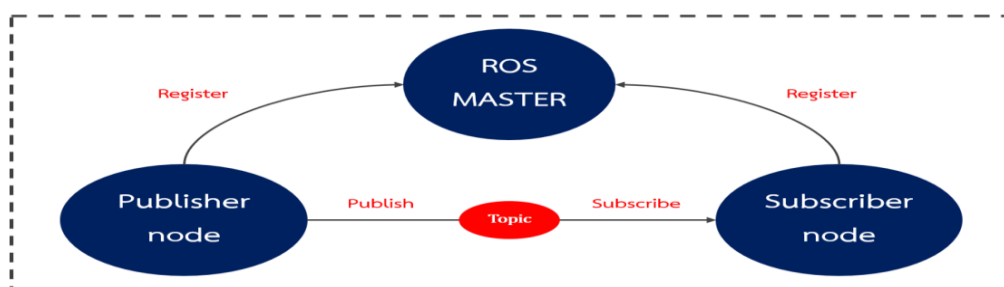


Figure III-1 Architecture de ROS

III. 2. 3 Composants clés de ROS :

Les composants clés de ROS comprennent :

- ✓ Nœuds (Nodes) : Les nœuds représentent les processus individuels dans le système robotique, tels que les capteurs, les actionneurs ou les algorithmes de traitement.
- ✓ Sujets (Topics) : Les sujets permettent aux nœuds de publier des messages et de s'abonner à des messages spécifiques. Cela permet une communication asynchrone entre les nœuds.
- ✓ Services (Services) : Les services permettent une communication synchrone entre les nœuds. Un nœud peut invoquer un service en envoyant une demande et attendre une réponse du nœud qui fournit le service.
- ✓ Actions (Actions) : Les actions sont similaires aux services, mais elles sont utilisées pour des tâches qui peuvent être longues et nécessitent un feedback périodique. Elles permettent d'effectuer des actions complexes tout en maintenant une communication asynchrone.

III. 2. 4 Communication et gestion des nœuds :

ROS fournit un système de communication transparent entre les nœuds grâce à sa couche de transport. Il gère également la découverte automatique des nœuds, permettant ainsi une configuration flexible et dynamique des systèmes robotiques.

III. 2. 5 Bibliothèques et outils intégrés à ROS :

ROS est livré avec un large éventail de bibliothèques et d'outils qui facilitent le développement de systèmes robotiques. Parmi les bibliothèques couramment utilisées, on trouve ROS Core (la base du système ROS), RViz (un outil de visualisation 3D), et MoveIt! (une bibliothèque de planification de mouvement). D'autres outils tels que roscpp (pour le développement en C++), rospy (pour le développement en Python) et rqt (un environnement de développement graphique) sont également disponibles.

III. 2. 6 Avantages de l'utilisation de ROS dans la robotique :

L'utilisation de ROS présente plusieurs avantages pour le développement de systèmes robotiques :

- Modularité : ROS permet de développer des systèmes robotiques modulaires en utilisant des nœuds autonomes. Cela facilite la réutilisation de modules et la mise à l'échelle des systèmes.

- Communication flexible : La communication asynchrone via les sujets, les services et les actions permet aux nœuds de communiquer de manière efficace et souple.
- Large communauté et écosystème : ROS bénéficie d'une communauté active qui partage des packages et des ressources. Cela facilite la collaboration et l'échange de connaissances dans le domaine de la robotique.
- Support multiplateforme : ROS est compatible avec différents systèmes d'exploitation, ce qui permet de développer des systèmes robotiques sur une variété de plates-formes matérielles.

III. 3 Gazebo (simulateur)

Gazebo est un simulateur de robotique open source et hautement extensible. Il fournit un environnement de simulation 3D réaliste dans lequel les développeurs peuvent modéliser, simuler et tester des robots et des scénarios complexes. Gazebo est largement utilisé dans la communauté de la robotique pour évaluer et valider les performances des robots avant de les déployer dans des environnements réels.

III. 3. 1 Fonctionnalités de Gazebo :

Gazebo offre de nombreuses fonctionnalités puissantes pour la simulation robotique, notamment :

- ✓ Graphiques 3D réalistes : Gazebo utilise un moteur de rendu graphique avancé pour créer des environnements simulés visuellement attrayants et réalistes.
- ✓ Simulation physique précise : Gazebo intègre un moteur de physique robuste qui simule avec précision les interactions entre les objets, les collisions, la gravité, les forces, etc.
- ✓ Support multi-robot : Gazebo permet la simulation de multiples robots interagissant dans le même environnement, ce qui facilite l'évaluation des interactions entre les robots et l'étude de scénarios multi-robot.
- ✓ Capteurs et actuators : Gazebo offre une gamme de capteurs et d'actuateurs simulés, tels que les capteurs de vision, les lidars, les radars, les moteurs, les vérins, etc. Cela permet aux développeurs de tester et d'évaluer les performances des systèmes de perception et de contrôle des robots.
- ✓ Développement de comportements : Gazebo permet de programmer les comportements des robots simulés en utilisant des bibliothèques de contrôle et des langages de programmation comme C++ et Python.

III. 3. 2 Modélisation de robots dans Gazebo :

Dans Gazebo, les robots sont modélisés à l'aide de fichiers de description qui décrivent leur géométrie, leurs propriétés physiques, leurs capteurs, leurs actuateurs, etc. Gazebo prend en charge différents formats de fichiers de modèles, tels qu'URDF (Unified Robot Description Format) et SDF (Simulation Description Format). Ces fichiers permettent aux développeurs de créer des modèles de robots précis et détaillés pour la simulation.

III. 3. 3 Environnements de simulation réalistes :

Gazebo offre la possibilité de créer des environnements de simulation réalistes en intégrant des éléments tels que des terrains, des bâtiments, des objets et des conditions environnementales (éclairage, météo, etc.). Cela permet aux développeurs de tester les performances des robots dans des scénarios variés et complexes, reproduisant ainsi des conditions réelles.

III. 3. 4 Intégration de ROS avec Gazebo :

L'une des grandes forces de Gazebo est son intégration étroite avec ROS. ROS et Gazebo peuvent être utilisés conjointement pour créer un environnement de développement complet pour la robotique. Les nœuds ROS peuvent communiquer avec les robots simulés dans Gazebo, permettant ainsi de tester et de valider les algorithmes de contrôle et de perception dans des scénarios simulés avant le déploiement sur des robots réels.

III. 3. 5 Avantages de l'utilisation de Gazebo pour la simulation robotique :

L'utilisation de Gazebo pour la simulation robotique présente plusieurs avantages :

- **Économie de coûts :** La simulation permet d'économiser des ressources financières en évitant d'avoir à construire des prototypes physiques coûteux.
- **Sécurité :** La simulation permet de tester des scénarios risqués ou dangereux pour les robots réels sans mettre en danger les personnes ou les équipements.
- **Développement itératif :** Gazebo facilite l'itération rapide du développement en permettant aux développeurs de tester et de modifier rapidement les algorithmes et les comportements des robots dans un environnement simulé.
- **Validation et débogage :** La simulation permet de valider les performances des robots dans des scénarios variés et de déboguer les erreurs plus facilement qu'avec des robots réels.

III. 4 Résultat de l'implémentation du système d'aide à la conduite

Afin de valider le système d'aide à la conduite proposé basé sur l'approche PassISS, il a été implémenté sur deux systèmes différents un robot différentiel et un tramway.

III. 4. 1 Cas d'un robot mobile différentiel

III. 4. 1. 1 Le model cinématique du robot

Dans notre premier cas d'étude, nous utilisons un robot différentiel (differential drive) qui comporte deux roues fixes commandées indépendamment, et une roue folle ajoutée à l'arrière du robot pour assurer sa stabilité. Avec un capteur LIDAR Hokuyo Le schéma du robot différentiel est présenté à la Figure III 2. On y a omis les roues folles, car elles n'interviennent pas dans la cinématique, mais elles assurent simplement l'équilibre [40].

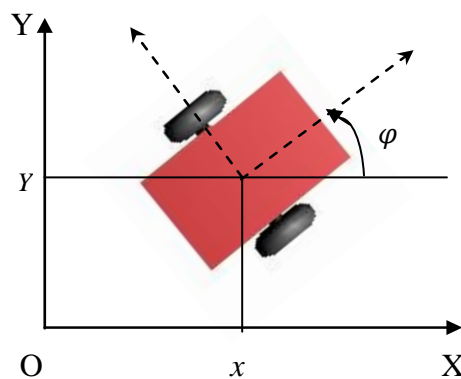


Figure III-2 Robot mobile différentiel

Ce type de robot est très répandu en raison de sa simplicité de construction et de ses propriétés cinématiques intéressantes, comme sa capacité à tourner sur lui-même. La figure III 3 présente le robot mobile utilisé dans notre cas.



Figure III-3 Le robot mobile dingo

III. 4. 1. 2 Modélisation

Centre instantané de rotation les roues motrices ayant même axe de rotation, le *CIR* du robot est un point de cet axe. Soit R le rayon de courbure de la trajectoire du robot, c'est-à-dire la distance du *CIR* au point O' (Figure III 4). Soit $2L$ la distance qui sépare les deux roues, et Ω la vitesse angulaire du robot par rapport au *CIR*. Alors les vitesses des roues droite et gauche, respectivement notées v_d et v_g vérifient :

$$v_d = (R + L)\Omega. \quad (12)$$

$$v_g = (R - L)\Omega. \quad (13)$$

Ce qui permet de déterminer R et Ω à partir des vitesses des roues :

$$\Omega = \frac{v_d - v_g}{2L} \quad (14)$$

$$R = L \frac{v_d + v_g}{v_d - v_g} \quad (15)$$

La vitesse linéaire v du robot au point O' est :

$$v = \frac{v_d + v_g}{2} \quad (16)$$

La vitesse de rotation du robot est égale à la vitesse de rotation autour du *CIR* :

$$\Omega = \varphi = \frac{v_d - v_g}{2L} \quad (17)$$

L'équation (4) permet de situer le CIR sur l'axe des roues. Par ailleurs ces équations expliquent deux propriétés particulières du mouvement des robots différentiels : si $v_d = v_g$, la vitesse angulaire Ω sera nulle et le rayon de courbure R est infinie donc le robot se déplace en ligne droite ; si $v_d = -v_g$, $\Omega \neq 0$ et R est nulle alors le robot effectue une rotation sur lui-même. Cependant dans le cas où $v_d \neq v_g$ le déplacement du robot est un virage à gauche ou à droite selon que v_d soit supérieur ou inférieur à v_g (dans une direction qui correspond à la vitesse inférieur).

L'utilisation de ce mode de locomotion, fournit une solution simple pour amener le robot d'une position à l'autre. C'est sans doute là une des raisons du succès de ce type de robot [40].

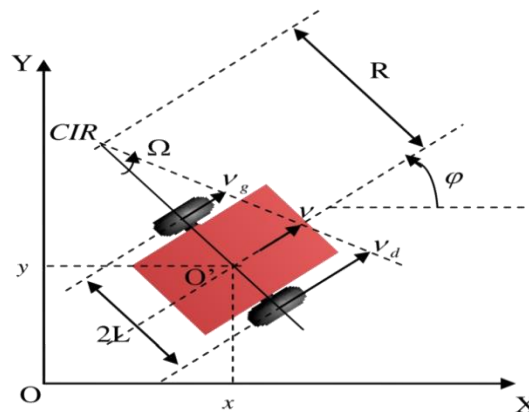


Figure III-4 centre instantanée de rotation CIR d'un robot mobile différentiel

Le modèle cinématique du robot différentiel est donnée par les équations suivantes :

$$\dot{x} = v \cos \varphi \quad (18)$$

$$\dot{y} = v \sin \varphi \quad (19)$$

$$\dot{\varphi} = \Omega \quad (20)$$

Ces équations relient la dérivée de la position (x, y, φ) du robot à la commande $u = (v, \Omega)^T$, avec φ rotation instantanée du robot par rapport au repère (O, X, Y) . De ce fait la position du robot est donnée par :

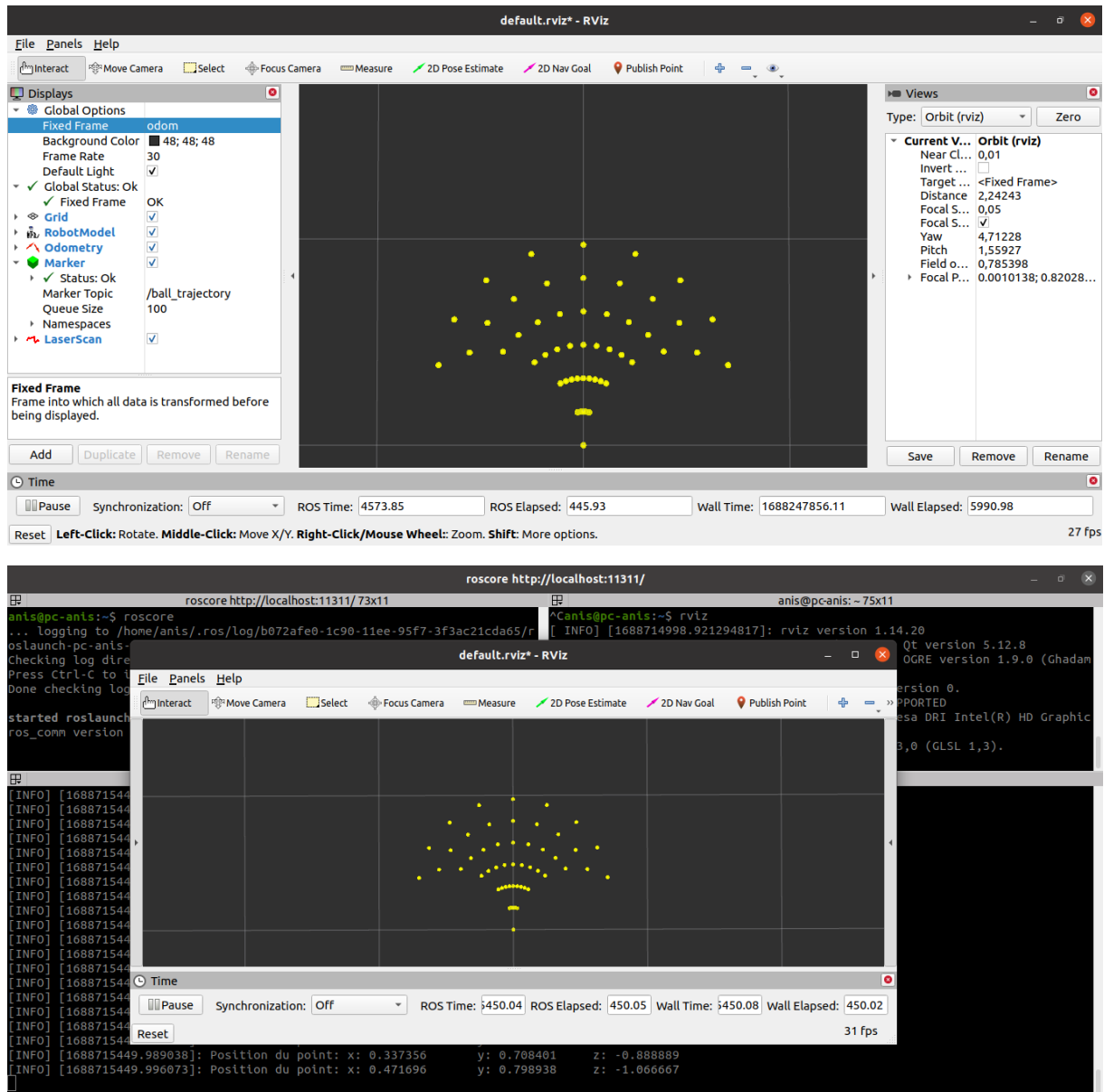
$$x(t) = \int_0^t v(\sigma) \cos(\varphi(\sigma)) d\sigma \quad (21)$$

$$y(t) = \int_0^t v(\sigma) \sin(\varphi(\sigma)) d\sigma \quad (22)$$

$$\varphi(t) = \int_0^t \Omega(\sigma) d\sigma \quad (23)$$

III. 4. 1. 3 La génération de trajectoire

Nous avons utilisé **Rviz** pour simuler les résultats antérieurs sur **ROS**. Pour ce faire, nous avons créé un fichier.py représentant notre nœud dans l'espace de travail, que nous avons ensuite lancé avec la commande "**roslaunch**". Dans Rviz, nous avons utilisé des sphères markers pour afficher toutes les coordonnées $S_{k+1}(x_{k+1},y_{k+1},\theta_{k+1})$ (ligne 1 de l’algorithme 5)de chaque trajectoire. La figure III 5 représente l’ensemble de trajectoires générées.



```

anis@pc-anis:~$ rosrund motion_plan trajectory.py
Pour une trajectoire avec n=9 points on a :
Vitesse linéaire = 0.2m/s et un temps de déplacement delta t = 2s
[INFO] [1688709615.771426, 0.000000]: Coordonees du point initial : x: 0.000000 y: 0.000000 z: 0.000000
[INFO] [1688709615.787863, 0.000000]: Avec une vitesse angulaire Omega = -0.296296:
[INFO] [1688709615.795989, 0.000000]: Position du point: x: 0.014381 y: 0.161360 z: -0.177778
[INFO] [1688709615.803587, 0.000000]: Position du point: x: 0.057071 y: 0.317634 z: -0.355556
[INFO] [1688709615.808757, 0.000000]: Position du point: x: 0.126724 y: 0.463896 z: -0.533333
[INFO] [1688709615.813404, 0.000000]: Position du point: x: 0.221144 y: 0.595535 z: -0.711111
[INFO] [1688709615.817969, 0.000000]: Position du point: x: 0.337356 y: 0.708401 z: -0.888889
[INFO] [1688709615.823425, 4720.715000]: Position du point: x: 0.471696 y: 0.798938 z: -1.066667
[INFO] [1688709615.829230, 4720.716000]: Avec une vitesse angulaire Omega = -0.222222:
[INFO] [1688709615.837537, 4720.716000]: Position du point: x: 0.010792 y: 0.161640 z: -0.133333
[INFO] [1688709615.842739, 4720.717000]: Position du point: x: 0.042976 y: 0.320411 z: -0.266667
[INFO] [1688709615.846920, 4720.717000]: Position du point: x: 0.095982 y: 0.473494 z: -0.400000
[INFO] [1688709615.852995, 4720.717000]: Position du point: x: 0.168868 y: 0.618172 z: -0.533333
[INFO] [1688709615.857569, 4720.718000]: Position du point: x: 0.260340 y: 0.751876 z: -0.666667
[INFO] [1688709615.864428, 4720.718000]: Position du point: x: 0.368774 y: 0.872234 z: -0.800000
[INFO] [1688709615.874158, 4720.721000]: Avec une vitesse angulaire Omega = -0.148148:
[INFO] [1688709615.881965, 4720.723000]: Position du point: x: 0.007198 y: 0.161840 z: -0.088889
[INFO] [1688709615.892130, 4720.725000]: Position du point: x: 0.028734 y: 0.322402 z: -0.177778
[INFO] [1688709615.911764, 4720.731000]: Position du point: x: 0.064438 y: 0.480419 z: -0.266667
[INFO] [1688709615.918891, 4720.734000]: Position du point: x: 0.114029 y: 0.634642 z: -0.355556
[INFO] [1688709615.925166, 4720.735000]: Position du point: x: 0.177115 y: 0.783854 z: -0.444444
[INFO] [1688709615.941262, 4720.741000]: Position du point: x: 0.253197 y: 0.926876 z: -0.533333
[INFO] [1688709615.954912, 4720.745000]: Avec une vitesse angulaire Omega = -0.074074:
[INFO] [1688709615.984350, 4720.747000]: Position du point: x: 0.003600 y: 0.161960 z: -0.044444
[INFO] [1688709616.012248, 4720.751000]: Position du point: x: 0.014392 y: 0.323600 z: -0.088889
[INFO] [1688709616.028214, 4720.753000]: Position du point: x: 0.032355 y: 0.484601 z: -0.133333
[INFO] [1688709616.037208, 4720.756000]: Position du point: x: 0.057453 y: 0.644645 z: -0.177778
[INFO] [1688709616.043986, 4720.756000]: Position du point: x: 0.089638 y: 0.803416 z: -0.222222
[INFO] [1688709616.067646, 4720.760000]: Position du point: x: 0.128844 y: 0.960600 z: -0.266667
[INFO] [1688709616.076518, 4720.762000]: Avec une vitesse angulaire Omega = 0.000000:
[INFO] [1688709616.089775, 4720.764000]: Position du point: x: 0.000000 y: 0.162000 z: 0.000000
[INFO] [1688709616.089775, 4720.764000]: Position du point: x: 0.000000 y: 0.162000 z: 0.000000
[INFO] [1688709616.103894, 4720.768000]: Position du point: x: 0.000000 y: 0.324000 z: 0.000000
[INFO] [1688709616.122757, 4720.771000]: Position du point: x: 0.000000 y: 0.486000 z: 0.000000
[INFO] [1688709616.132493, 4720.773000]: Position du point: x: 0.000000 y: 0.648000 z: 0.000000
[INFO] [1688709616.138230, 4720.774000]: Position du point: x: 0.000000 y: 0.810000 z: 0.000000
[INFO] [1688709616.155157, 4720.775000]: Position du point: x: 0.000000 y: 0.972000 z: 0.000000
[INFO] [1688709616.162611, 4720.778000]: Avec une vitesse angulaire Omega = 0.074074:
[INFO] [1688709616.184099, 4720.782000]: Position du point: x: -0.003600 y: 0.161960 z: 0.044444
[INFO] [1688709616.190713, 4720.782000]: Position du point: x: -0.014392 y: 0.323600 z: 0.088889
[INFO] [1688709616.198161, 4720.784000]: Position du point: x: -0.032355 y: 0.484601 z: 0.133333
[INFO] [1688709616.211053, 4720.785000]: Position du point: x: -0.057453 y: 0.644645 z: 0.177778
[INFO] [1688709616.228362, 4720.787000]: Position du point: x: -0.089638 y: 0.803416 z: 0.222222
[INFO] [1688709616.234533, 4720.789000]: Position du point: x: -0.128844 y: 0.960600 z: 0.266667
[INFO] [1688709616.241022, 4720.790000]: Avec une vitesse angulaire Omega = 0.148148:
[INFO] [1688709616.260918, 4720.793000]: Position du point: x: -0.007198 y: 0.161840 z: 0.088889
[INFO] [1688709616.275035, 4720.796000]: Position du point: x: -0.028734 y: 0.322402 z: 0.177778
[INFO] [1688709616.289314, 4720.799000]: Position du point: x: -0.064438 y: 0.480419 z: 0.266667
[INFO] [1688709616.299083, 4720.800000]: Position du point: x: -0.114029 y: 0.634642 z: 0.355556
[INFO] [1688709616.305939, 4720.802000]: Position du point: x: -0.177115 y: 0.783854 z: 0.444444
[INFO] [1688709616.312542, 4720.803000]: Position du point: x: -0.253197 y: 0.926876 z: 0.533333
[INFO] [1688709616.324254, 4720.805000]: Avec une vitesse angulaire Omega = 0.222222:
[INFO] [1688709616.334405, 4720.806000]: Position du point: x: -0.010792 y: 0.161640 z: 0.133333
[INFO] [1688709616.346174, 4720.810000]: Position du point: x: -0.042976 y: 0.320411 z: 0.266667
[INFO] [1688709616.352367, 4720.811000]: Position du point: x: -0.095982 y: 0.473494 z: 0.400000
[INFO] [1688709616.359185, 4720.812000]: Position du point: x: -0.168868 y: 0.618172 z: 0.533333
[INFO] [1688709616.370910, 4720.814000]: Position du point: x: -0.260340 y: 0.751876 z: 0.666667
[INFO] [1688709616.382631, 4720.816000]: Position du point: x: -0.368774 y: 0.872234 z: 0.800000
[INFO] [1688709616.388534, 4720.818000]: Avec une vitesse angulaire Omega = 0.296296:
[INFO] [1688709616.396324, 4720.819000]: Position du point: x: -0.014381 y: 0.161360 z: 0.177778
[INFO] [1688709616.406322, 4720.820000]: Position du point: x: -0.057071 y: 0.317634 z: 0.355556
[INFO] [1688709616.417443, 4720.821000]: Position du point: x: -0.126724 y: 0.463896 z: 0.533333
[INFO] [1688709616.433183, 4720.825000]: Position du point: x: -0.221144 y: 0.595535 z: 0.711111
[INFO] [1688709616.444067, 4720.826000]: Position du point: x: -0.337356 y: 0.708401 z: 0.888889
[INFO] [1688709616.451731, 4720.828000]: Position du point: x: -0.471696 y: 0.798938 z: 1.066667

```

Figure III-5 Simulation de génération des trajectoires sur RVIZ avec des sphères List points

Les conditions de tests sont les suivant :

❖ **Caractéristiques du robot Dingo**

Nom	Vitesse maximale (v-x) [m/s]	Vitesse de rotation maximale (θ -y) [rad/s]	Rayonne [m]	Arrêt d'urgence ¹	Portée laser [m]
Dingo	1.3	(4.0)	0,378	faire	30,0

Tableau 3-6 les caractéristiques du robot dingo

III. 4. 2 Scénario de test

C'est un environnement urbain contenant les éléments suivants : une aire de jeux, un cabinet d'avocats, un magasin, un parking, des maisons, une fontaine, un bureau de poste, une station-service, une caserne des pompiers, un supermarché, un parc d'attractions, des arbres et une école.

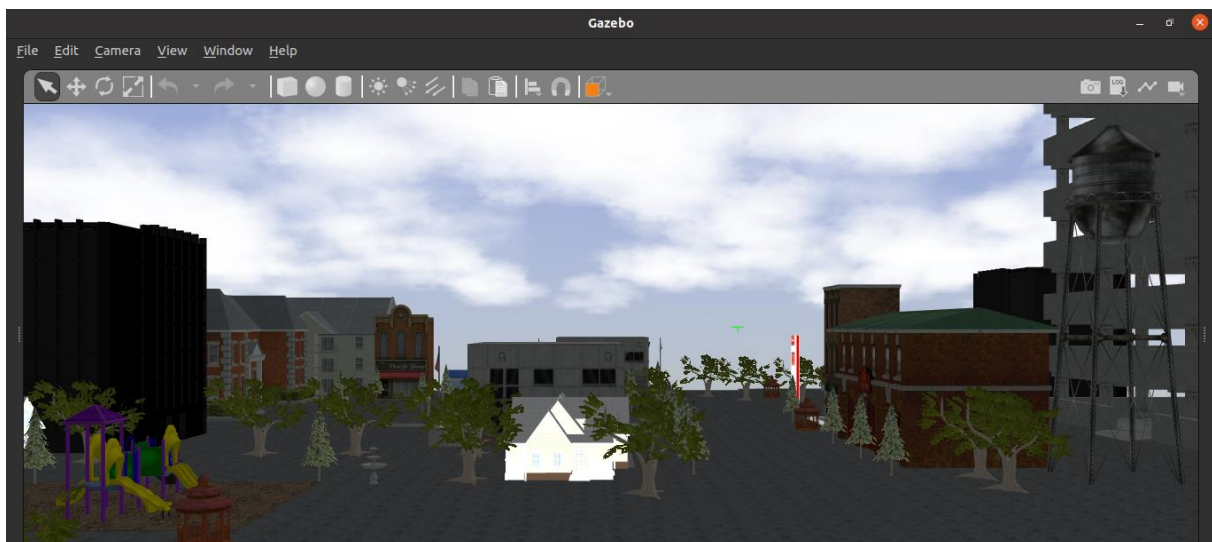


Figure III-6 Scénario de test dans le simulateur Gazebo.

III. 4. 2. 1 Simulation des zones de perception du capteur laser

Les zones découpées présentées en bleu dans les figures (Figure III 7)et (Figure III 8)sont les zones de perception construites à partir des données du capteur laser, tel qu'illustré dans l'algorithme de la fonction Determine Admissible Zone() (algorithme 2).

Le robot est capable de générer et d'exécuter différentes trajectoires en lui fournissant plusieurs contrôles (ω, v), avec chaque contrôle correspondant à chaque zone. Ce processus est décrit en détail dans le chapitre précédent, où nous avons expliqué comment le robot manipule ces contrôles pour créer et suivre les différentes trajectoires.

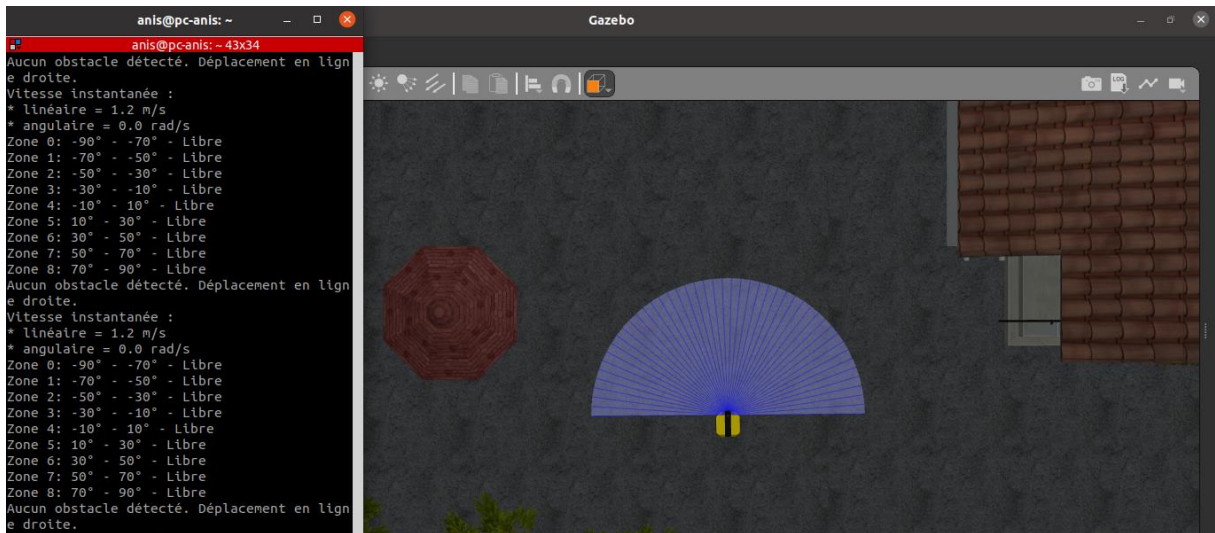


Figure III-7 Résultat de la simulation des zones de perception sur Gazebo.

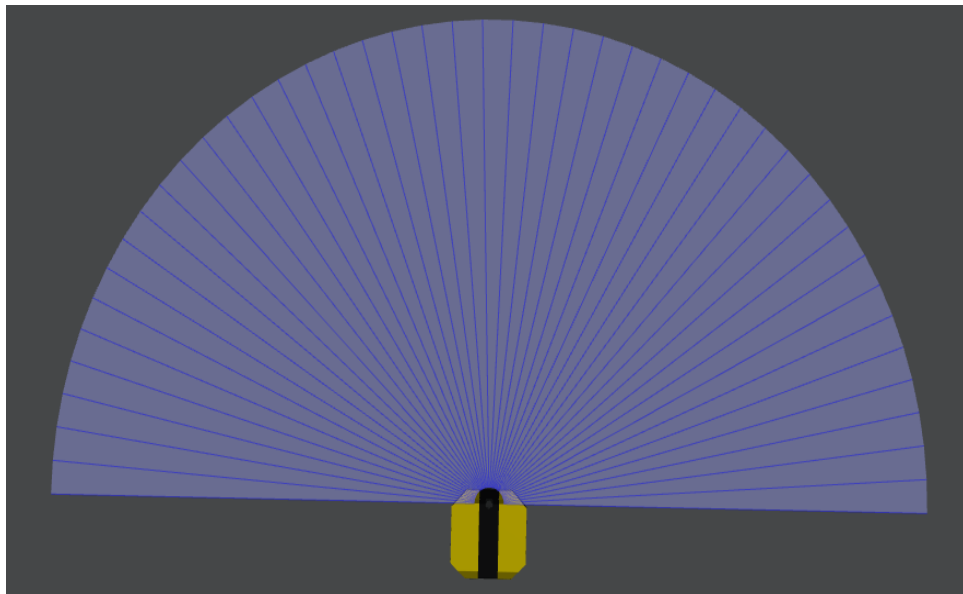


Figure III-8 Résultat de la simulation des zones de perception sur Gazebo (un zoom sur le robot).

III. 4. 3 Navigation sans obstacle

Comme on le voit dans la figure (Figure III 9), le robot se déplace d'un point initial vers un autre point

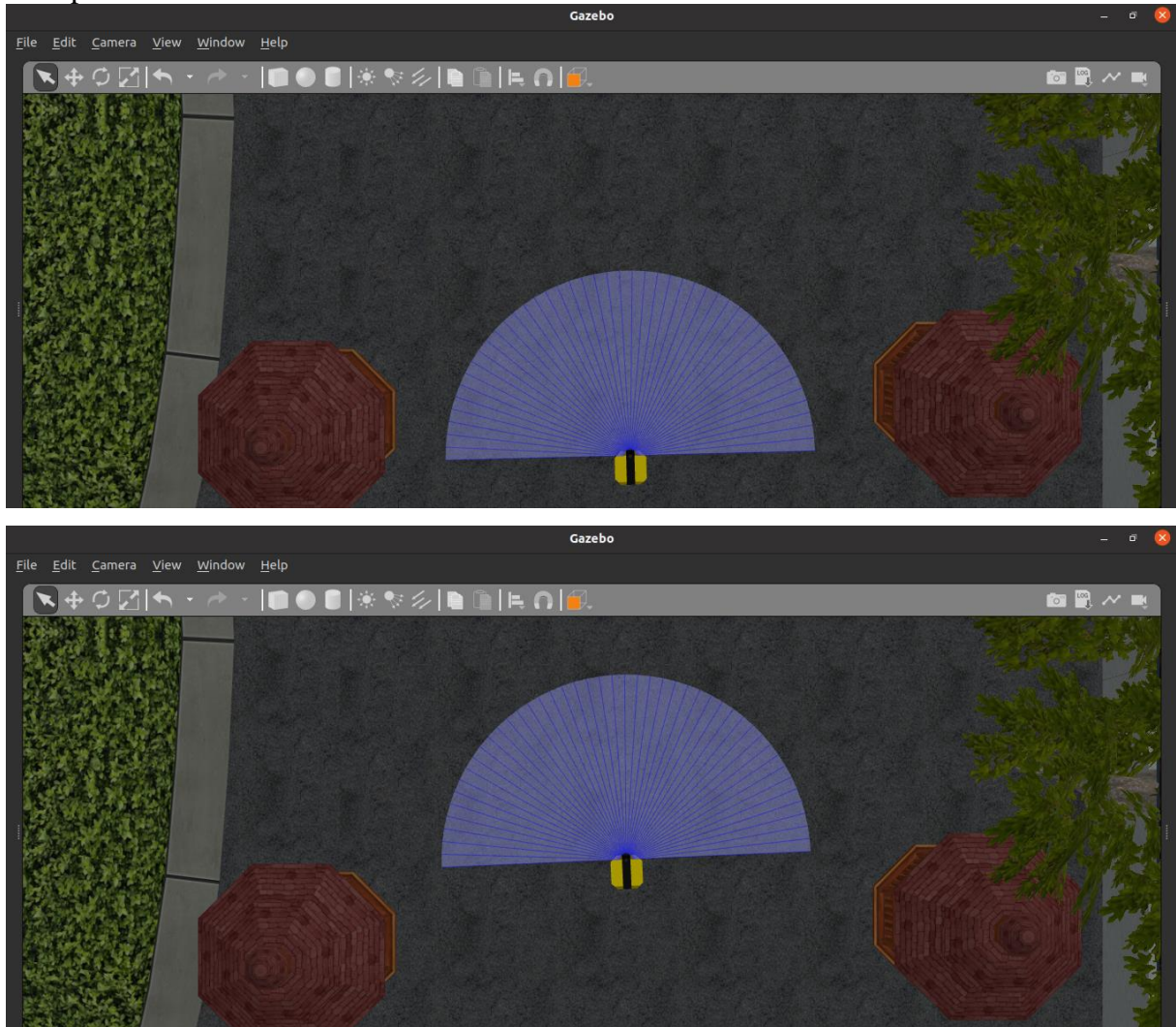


Figure III-9 Résultat de la simulation des zones de perception sur Gazebo(avec déplacement de robot).

III. 4. 3. 1 Détection d'un obstacle grâce au capteur laser

Lorsque le capteur détecte la présence d'un obstacle, la zone où se trouve cet obstacle est graphiquement modifiée en changeant de couleur pour devenir d'un bleu foncé. Cette zone est alors considérée comme occupée et inappropriée (non admissible) (voir Figure III 10).

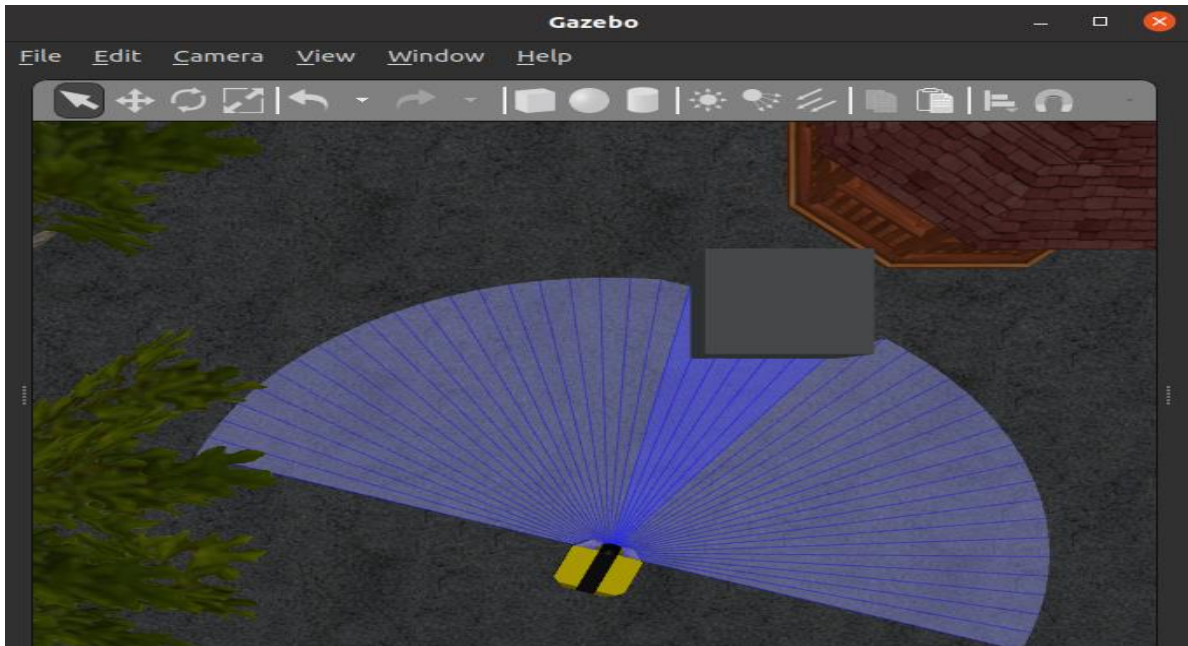


Figure III-10 Détection d'un obstacle grâce au capteur laser

III. 4. 4 Détection et évitement de plusieurs obstacles

Pour permettre au robot d'éviter plusieurs obstacles, il doit d'abord les détecter à l'avance en calculant la distance entre son point actuel et chaque obstacle. Cela garantit la sécurité et évite toute collision potentielle. Ensuite, le robot sélectionne une trajectoire sûre pour les éviter.

a. Détection

Dans ce cas, le robot a détecté deux obstacles situés dans deux zones différentes voir la figure (Figure III 11)

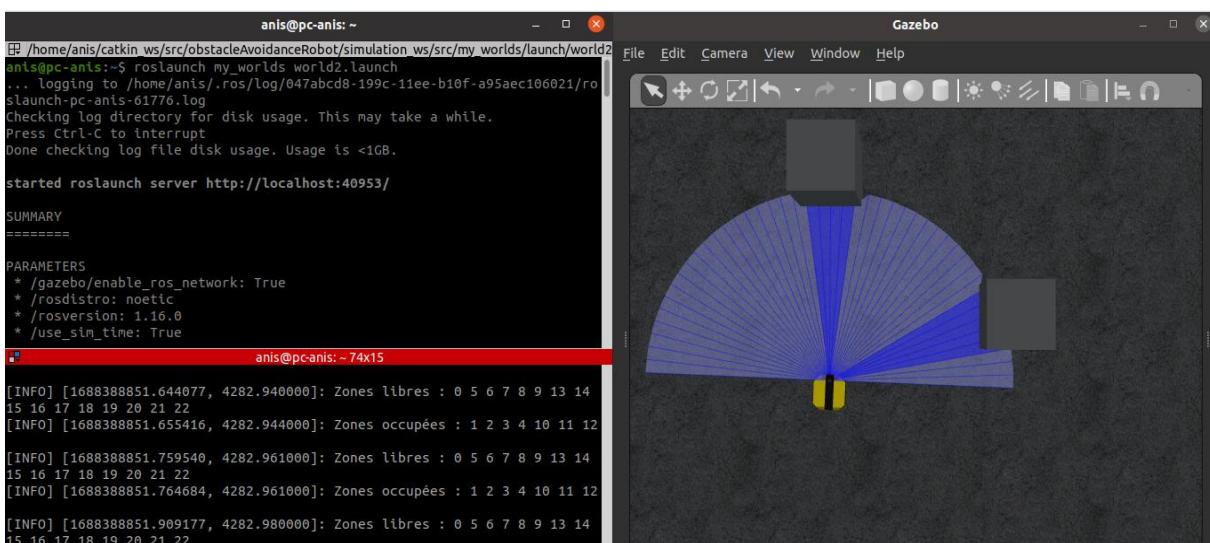


Figure III-11 Détection de deux obstacles par le robot

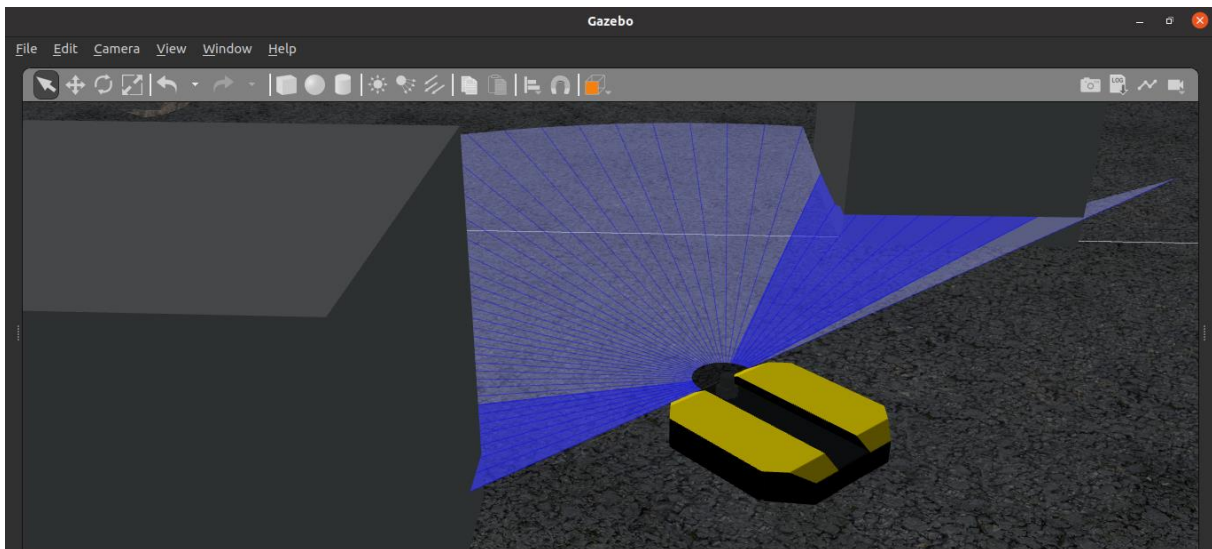
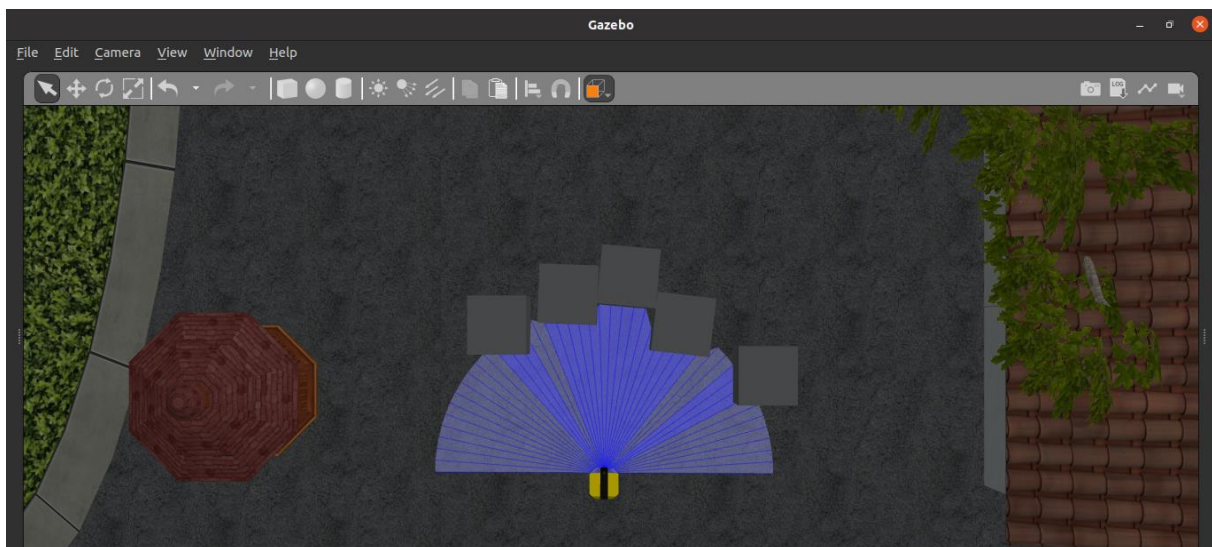
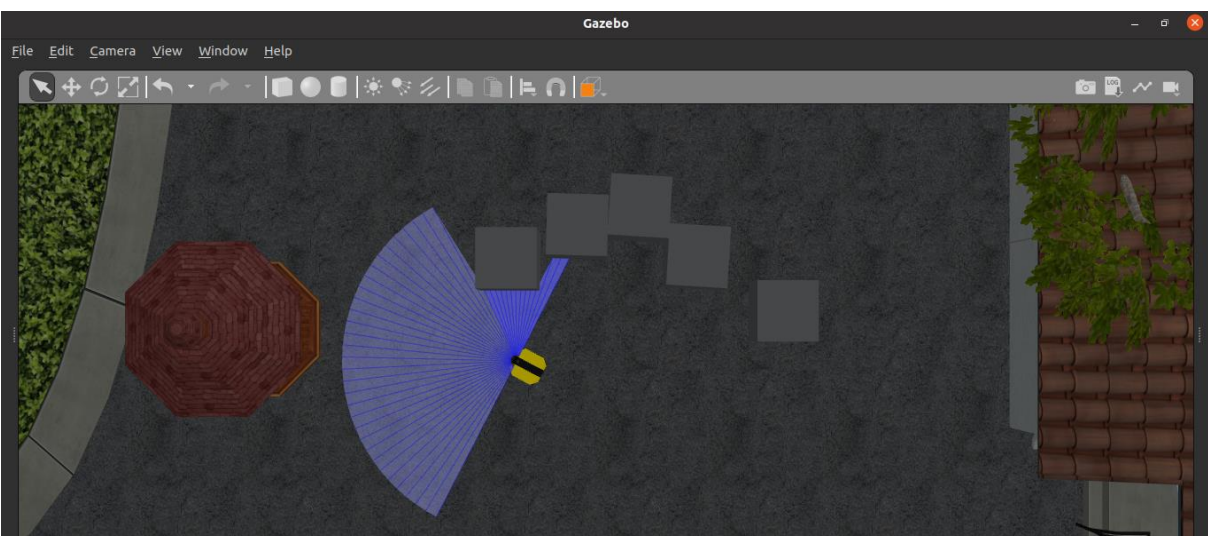
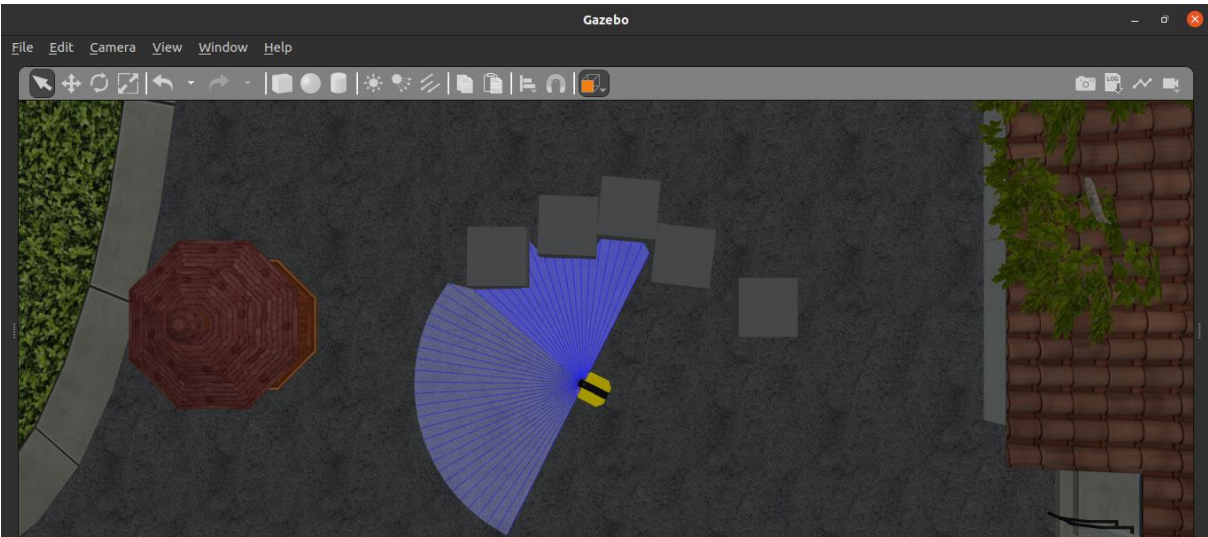
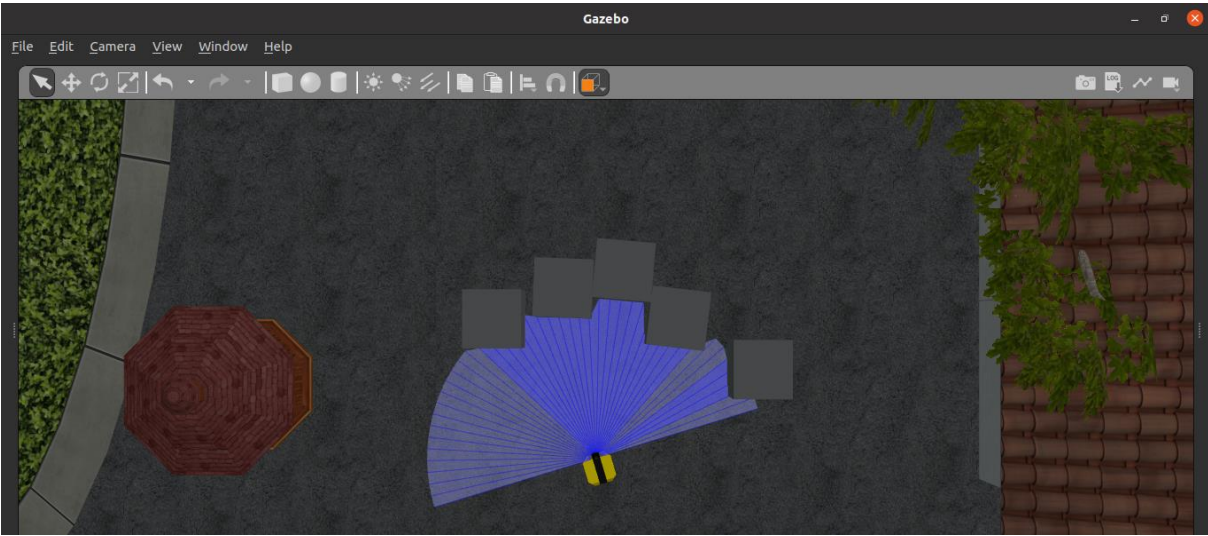


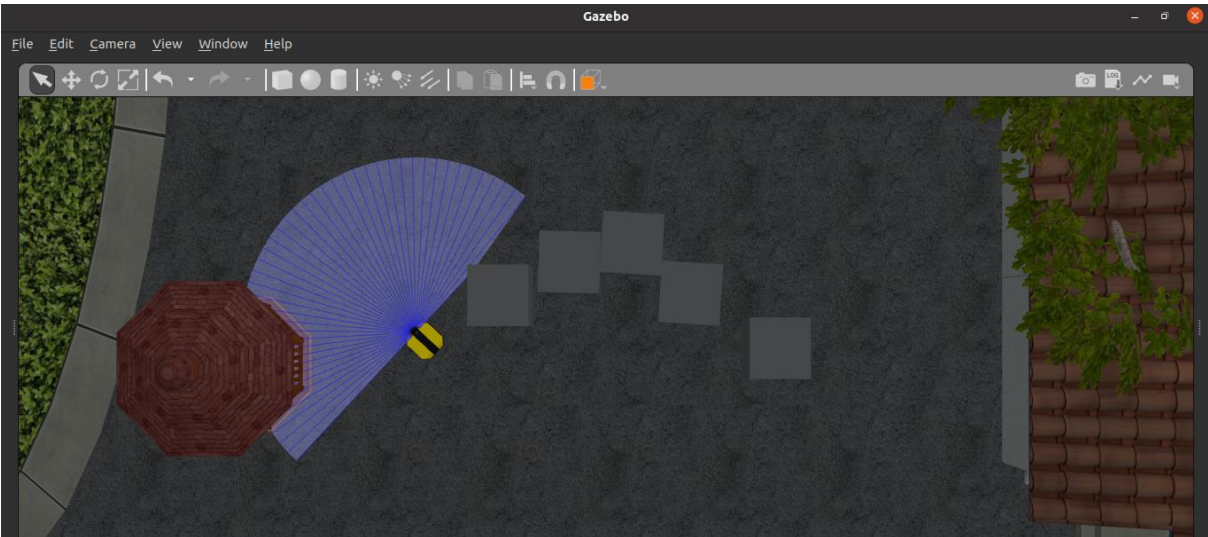
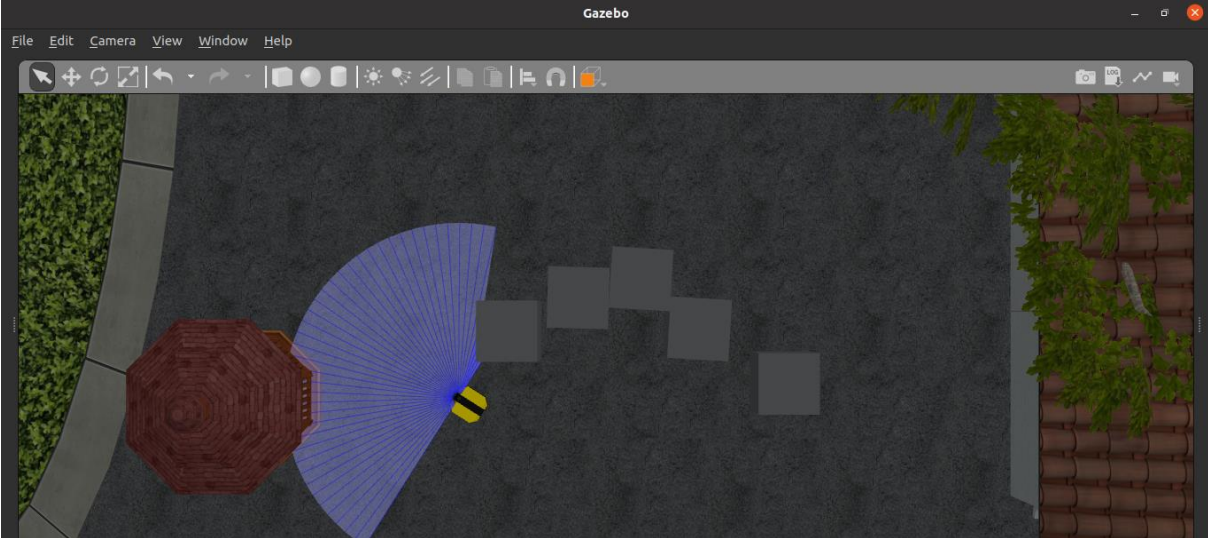
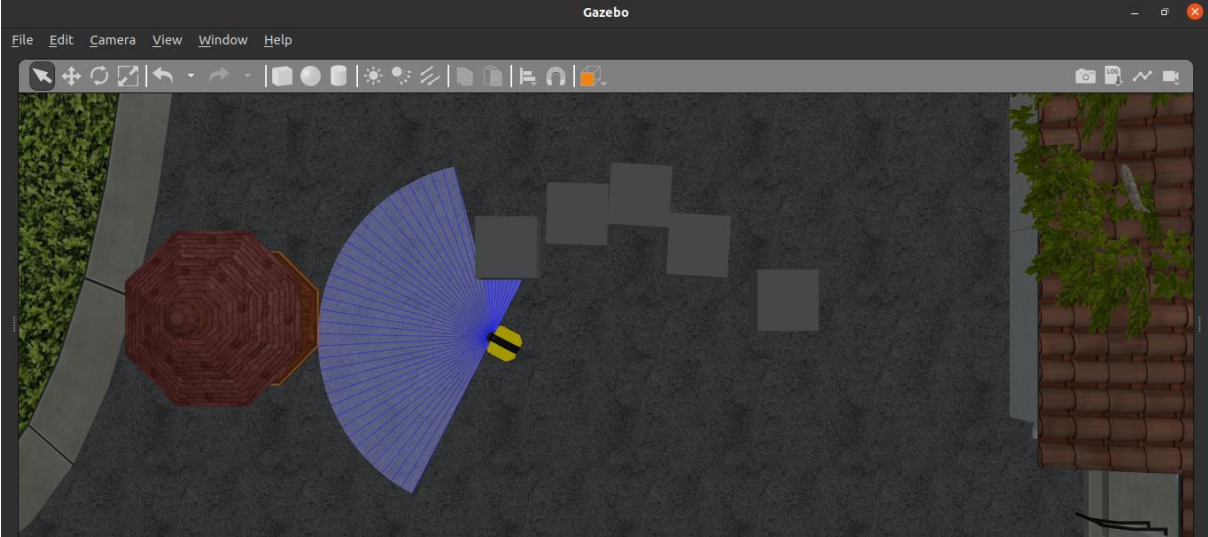
Figure III-12Détection de deux obstacles par le robot (un zoom sur le robot).

b. Evitement

Le robot choisit une trajectoire lui permettant d'éviter les obstacles de l'environnement et poursuit son parcours avec une vitesse linéaire de 0.7 m/s (Figure III 13).







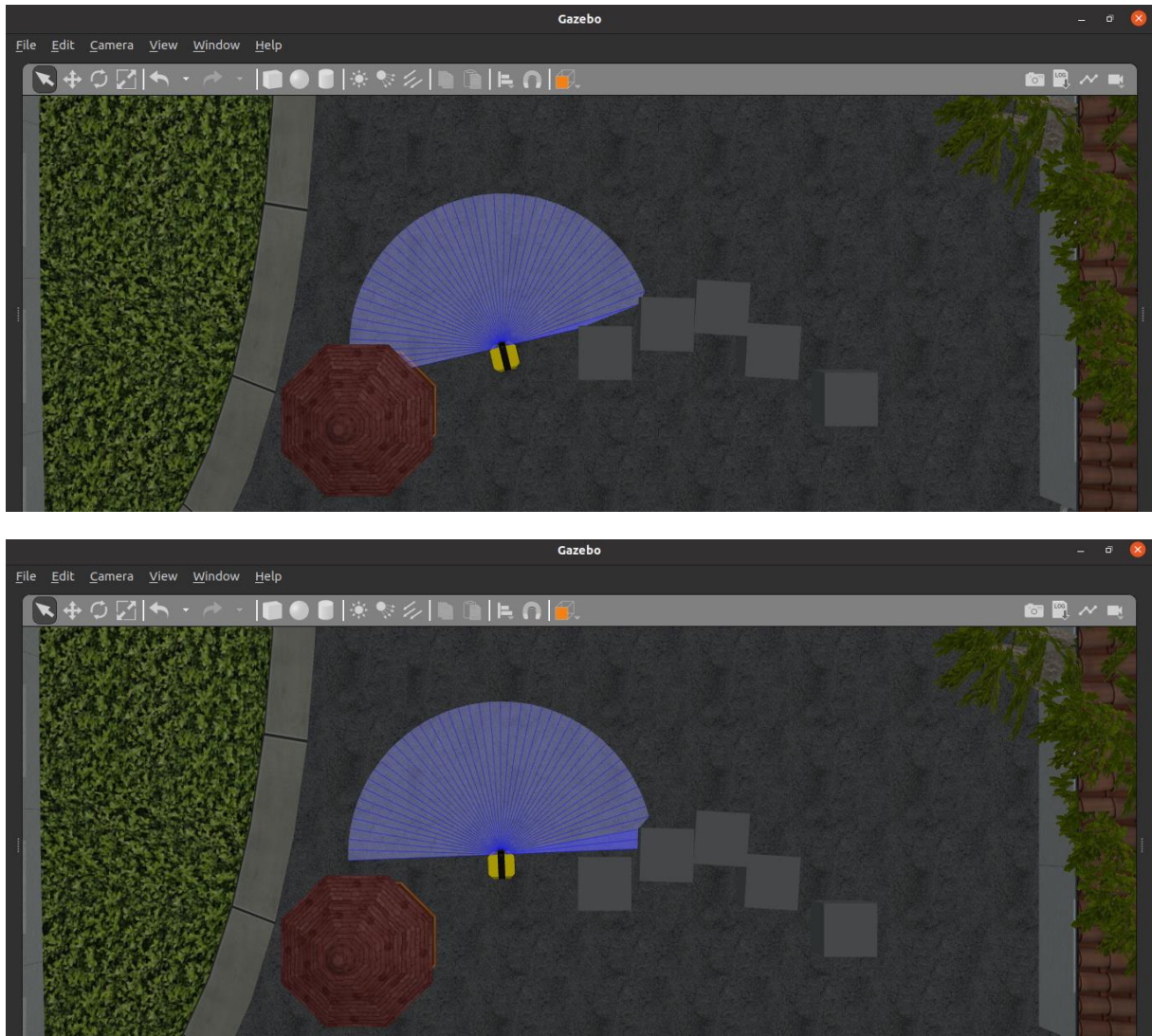


Figure III-13Le processus d'évitement des obstacles par le robot

III. 5 Le système d'aide à la conduite

Comme nous l'avons déjà expliqué dans les chapitres précédents, chaque commande inclue une vitesse linéaire v et une vitesse angulaire ω défini par un angle $\Delta\theta$ qui est divisé sur les nombres de trajectoires « n » pour générer les trajectoires.

Défini par un angle $\Delta\theta$ qui est divisé sur les nombres de trajectoires n pour générer les trajectoires.

Nous avons généré les trajectoires de chaque commande sur Rviz en utilisant les sphères markers :

Avec $n = 9$ (le nombre de trajectoires à générer) et un $\Delta\theta=0.4$ (radians) et donc l'oméga est calculé comme suit.

-pour la commande avant:

$$\Omega_{avant} = ((\Delta\theta_{avant}/n/dt) * q$$

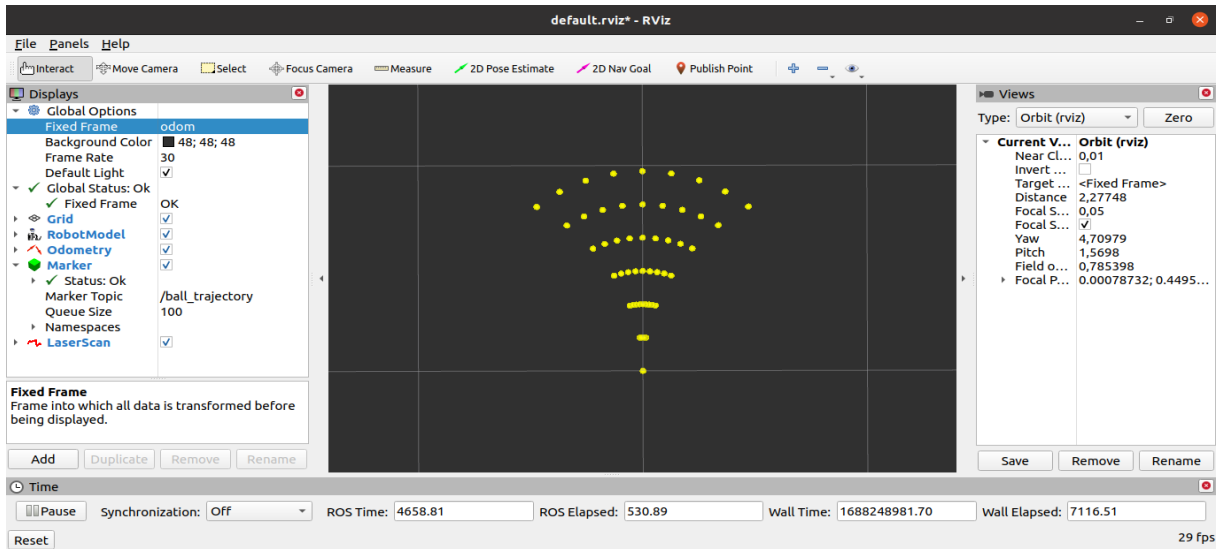


Figure III-14 Simulation des trajectoires générées par la commande avant sur Rviz

-pour la commande gauche:

$$\Omega_{agauche} = (\Delta\theta_{avant}/dt) + (((\Delta\theta_{agauche}/n) /dt) * q)$$

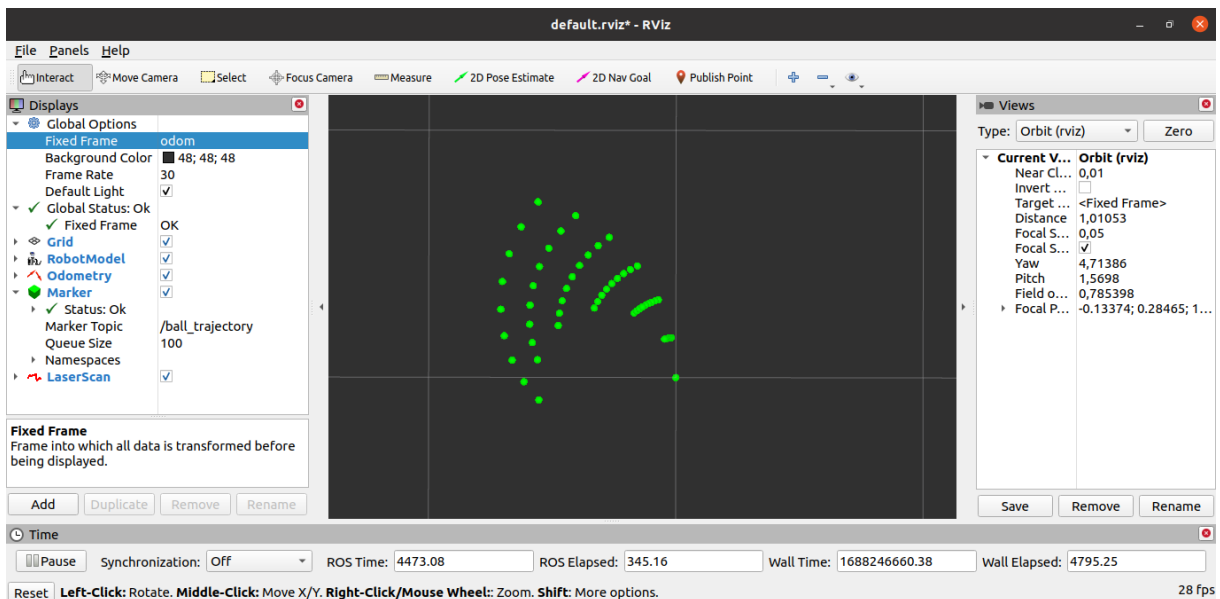


Figure III-15 Simulation des trajectoires générées par la commande gauche sur Rviz

-Pour la commande droite:
 Pour la commande du mouvement à droite, il suffit jute d'inverser la commande gauche

$$\Omega_{adroite} = -(\Delta\theta_{avant}/dt) + (((\Delta\theta_{adroite}/n) /dt) * -q)$$

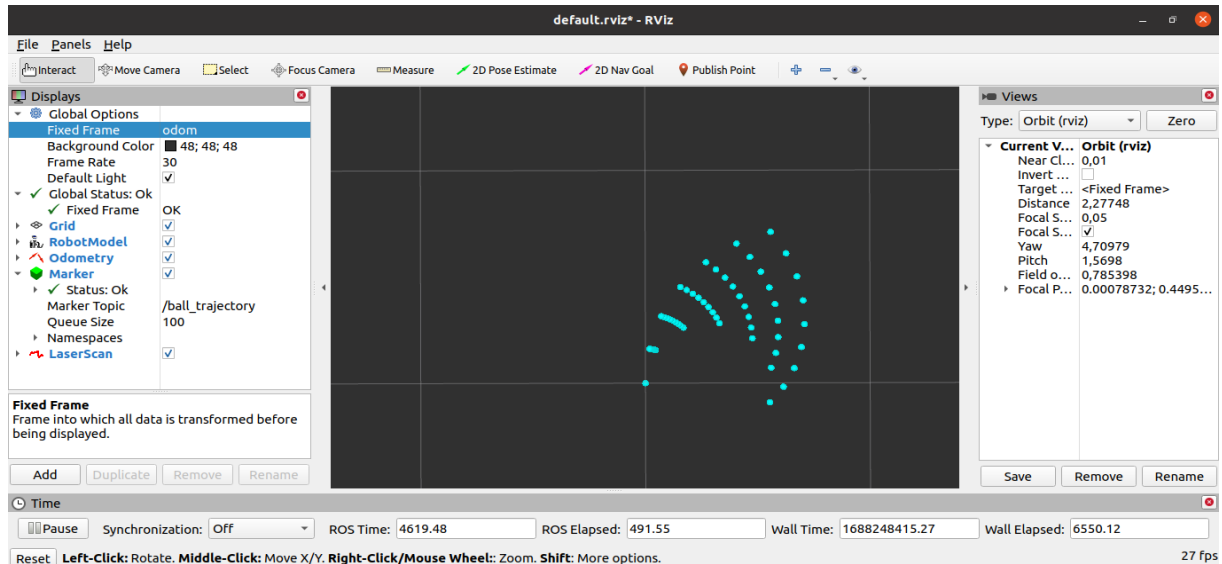


Figure III-16 Simulation des trajectoires générées par la commande droite sur Rviz

III. 6 Cas d'un tramway

III. 6. 1. 1 Le modèle cinématique du tramway

En ce qui concerne notre deuxième cas d'étude, bien que les robots omnidirectionnels et les tramways puissent être considérés comme holonomes du point de vue cinématique, le tramway est donc un robot omnidirectionnel.

III. 6. 1. 2 Description

Un robot mobile est dit omnidirectionnel si l'on peut agir indépendamment sur les vitesses de translation selon les axes $\vec{x}\vec{y}$ la vitesse de rotation autour de z

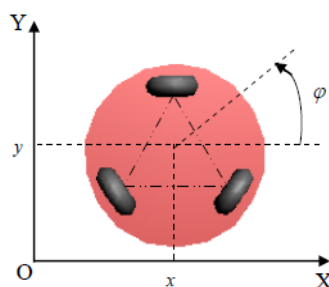


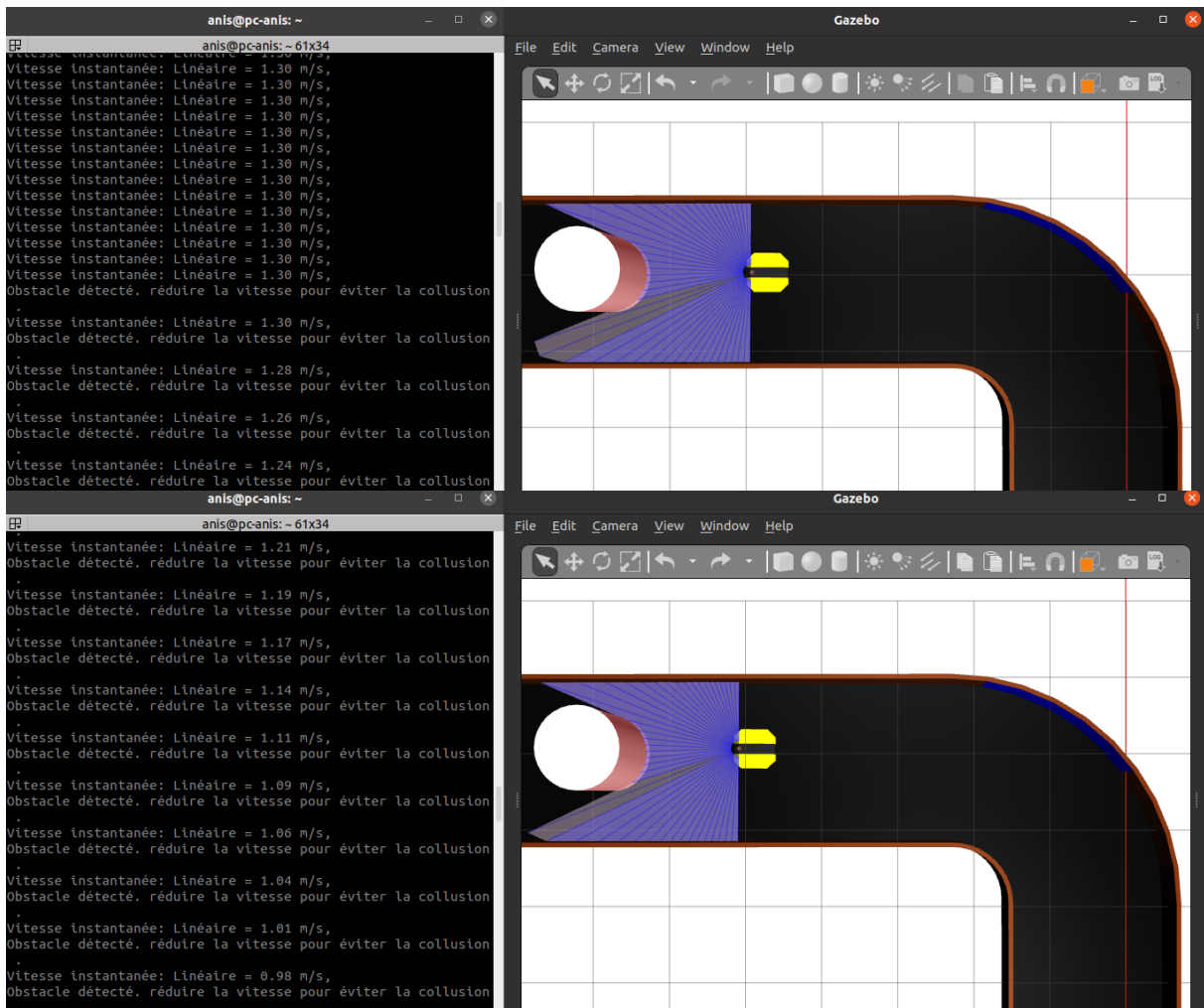
Figure III-17 Robot mobile omnidirectionnel

III. 6. 1. 5 Détection et évitement des obstacles

Pour que le robot puisse éviter un obstacle, il doit le détecter préalablement en calculant la distance entre sa position actuelle et l'obstacle, afin d'assurer la sécurité et d'éviter toute collision. Ensuite, il ajuste sa vitesse en fonction de la distance de l'obstacle, ce qui lui permet de l'éviter en réduisant sa vitesse.

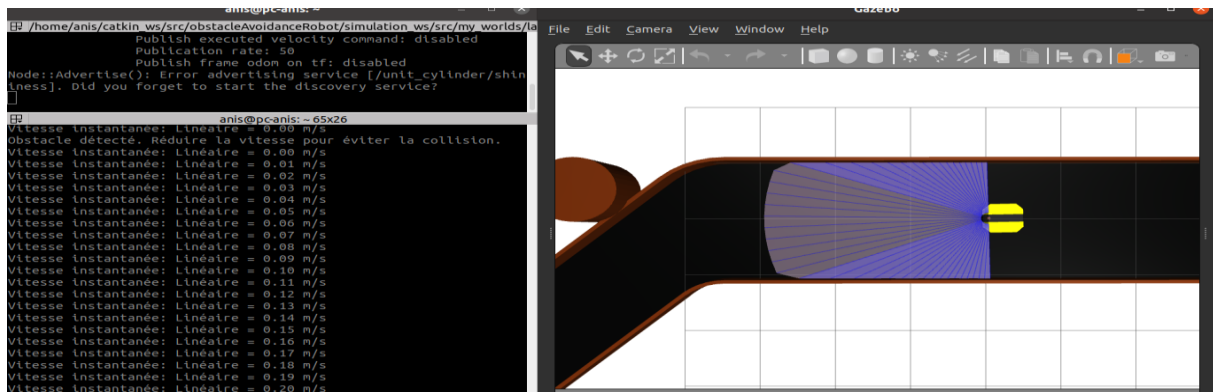
III.6.1.5.1 Navigation avec obstacle :

Pendant son déplacement, le robot détecte la présence d'un obstacle grâce à un capteur laser. Lorsque le robot rencontre un obstacle devant lui, sa vitesse diminue progressivement jusqu'à éviter l'obstacle. Cette diminution de vitesse est affichée dans le terminal sous forme d'une liste, et est également illustrée par une figure explicative.





Lorsque les obstacles sont dégagés sur la trajectoire, le robot augmente sa vitesse et continue à se déplacer.



FigureIII-20 Augmentation de la vitesse de robot et continue à se déplacer.

III. 7 Conclusion

Grâce à ces outils de simulation basés sur ROS (Robot Operating System) et Gazebo, nous avons pu simuler, évaluer et tester notre système d'aide à la conduite dans des scénarios réalistes avant son déploiement sur le terrain. La simulation nous a offert une plateforme sûre et efficace pour tester et améliorer notre système, réduisant ainsi les coûts et les risques liés à des tests en conditions réelles

Conclusion générale

Au terme de cette étude approfondie sur l'aide à la conduite pour un système de transport, nous avons pu explorer différents aspects clés liés à la sécurité routière et à l'efficacité des systèmes anti-collision sur véhicules, tramway, robots, etc. Notre objectif principal était de concevoir un système d'aide à la conduite capable de détecter et de prévenir les collisions, contribuant ainsi à une expérience de conduite plus sécurisée et à une réduction des risques d'accidents pour les passagers, les conducteurs et les piétons.

Dans notre premier chapitre, nous avons examiné les principes de fonctionnement d'un système anti-collision, en mettant l'accent sur les technologies et les composants essentiels. Cette compréhension approfondie nous a permis d'élaborer une base solide pour la conception de notre propre système d'aide à la conduite.

Le deuxième chapitre a porté sur les approches de navigation et les méthodes d'échantillonnage en particulier. Nous avons étudié différentes techniques de navigation dans des environnements dynamiques. En choisissant la meilleure méthode à savoir, la méthode d'échantillonnage ISS adaptées à notre système de transport.

Le cœur de notre étude résidait dans ce chapitre, où nous avons abordé la conception d'un système d'aide à la conduite basée sur une variante de la méthode ISS, appelée PassISS permettant de garantir la sûreté de mouvement. À travers la modélisation des capteurs, l'intégration des données, les algorithmes de détection et de prévention des collisions, nous avons créé une architecture solide pour notre système. Les considérations de validation et d'optimisation nous ont permis de peaufiner notre conception et de garantir son efficacité et sa fiabilité.

Enfin, le troisième chapitre s'est concentré sur l'utilisation des outils de simulation basés sur ROS (Robot Operating System) et Gazebo. Grâce à ces outils, nous avons pu simuler, évaluer et tester notre système d'aide à la conduite dans des scénarios réalistes avant son déploiement sur le terrain. La simulation nous a offert une plateforme sûre et efficace pour tester et améliorer notre système, réduisant ainsi les coûts et les risques liés à des tests en conditions réelles.

Ce travail peut être étendu en impliquant la communication entre plusieurs systèmes de transport et la prise en considération de l'échange de données.

Bibliographie

- [1]Fort, Alexandra, Mercedes Bueno, and Christophe Jallais. Systèmes d'alerte anti-collision: état de l'art et impact du niveau de fiabilité et du moment de déclenchement. Diss. IFSTTAR-Institut Français des Sciences et Technologies des Transports, de l'Aménagement et des Réseaux, 2014.
- [2]Liu, Tsung-Hsin, Miao-Lin Hsu, and Zou-Min Tsai. "Mutual interference of pseudorandom noise radar in automotive collision avoidance application at 24 GHz." 2016 IEEE 5th Global Conference on Consumer Electronics. IEEE, 2016.
- [3]Kotur, Mila, et al. "Camera and LiDAR Sensor Fusion for 3D Object Tracking in a Collision Avoidance System." 2021 Zooming Innovation in Consumer Technologies Conference (ZINC). IEEE, 2021.
- [4]Imam, Mohamed, et al. "The future of mine safety: a comprehensive review of anti-collision systems based on computer vision in underground mines." *Sensors* 23.9 (2023): 4294.
- [5]Liu, Yucheng, et al. An automatic emergency braking system for collision avoidance assist of multi-trailer vehicle based on model prediction control. No. 2021-01-0117. SAE Technical Paper, 2021
- [6]Cicchino, Jessica B. "Effects of automatic emergency braking systems on pedestrian crash risk." *Accident Analysis & Prevention* 172 (2022): 106686.
- [7]Pananurak, Worrawut, Somphong Thanok, and Manukid Parnichkun. "Adaptive cruise control for an intelligent vehicle." 2008 IEEE International Conference on Robotics and Biomimetics. IEEE, 2009.
- [8]Gamal, Islam, et al. "A robust, real-time and calibration-free lane departure warning system." *Microprocessors and Microsystems* 71 (2019): 102874.
- [9]Yue, Lishengsa, et al. "Effects of forward collision warning technology in different pre-crash scenarios." *Transportation research part F: traffic psychology and behaviour* 76 (2021): 336-352.
- [10]Cicchino, Jessica B. "Effects of blind spot monitoring systems on police-reported lane-change crashes." *Traffic injury prevention* 19.6 (2018): 615-622.
- [11]Williamson, Thomas, and Ned A. Spencer. "Development and operation of the traffic alert and collision avoidance system (TCAS)." *Proceedings of the IEEE* 77.11 (1989): 1735-1744.
- [12]Breen, Barry C. "Controlled flight into terrain and the enhanced ground proximity warning system." *IEEE aerospace and electronic systems magazine* 14.1 (1999): 19-24.
- [13]Zhu, Li, et al. "Cross-layer handoff design in MIMO-enabled WLANs for communication-based train control (CBTC) systems." *IEEE Journal on Selected Areas in Communications* 30.4 (2012): 719-728.

Bibliographie

- [14] Aguiar, Braulio, et al. "A robust and fault tolerant approach for automatic train stop control system design." *IFAC-PapersOnLine* 50.1 (2017): 8549-8554.
- [15] Maile, Michael, et al. "Intersection collision avoidance: From driver alerts to vehicle control." 2015 IEEE 81st Vehicular Technology Conference (VTC Spring). IEEE, 2015.
- [16] Harati-Mokhtari, Abbas, et al. "Automatic Identification System (AIS): data reliability and human error implications." *The Journal of Navigation* 60.3 (2007): 373-389.
- [17] Zhao, Liangbin, and Xiuju Fu. "A method for correcting the closest point of approach index during vessel encounters based on dimension data from AIS." *IEEE Transactions on Intelligent Transportation Systems* 23.8 (2021): 13745-13757.
- [18] Cai, Chengtao, et al. "Collision avoidance in multi-robot systems." 2007 International Conference on Mechatronics and Automation. IEEE, 2007.
- [19] Di Palma, Catello, et al. "Driver assistance system for trams: Smart tram in smart cities." 2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe). IEEE, 2020.
- [20] Simon, Claude, and Patrick Longuet. *Le tramway*. Minuit, 2001.
- [21] M. Mizumachi, A. Kaminuma, N. Ono, and S. Ando, "Robust sensing of approaching vehicles relying on acoustic cues," *Sensors*, vol. 14, no. 6, pp. 9546–9561, May 2014.
- [22] C. Blanc, R. Aufrere, L. Malaterre, J. Gallice, and J. Alizon, "Obstacle detection and tracking by millimeter wave radar," in *Proc. IEEE Symp. IAV, Lisboa, Portugal, 2004*, pp. 1–6. [Online]. Available: [http:// christophe-blanc.info/fics/iav2004/iav04blanc.pdf](http://christophe-blanc.info/fics/iav2004/iav04blanc.pdf)
- [23] H.-J. Cho and M.-T. Tseng, "A support vector machine approach to CMOS-based radar signal processing for vehicle classification and speed estimation," *Math. Comput. Modelling*, vol. 58, no. 1/2, pp. 438–448, Jul. 2012.
- [24] S. Xuan, Z. Zheng, Z. Chenglin, and Z. Weixia, "A compressed sensing radar detection scheme for closing vehicle detection," in *Proc. IEEE ICC, 2012*, pp. 6371–6375.
- [25] S. J. Park, T. Y. Kim, S. M. Kang, and K. H. Koo, "A novel signal processing technique for vehicle detection radar," in *Proc. IEEE MTT-S Int. Microw. Symp. Dig., 2003*, pp. 607–610, vol. 1
- [26] F. Nashashibi and A. Bargeton, "Laser-based vehicles tracking and classification using occlusion reasoning and confidence estimation," in *Proc. IEEE Intell. Veh. Symp., 2008*, pp. 847–852.
- [27] D. J. Natale, R. L. Tutwiler, M. S. Baran, and J. R. Durkin, "Using full motion 3D Flash LIDAR video for target detection, segmentation, and tracking," in *Proc. IEEE SSI AI, 2010*, pp. 21–24.
- [28] Mégie, Gérard. "Applications des lasers à l'étude de l'atmosphère." *Revue de Physique Appliquée* 14.2 (1979): 369-378.

Bibliographie

- [29]M. Bertozzi et al., “Artificial vision in road vehicles,” Proc. IEEE, vol. 90, no. 7, pp. 1258–1271, Jul. 2002.
- [30]K. Jisu, H. Sungjun, B. Jeonghyun, K. Euntai, and L. HeeJin, “Autonomous vehicle detection system using visible and infrared camera,” in Proc. 12th ICCAS, 2012, pp. 630634.
- [31]Bouraine, S. Contribution à la localisation dynamique du robot mobile d’intérieur b21r en utilisant la plateforme multi sensorielle. Diss. Master’s thesis, Université de Saad Dahleb de Blida, Faculté des sciences d’ingénieur, 2007.
- [32]Delsart, Vivien. Navigation autonome en environnement dynamique: Une approche par déformation de trajectoire. Diss. Université de Grenoble, 2010.
- [33]Ribeiro, Maria Isabel. "Obstacle avoidance." Instituto de Sistemas e Robótica, Instituto Superior Técnico 1 (2005).
- [34] Van den Berg, Jur, Ming Lin, and Dinesh Manocha. "Reciprocal velocity obstacles for real-time multi-agent navigation." 2008 IEEE international conference on robotics and automation. Ieee, 2008.
- [35] Fraichard Th. and Asama H., —Inevitable collision states. a step towards safer robots?!, Advanced Robotics, V. 18, Issue 10, (2004), pages 1001-1024.
- [36]Todoran, Horatiu George. Optimal Local Path-Planning and Control for Mobile Robotics. Diss. TU Wien, 2018.
- [37]Fraichard Th., —A short paper about motion safety!, IEEE Int. Conf. on Robotics and Automation, (2007),
- [38]Bouraine, S., Fraichard, T., &Salhi, H. (2012). Provablysafe navigation for mobile robots withlimitedfield-of-views in dynamicenvironments. *Autonomous Robots*, 32, 267-283.
- [39]Fairchild, Carol, and Thomas L. Harman. ROS Robotics By Example: Learning to control wheeled, limbed, and flying robots using ROS Kinetic Kame. Packt Publishing Ltd, 2017.
- [40]Parthasarathi R. and Fraichard T., —An inevitable collision state checker for a car-like vehicle!, in IEEE International Conference on Robotics and Automation, (2007), pages 3068-3073.