



Stratégies avancées de planification et d'optimisation des déplacements du bras robotique dans le
contexte évolutif de l'Industrie 5.0

par Salick Diagne

Mémoire présenté à l'Université du Québec à Chicoutimi en vue de l'obtention du grade de Maîtrise
ès sciences appliquées (M. Sc. A.) en ingénierie

Jury :

Martin Otis, ing. Ph.D., Université du Québec à Chicoutimi, Directeur de recherche

Yacine Yaddaden, Ph.D., Professeur, Université du Québec à Rimouski, Membre externe

Remy Meziane, chercheur CNRC, professeur UQAC, Membre interne

Québec, Canada

© Salick Diagne, 2025

RÉSUMÉ

Ce mémoire s'inscrit dans la dynamique de l'industrie 5.0, où l'interaction avancée entre humains et robots devient un levier stratégique pour renforcer la flexibilité, la sécurité et la performance des systèmes de production. L'étude se concentre sur l'optimisation des trajectoires de bras robotiques opérant à proximité d'opérateurs humains dans le cadre spécifique du désassemblage de batteries de véhicules électriques, une tâche à la fois complexe, sensible et à haut risque.

L'objectif principal est de garantir un haut niveau de sécurité pour les opérateurs tout en assurant des trajectoires précises, efficaces et réactives pour le bras robotique. Pour ce faire, une architecture logicielle complète a été développée intégrant plusieurs composantes complémentaires : une cartographie 3D dynamique de l'environnement, construite à l'aide de la bibliothèque OctoMap ; un module *IETA* simplifié (modélisation *Time To Collision* dans ROS2) pour l'évaluation proactive des risques de collision ; une planification globale hybride basée sur les algorithmes Thêta* et D* Lite ; et un planificateur local *TEB* (*Timed Elastic Band*) pour l'adaptation fine des trajectoires en présence d'obstacles dynamiques. Les résultats expérimentaux obtenus en simulation montrent que cette architecture permet de concilier robustesse, adaptabilité et efficacité. La combinaison Thêta* et D* Lite assure une planification fluide et réactive, tandis que TEB améliore localement la fluidité du mouvement sans compromettre la sécurité. L'ensemble du système est déployé dans l'environnement ROS2/MoveIt2 avec un bras robotique UR5 simulé.

Au-delà des performances techniques, cette démarche ouvre des perspectives concrètes pour une robotique collaborative plus sûre et plus intelligente, capable de s'adapter à la complexité croissante des environnements industriels. L'approche développée peut être généralisée à d'autres scénarios de manipulation robotique dans le cadre de l'économie circulaire et des systèmes de production durables.

ABSTRACT

This thesis is part of the industry 5.0 movement, where advanced human-robot interaction emerges as a strategic lever to enhance flexibility, safety, and performance in production systems. The study focuses on optimizing the trajectories of robotic arms operating near human operators, within the specific context of electric vehicle battery disassembly, a complex, sensitive, and high-risk task.

The main objective is to ensure a high level of operator safety while enabling the robotic arm to execute precise, efficient, and reactive trajectories. To achieve this, a comprehensive software architecture has been developed, integrating several complementary components: a dynamic 3D mapping of the environment built using the OctoMap library; a simplified IETA module (Time To Collision modeling within ROS2) for proactive collision risk assessment; a hybrid global planner combining Thêta* and D* Lite algorithms; and a local planner based on the Timed Elastic Band (TEB) algorithm for fine trajectory adaptation in the presence of dynamic obstacles.

Experimental results obtained in simulation show that this architecture successfully balances robustness, adaptability and efficiency. The combination of Thêta* and D* Lite ensures smooth and reactive global planning, while TEB improves local trajectory smoothness without compromising safety. The entire system is implemented within the ROS2/MoveIt2 ecosystem using a simulated UR5 robotic arm.

Beyond technical performance, this work opens concrete perspectives for safer and smarter collaborative robotics, capable of adapting to the increasing complexity of industrial environments. This suggested methodology has the potential to be broadened to encompass additional robotic manipulation contexts within the framework of the circular economy and sustainable manufacturing systems.

TABLE DES MATIÈRES

RÉSUMÉ	ii
ABSTRACT	iii
TABLE DES MATIÈRES	iv
LISTE DES TABLEAUX	vi
LISTE DES FIGURES	vii
LISTE DES ABRÉVIATIONS	viii
REMERCIEMENTS	x
CHAPITRE 1	1
INTRODUCTION GÉNÉRALE	1
1.1. CONTEXTE	2
1.2. PROBLÉMATIQUE	3
1.4. MÉTHODOLOGIE GÉNÉRALE	5
1.5. ORGANISATION DES CHAPITRES	6
CHAPITRE 2	9
REVUE DE LA LITTÉRATURE	9
1. INTRODUCTION	9
2. METHODOLOGIE DE LA REVUE DE LITTÉRATURE	10
2.2.1. ANALYSE DES DOCUMENTS SELECTIONNES	13
2.2.1.1. ALGORITHMES TRADITIONNELS ET LEURS LIMITES	13
2.2.1.2. L'ÉMERGENCE DES APPROCHES HYBRIDES	14
2.2.1.3. INNOVATIONS ALGORITHMIQUES RÉCENTES	15
2.2.1.4. ÉVITEMENT D'OBSTACLES DYNAMIQUES	16
2.2.1.5. OPTIMISATION ÉNERGÉTIQUE ET DISTANCE PARCOURUE POUR UNE AUTONOMIE MAXIMALE	17
2.2.1.6. ROBOTIQUE COLLABORATIVE, DÉFIS ET SOLUTIONS POUR LES BRAS INDUSTRIELS	18
2.2.1.7. ANALYSE CRITIQUE ET PERSPECTIVES	19
3. CONCLUSION	21
CHAPITRE 3	23
MÉTHODOLOGIE	23
3.1. INTRODUCTION	23
3.2. HYPOTHÈSES DE RECHERCHE	24
3.3. MÉTHODOLOGIE	24
3.3.1. CARTOGRAPHIE 3D DE L'ENVIRONNEMENT	26
3.3.1.1. ACQUISITION DES DONNEES PAR LA CAMERA INTEL REALSENSE D455	26
3.3.1.1.1. PRINCIPE DE LA MESURE DE PROFONDEUR	27
3.3.1.1.2. GESTION DE LA BANDE DE PROFONDEUR INVALIDE	28
3.3.2. CONSTRUCTION DE LA CARTE 3D AVEC OCTOMAP	30
3.3.3. INTÉGRATION DANS ROS2 ET VISUALISATION	32
3.3.4. CRÉATION DU COSTMAP STATIQUE ET DYNAMIQUE	33
3.3.4.1. FONCTIONNEMENT ET ARCHITECTURE DU NŒUD DYNAMICCOSTMAPNODE	35
3.3.4.2. INTÉGRATION DU MODULE IETA SIMPLIFIÉ (OU TTC)	36
3.3.5. PLANIFICATEUR GLOBAL ET LOCAL	38

3.3.5.1.	PLANIFICATEUR GLOBAL	38
3.3.5.1.1.	ALGORITHME THÊTA*	38
3.3.5.1.2.	ALGORITHME D* LITE	39
3.3.5.2.	PLANIFICATEUR LOCAL.....	40
3.4.	CONCLUSION.....	41
CHAPITRE 4		42
RÉSULTATS ET DUSCUSSIONS		42
4.1.	INTRODUCTION	42
4.2.	PERFORMANCES DES PLANIFICATEURS.....	43
4.2.1.	POSE SANS PRÉSENCE D'OBSTACLES	46
4.2.2.	POSE ELOIGNEE DES OBSTACLES	47
4.2.3.	POSE PROCHE D'UN OBSTACLE	49
4.2.4.	POSE TRÈS PROCHE DE L'OBSTACLE	51
4.2.5.	POSE PRESQUE DERRIÈRE UN OBSTACLE	53
4.3.	INFLUENCE DU TEB SUR L'ALGORITHME D* LITE RAFFINÉ	56
4.4.	CONCLUSION.....	57
CHAPITRE 5		59
CONCLUSION		59
LISTE DE RÉFÉRENCES.....		61

LISTE DES TABLEAUX

TABLEAU 1 : REQUETE DE MOTS CLES UTILISES	12
TABLEAU 2 : COMPARAISON DE QUELQUES TRAVAUX DE RECHERCHE AVEC NOTRE APPROCHE.....	22

LISTE DES FIGURES

FIGURE 1 : METHODOLOGIE GENERALE	6
FIGURE 2 : ORGANISATION DES CHAPITRES DU MEMOIRE	8
FIGURE 3 : TAXINOMIE DES ALGORITHMES DE PLANIFICATION DE CHEMIN	10
FIGURE 4 : REVUE DE LA LITTÉRATURE : A) NOMBRE D'ARTICLES PUBLIÉS ; B) MÉTHODOLOGIE PRISMA	12
FIGURE 5 : DIAGRAMME DESCRIPTIF DE LA METHODE	26
FIGURE 6 : EXEMPLE D'OCTREE STOCKANT DES CELLULES LIBRES (BLANC) ET OCCUPEES (BLEU) DANS OCTOMAP (MODIFIE)[62].....	32
FIGURE 7 : CREATION DU COSTMAP DYNAMIQUE	34
FIGURE 8 : ILLUSTRATION DU COSTMAP 3D DYNAMIQUE	43
FIGURE 9 : EXEMPLE DE TRAJECTOIRE POUR LES TROIS ALGORITHMES.....	45
FIGURE 10 : BANC D'ESSAI UTILISE POUR LES TESTS	45
FIGURE 11 : PERFORMANCES DES ALGORITHMES POUR UNE POSE SANS OBSTACLES..	47
FIGURE 12 : PERFORMANCES DES ALGORITHMES POUR UNE POSE ELOIGNEE DES OBSTACLES	49
FIGURE 13 : PERFORMANCES DES ALGORITHMES POUR UNE POSE PROCHE D'UN OBSTACLE.....	51
FIGURE 14 : PERFORMANCES DES ALGORITHMES POUR UNE POSE TRES PROCHE D'UN OBSTACLE.....	53
FIGURE 15 : PERFORMANCES DES ALGORITHMES POUR UNE POSE PRESQUE DERRIERE UN OBSTACLE	55
FIGURE 16 : INFLUENCE DU TEB SUR L'ALGORITHME D*LITE RAFFINE	57

LISTE DES ABRÉVIATIONS

Abréviations	Français	Anglais
ACO	Optimisation par Colonie de Fourmis	Ant Colony Optimization
AGA	Algorithme Génétique Adaptatif	Adaptive Genetic Algorithm
APF	Champ de Potentiel Artificiel	Artificial Potential Field
ARA	Réparation à tout moment A	Anytime Repairing A
BP	Rétropropagation	BackPropagation
CNN	Réseaux de neurones convolutifs	Convolutional Neural Network
DQN	Réseau Q profond	Deep Q-Network
DRL	Apprentissage par renforcement profond	Deep Reinforcement Learning
DVP	Perception Visuelle Profonde	Deep Visual Perception
DWA	Approche de la fenêtre dynamique	Dynamic Window Approach
ERG	Représentation étendue des graphes	Extended Representation of Graphs
GA	Algorithme génétique	Genetic Algorithm
IA	Intelligence Artificielle	Artificial Intelligence
IABC	Colonie d'abeilles artificielles améliorée	Improved Artificial Bee Colony
IDWA	Approche de fenêtre dynamique améliorée	Improved Dynamic Window Approach
IHR	Interaction Humain-Robot	Human-Robot Interaction
IRH	Interaction Robot-Humain	Robot-Human Interaction
IVPF	Champ de potentiel de vitesse amélioré	Improved Velocity Potential Field
PCC	Plus Court Chemin	Shortest Path
PPO	Optimisation de politique proximale	Proximal Policy Optimization
PRISMA	Critères d'évaluation pour les revues systématiques et les méta-analyses	Preferred Reporting Items for Systematic Reviews and Meta-Analyses

PRM	Carte probabiliste	Probabilistic Roadmap
PSO	Optimisation par essaim de particules	Particle Swarm Optimization
RBF	Fonction de base radiale	Radial Basis Function
RNN	Réseaux de Neurones Récurrents	Recurrent Neural Networks
ROS	Système d'exploitation pour robots	Robot Operating System
RRT	Arbre de recherche rapide aléatoire	Rapidly-exploring Random Tree
SBPL	Bibliothèque de planification basée sur la recherche	Search-Based Planning Library
TEB	Bande élastique temporelle	Timed Elastic Band
UAV	Véhicule aérien sans pilote	Unmanned Aerial Vehicle
USV	Véhicule de surface sans pilote	Unmanned Surface Vehicle
VIS-LT	Arbre local de visibilité	Visibility Local Tree
VPF	Carte probabiliste	Virtual Potential Field

REMERCIEMENTS

Je tiens tout d'abord à exprimer ma profonde gratitude au Professeur Martin Otis, directeur de ce mémoire, pour sa disponibilité, son encadrement bienveillant et la qualité de ses conseils tout au long de ce travail. Son accompagnement a été déterminant dans l'avancement de ce projet.

Je remercie chaleureusement l'ensemble des membres du Laboratoire d'Automatique et de Robotique interactive (LAR.i) pour leurs échanges stimulants, leur soutien technique et leurs retours constructifs, qui ont grandement enrichi cette recherche.

Mes remerciements s'adressent également à toute l'administration de l'université ainsi qu'à l'ensemble des enseignants, pour leur dévouement et la qualité des formations dispensées tout au long de mon parcours.

Enfin, je tiens à remercier mes amis et camarades, pour leur soutien moral, leur présence constante, et les nombreux moments de partage qui ont rendu cette expérience universitaire inoubliable.

CHAPITRE 1

INTRODUCTION GÉNÉRALE

Ces dernières années, l'industrie a connu une profonde transformation sous l'impulsion des technologies émergentes. Après l'ère de l'industrie 4.0, marquée par la connectivité des systèmes et l'automatisation intelligente, l'industrie 5.0 ouvre une nouvelle voie, celle de la collaboration étroite entre les humains et les robots. Dans cette approche où l'humain est au centre du processus industriel, les robots ne remplacent pas les opérateurs, mais les assistent, en partageant leur espace de travail de manière fluide et sécurisée [1, 2].

Dans ce contexte, la robotique collaborative prend une importance croissante dans de nombreux secteurs industriels. Les bras robotiques et en particulier les cobots offrent une flexibilité et une précision qui en font des partenaires idéaux pour les tâches complexes, dangereuses ou répétitives. Cependant, pour garantir une collaboration efficace, ces robots doivent être capables de percevoir leur environnement, d'anticiper les mouvements humains, et de s'adapter en temps réel.

Le cas du désassemblage des batteries de véhicules électriques illustre parfaitement ces enjeux. Le désassemblage manuel de ces batteries en fin de vie reste une tâche délicate et risquée. L'automatisation partielle de ce processus, en combinant l'intelligence des opérateurs avec la robustesse des cobots, représente une solution prometteuse [3]. Néanmoins, cette collaboration impose de nouveaux défis, notamment en matière de planification dynamique de trajectoires, dans un environnement partagé et évolutif.

Ce mémoire s'inscrit dans cette problématique. Il propose le développement d'une approche intégrée visant à optimiser les trajectoires des bras robotiques en interaction étroite avec les opérateurs humains. L'objectif est de concevoir un système réactif, capable d'assurer à la fois la sécurité des travailleurs et l'efficacité du processus de désassemblage, en s'appuyant sur les technologies avancées de perception 3D, de planification adaptative et de contrôle local.

Ce chapitre introductif présente le contexte de l'étude, la problématique de recherche, les objectifs poursuivis, ainsi que la méthodologie générale mise en œuvre pour y répondre. L'organisation détaillée du mémoire est exposée à la fin de ce chapitre.

1.1. CONTEXTE

La transition vers l'industrie 5.0 introduit une nouvelle ère de collaboration entre les humains et les robots, avec pour ambition de rendre les processus industriels plus flexibles, plus efficaces, et surtout plus sûrs. Dans cette vision, l'humain reste au cœur des lignes de production, tandis que les systèmes robotiques deviennent des partenaires intelligents, capables de s'adapter dynamiquement à l'environnement de travail.

L'intégration des bras robotiques collaboratifs dans les ateliers modernes est aujourd'hui en pleine expansion [4, 5] . Ces dispositifs sont conçus pour effectuer des tâches répétitives, précises et potentiellement dangereuses, tout en partageant l'espace de travail avec des opérateurs humains. Leur utilisation permet non seulement d'augmenter la productivité, mais aussi de réduire les risques liés à la pénibilité et à la sécurité au travail.

Le secteur du désassemblage des batteries de véhicules électriques illustre parfaitement ce besoin de collaboration homme-robot. Avec l'essor de la mobilité électrique, le nombre de batteries en fin de vie augmente considérablement. Leur désassemblage, préalable essentiel au recyclage [6-8], demeure un processus complexe et délicat. Certaines tâches telles que le découpe de câbles, déconnexion de *Batterie Management System (BMS)* nécessitent encore l'intervention humaine pour garantir la flexibilité et la précision requises. Or, la manipulation de batteries accidentées ou usées expose les opérateurs à des risques électriques, chimiques et mécaniques.

L'automatisation partielle de ce processus à l'aide de cobots constitue une réponse pertinente à ces défis. Toutefois, cette collaboration impose que les bras robotiques soient capables de planifier et d'ajuster leurs trajectoires, en tenant compte des mouvements des opérateurs humains. Garantir une interaction fluide et sécurisée devient ainsi un enjeu central.

Dans ce contexte, la planification avancée des trajectoires robotiques prend toute son importance. Elle vise à optimiser les déplacements du robot pour assurer l'efficacité du processus, tout en minimisant les risques de collision et en respectant les contraintes spécifiques du poste de travail.

1.2. PROBLÉMATIQUE

Si l'intégration des robots dans le processus de désassemblage des batteries offre de nombreux avantages, elle soulève néanmoins des défis importants, en particulier en ce qui concerne la planification des trajectoires en environnement dynamique et partagé.

Le premier défi réside dans la sécurité de l'opérateur humain. Le robot doit pouvoir anticiper et éviter les collisions potentielles avec les travailleurs, dont les mouvements sont souvent imprévisibles. Cet objectif nécessite la mise en œuvre d'algorithmes capables de détecter les obstacles dynamiques et de replanifier les trajectoires en temps réel.

Le second défi concerne l'optimisation des trajectoires. Celles-ci doivent non seulement garantir la sécurité, mais aussi favoriser un déroulement fluide et rapide des opérations. Une trajectoire efficace permet de réduire les temps de cycle, d'économiser de l'énergie et de limiter l'usure mécanique du robot.

À ces contraintes s'ajoutent des spécificités propres à l'environnement de travail notamment la :

- Variabilité des positions et des configurations des batteries à désassembler ;
- Nécessité d'adapter dynamiquement la vitesse maximale et la force de l'effecteur en fonction de la distance aux opérateurs et de son état mental, bien que cette vitesse maximale en proximité d'humain soit définie dans la norme CSA/ISO Z434 ;
- Prise en compte des caractéristiques mécaniques et électriques des batteries (taille, fragilité, modèle), qui imposent une manipulation soignée.

Enfin, le désassemblage s'effectue en plusieurs étapes successives, chacune nécessitant des trajectoires adaptées. La planification doit donc être capable de gérer des scénarios complexes,

évolutifs, et partiellement inconnus. Face à ces exigences, il devient essentiel de concevoir un système de planification de trajectoires capable de répondre simultanément aux objectifs de sécurité, efficacité et flexibilité.

1.3. OBJECTIF

Ce mémoire vise à développer une solution complète pour la planification de trajectoires sécurisées et optimisées d'un bras robotique UR5, dans le cadre du désassemblage de batteries électriques en environnement industriel collaboratif.

Le premier objectif consiste à concevoir une cartographie de l'environnement combinant une carte de coûts statique (obstacles fixes) et une carte de conflit 3D dynamique (obstacles mobiles), enrichie par un module *IETA* (*Interference Estimated Time to Arrival*). Cette cartographie permettra au robot d'anticiper les situations de risque et de réagir de manière appropriée.

Le second objectif est d'évaluer la pertinence de la *combinaison des algorithmes Thêta* et D* Lite* pour la planification globale des trajectoires en environnement dynamique. L'évaluation portera notamment sur la capacité de ces algorithmes à produire des trajectoires optimales en termes de sécurité, de rapidité et de réactivité.

Le troisième objectif est d'évaluer l'impact du planificateur local (TEB) sur la combinaison des algorithmes Thêta* et D* Lite, en particulier sur la qualité, la fluidité et la sécurité des trajectoires générées. Ce module jouera un rôle essentiel pour renforcer la sécurité de l'interaction homme-robot en optimisant localement les trajectoires en environnement dynamique.

Ces objectifs seront validés expérimentalement à travers une série de scénarios représentatifs du contexte industriel visé.

1.4. MÉTHODOLOGIE GÉNÉRALE

La méthodologie générale adoptée dans le cadre de ce mémoire est présentée à la Figure 1. Elle a été structurée de manière à valider les hypothèses formulées et à évaluer de façon rigoureuse les performances du système proposé, en tenant compte de la complexité des environnements industriels collaboratifs. Après le chapitre introductif et la revue de la littérature, deux hypothèses sont émises. La première hypothèse permet de vérifier si la création d'une cartographie hybride combinant une carte de coûts 3D statique et une carte de conflit 3D dynamique enrichie par le module *IETA (TTC)* améliorent significativement la capacité d'évitement des collisions du bras robotique. Pour ce faire, une étape initiale consiste à générer cette cartographie hybride à partir des données issues de la caméra Intel RealSense D455 et traitées par la bibliothèque OctoMap. La carte dynamique est ensuite enrichie en temps réel par le module *IETA*, qui anticipe les risques de collision en fonction de la position et de la vitesse des obstacles mobiles.

Une fois cette cartographie en place, elle sert d'entrée pour la planification globale des trajectoires. C'est dans ce cadre que s'inscrit la seconde hypothèse où nous supposons que l'utilisation combinée des algorithmes *Thêta** et *D* Lite* permet d'améliorer la réactivité et la précision des trajectoires générées par rapport à une utilisation individuelle de ces algorithmes. Pour valider cette hypothèse, les deux algorithmes sont d'abord testés séparément, puis combinés au sein d'une architecture hybride. La performance de cette combinaison est évaluée selon plusieurs critères. Une fois le plan de trajectoire global généré, il est affiné par un planificateur local *TEB (Timed Elastic Band)*. Ce contrôleur ajuste en temps réel les trajectoires en fonction des variations de l'environnement et des mouvements des opérateurs. L'efficacité de ce module sera également analysée en termes de fluidité du mouvement, de maintien des marges de sécurité, et de robustesse face aux perturbations.

Enfin, l'ensemble du système est validé à travers une série de scénarios expérimentaux représentatifs, simulés dans RViz. Ces scénarios permettent d'évaluer l'apport de chaque composant de la méthodologie proposée et de valider les hypothèses initiales.

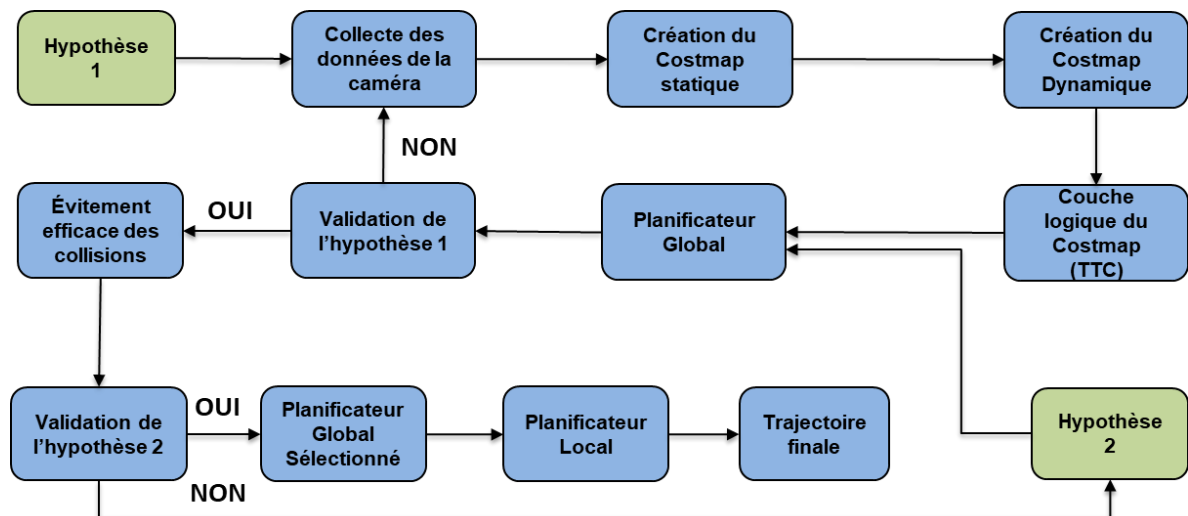


Figure 1 : Méthodologie générale

1.5. ORGANISATION DES CHAPITRES

Ce mémoire, tel qu'illustré à la figure 2, est structuré en cinq chapitres :

- Tout d'abord, le présent chapitre (chapitre 1) introduit le projet en présentant le contexte général de l'étude, la problématique de recherche, les objectifs poursuivis ainsi que la méthodologie générale adoptée ;
- Le chapitre 2 propose une revue détaillée de la littérature, couvrant les approches existantes en matière de robotique collaborative, de planification de trajectoires en environnement dynamique et de perception avancée. Cette revue permet de positionner les contributions du projet par rapport à l'état de l'art ;
- Le chapitre 3 décrit en détail la méthodologie mise en œuvre pour atteindre les objectifs du projet. Il expose l'architecture matérielle et logicielle retenue, les algorithmes développés, ainsi que le protocole expérimental permettant de valider les hypothèses de recherche ;
- Le chapitre 4 présente et analyse les résultats expérimentaux obtenus. Il propose une discussion critique sur les performances du système, l'efficacité des différentes approches de planification testées et la validation des hypothèses ;

- Enfin, le chapitre 5 conclut le mémoire en synthétisant les contributions majeures apportées par ce travail de recherche et en proposant des pistes d'amélioration et des perspectives pour de futurs travaux.

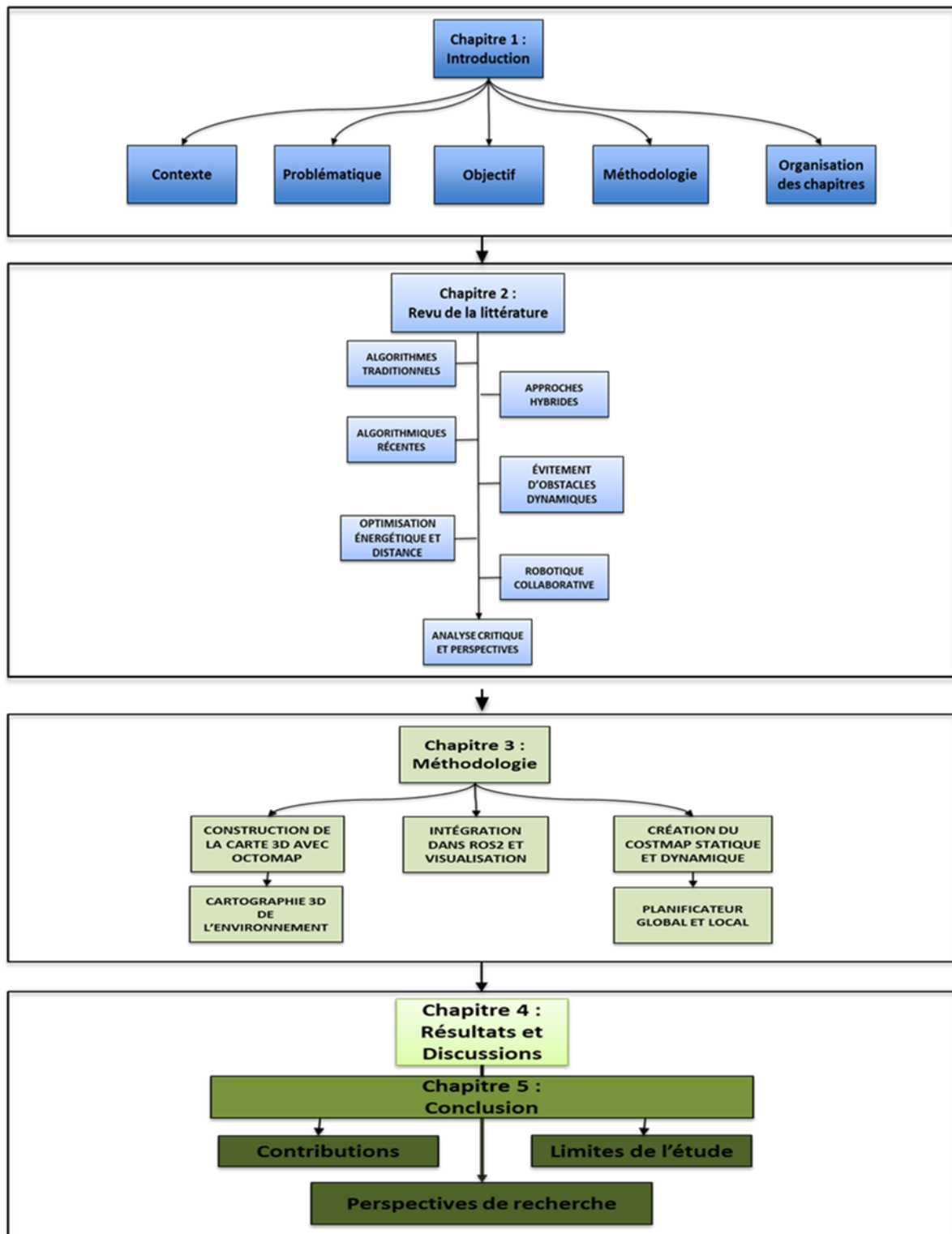


Figure 2 : Organisation des chapitres du mémoire

CHAPITRE 2 REVUE DE LA LITTÉRATURE

1. INTRODUCTION

Le chapitre précédent a permis d'exposer les besoins du projet et de définir son périmètre. La présente revue de littérature vise à explorer les solutions existantes permettant de répondre à la problématique posée, en identifiant les approches les plus pertinentes pour l'optimisation des trajectoires de bras robotiques en environnement collaboratif. Les travaux de recherche dans ce domaine ont été menés dans divers secteurs tels que la planification de trajectoires pour les véhicules aériens sans pilote (*UAV*), les navires de surface sans pilote (*USV*), mais surtout pour les bras robotiques opérant dans des environnements dynamiques. Dans ce cadre, une classification des algorithmes couramment utilisés est proposée. La figure 3 présente une vue d'ensemble de cette classification. Afin de structurer cette revue, la méthode *PRISMA* a été adoptée. L'analyse des publications récentes (cinq dernières années) permet de dégager les tendances actuelles, les limitations des approches existantes et les opportunités pour de nouvelles contributions, notamment dans le contexte de la collaboration humain-robot pour le désassemblage de produits en fin de vie.

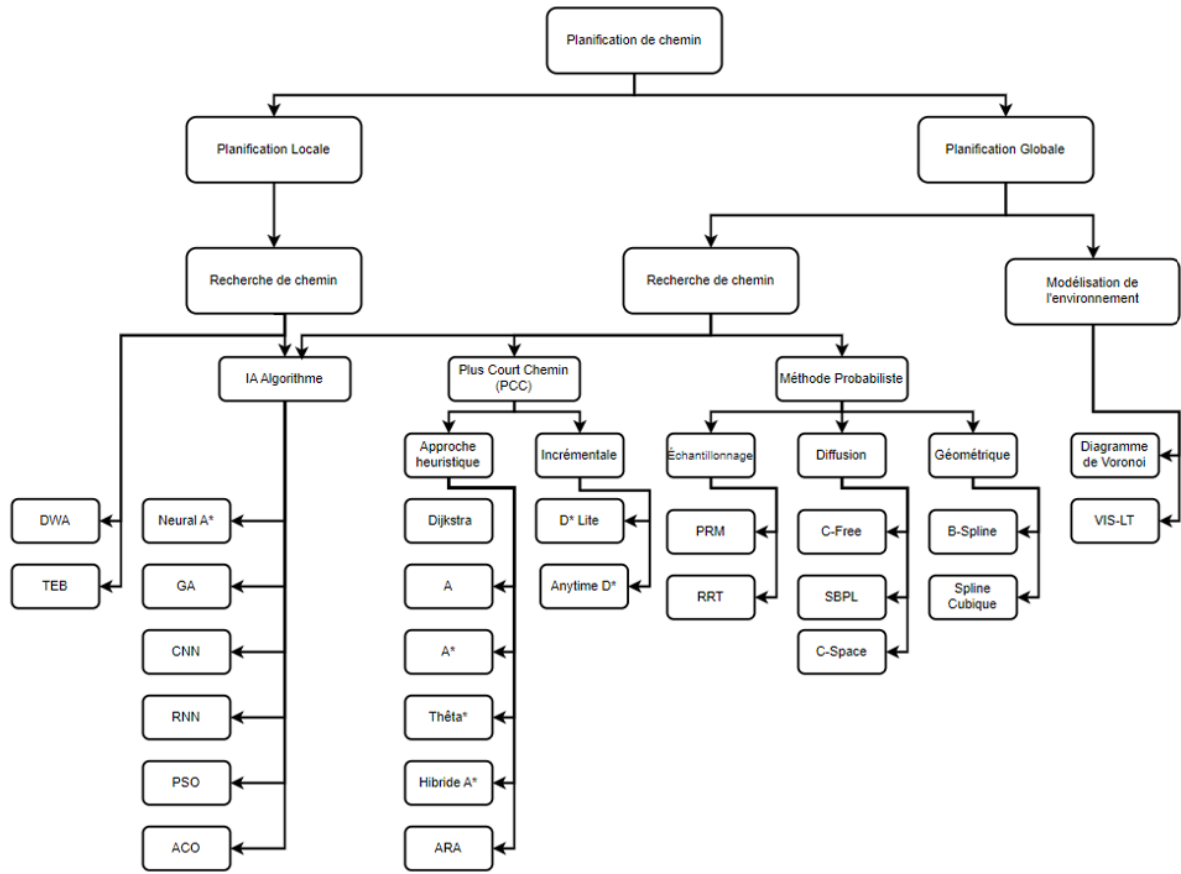


Figure 3 : Taxinomie des algorithmes de planification de chemin

2. METHODOLOGIE DE LA REVUE DE LITTÉRATURE

La revue de littérature a été conduite selon les principes de la méthode *PRISMA* (*Preferred Reporting Items for Systematic Reviews and Meta-Analyses*), assurant une sélection rigoureuse, reproductible et transparente des sources scientifiques. Cette approche systématique se décompose généralement en quatre étapes principales [9].

La première étape consiste à définir une requête de recherche basée sur des mots-clés pertinents, directement liés à la problématique étudiée. Cette requête est ensuite exécutée sur une ou plusieurs bases de données scientifiques afin d'identifier un ensemble initial de références. La deuxième étape concerne l'élimination des doublons afin d'obtenir un corpus de publications unique et non redondant. La troisième étape repose sur une présélection des documents, fondée sur la lecture des titres et

des résumés, permettant d'identifier les études potentiellement éligibles. Enfin, la quatrième étape consiste à effectuer une lecture intégrale des articles retenus, afin de valider définitivement leur inclusion dans le corpus d'analyse.

Dans le cadre de ce projet, la revue a été structurée de manière à examiner de façon progressive l'évolution des approches de planification de trajectoires. L'analyse a débuté par une vision transversale incluant plusieurs domaines applicatifs notamment les véhicules autonomes aériens (UAV), maritimes (USV) et la robotique mobile pour ensuite se recentrer sur les méthodes spécifiquement dédiées à la robotique collaborative et aux bras manipulateurs industriels. Ainsi, une requête globale et inclusive (présentée au tableau 1) a été formulée dans la base Scopus, de manière à capturer à la fois les approches de planification globale et locale, les stratégies d'évitement d'obstacles ainsi que les méthodes d'adaptation aux environnements dynamiques. Le choix d'inclure différents types de plateformes robotiques (UAV, USV, manipulateurs, cobots) visait à identifier des principes et techniques transférables en particulier ceux liés à la perception 3D, à la sécurité et à la replanification en temps réel.

La requête initiale a permis d'identifier 2113 documents publiés entre 1985 et 2025. Un premier filtrage a été appliqué en fonction de plusieurs critères : type de publication (articles scientifiques, conférences), langue (anglais), domaines scientifiques (ingénierie), source (journaux et procédure de conférences) et date de publication. La période de recherche a été délimitée aux cinq dernières années afin de garantir une couverture des travaux les plus récents. Ce filtrage a permis de réduire le corpus à 1345 publications. Pour affiner davantage la sélection sur les enjeux liés à la collaboration homme-robot et à la perception tridimensionnelle, des mots-clés supplémentaires tels que *"human"* et *"3D"* ont été intégrés à la requête. Cette phase a conduit à une sélection finale de 146 documents jugés pertinents.

Enfin, parmi les 146 articles retenus, seuls ceux répondant de manière explicite aux objectifs méthodologiques du projet ont été sélectionnés pour une analyse approfondie. Cette sélection finale s'est fondée sur leur pertinence scientifique, leur applicabilité directe au contexte de la robotique collaborative, ainsi que sur la qualité et la rigueur des approches proposées. Par ailleurs, quelques

documents (de d'autres base de recherche) jugés particulièrement pertinents ont été intégrés à l'analyse malgré une date de publication antérieure à la limite fixée de 2019, en raison de leur forte valeur de référence dans le domaine. L'ensemble du processus suivi pour cette revue systématique est synthétisé à la figure 4.

Tableau 1 : Requête de mots clés utilisés

Mots clés	<i>(("robotic arm" OR "robot manipulator" OR "collaborative robot" OR "cobot" OR "vesel" OR "Unmanned Surface Vehicle" OR "usv" OR "UAV" OR "unmanned aerial vehicle") AND ("path planning" OR "trajector planning" OR "global path planning" OR "local path planning") AND ("dynamic environment" OR "collision avoidance" OR "obstacle avoidance")))</i>
-----------	--

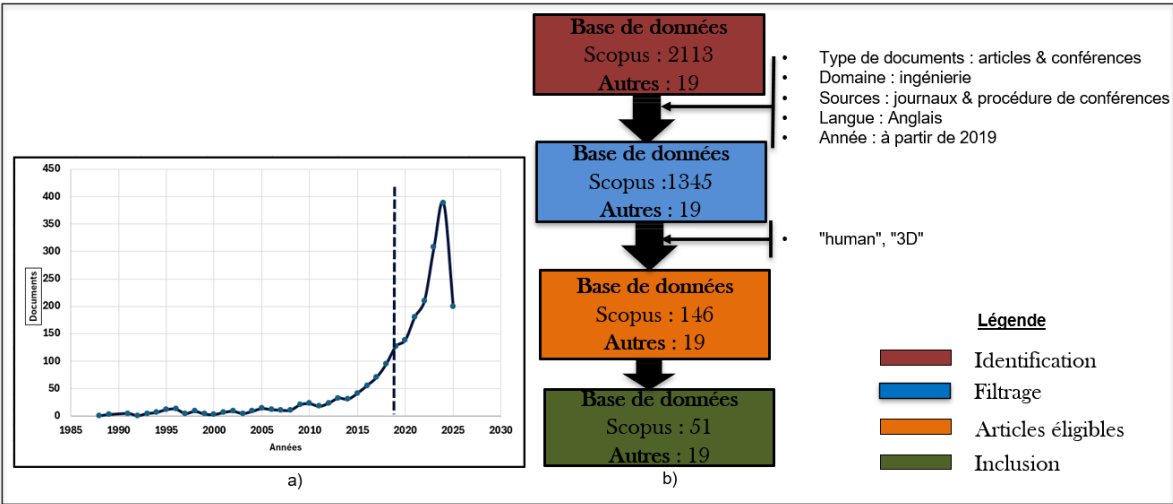


Figure 4 : Revue de la littérature : a) nombre d'articles publiés ; b) Méthodologie PRISMA

2.2.1.ANALYSE DES DOCUMENTS SELECTIONNES

Les documents retenus ont été analysés selon les types d'algorithmes utilisés (planification globale, planification locale, hybridation), les méthodes de perception (capteurs de profondeur, caméras stéréo, capteurs de mouvement) et les techniques d'optimisation (algorithmes de recherche, méthodes heuristiques, apprentissage automatique). Une attention particulière a été accordée aux approches utilisant les algorithmes d'évitement dynamique en raison de leur pertinence pour ce projet.

2.2.1.1. ALGORITHMES TRADITIONNELS ET LEURS LIMITES

Les algorithmes de planification de trajectoire visent à déterminer un chemin optimal permettant à un robot d'atteindre une cible tout en évitant les obstacles et en respectant les contraintes cinématiques et environnementales. Parmi les méthodes les plus répandues, l'algorithme A* [10] et ses variantes telles que Thêta* [11-13] et S-Thêta* [14, 15] restent des piliers de la planification dans des environnements statiques. Basés sur une représentation discrète de l'espace (grilles), ces algorithmes utilisent une fonction heuristique pour guider la recherche du chemin le plus court.

Leur popularité tient à leur simplicité d'implémentation, à leur efficacité en temps de calcul et à leur capacité à produire des trajectoires optimales sur des cartes discrètes. Toutefois, les trajectoires générées par A* présentent un aspect en escaliers, inadapté pour les robots nécessitant des mouvements plus fluides, comme les bras robotiques. C'est pour pallier cette limitation que des variantes comme Thêta* ont été proposées en introduisant une vérification de ligne de visée entre les nœuds. Cela permet de générer des trajectoires plus directes et mieux adaptées aux contraintes mécaniques des systèmes robotiques. Cependant, l'utilisation de ces algorithmes dans des environnements dynamiques présente plusieurs limites. D'une part, la planification initiale est généralement conçue pour un environnement figé, tout changement majeur (déplacement ou

apparition d'obstacles) implique de recalculer entièrement la trajectoire, ce qui est coûteux en temps de calcul. D'autre part, la rigidité de ces approches ne permet pas une adaptation fluide en temps réel ce qui est indispensable dans des contextes de collaboration homme-robot ou dans des environnements industriels évolutifs.

Ces limitations ont motivé l'exploration de solutions plus sophistiquées intégrant une capacité d'adaptation continue et une gestion proactive des obstacles dynamiques. Parmi ces évolutions, les planificateurs incrémentaux tels que D* et D* Lite, les architectures hybrides combinant planification globale et locale ainsi que les algorithmes bio-inspirés ou basés sur l'apprentissage automatique, constituent aujourd'hui des pistes de recherche majeures pour surmonter les contraintes des approches traditionnelles.

2.2.1.2. L'ÉMERGENCE DES APPROCHES HYBRIDES

Pour pallier les limitations des algorithmes traditionnels face aux environnements dynamiques de nombreux chercheurs ont exploré des architectures hybrides combinant une planification globale, visant à définir une trajectoire optimale vers l'objectif et des stratégies locales d'évitement capables de réagir rapidement aux obstacles mouvants. Ces architectures permettent d'associer le meilleur des deux mondes, la capacité de raisonnement global sur la carte de l'environnement et la réactivité face aux imprévus. Ainsi, dans le domaine maritime, les auteurs de l'article [16] proposent une solution combinant A* pour la planification globale avec *Dynamic Window Approach (DWA)* [17, 18] pour l'évitement local. Cette combinaison permet une navigation sûre et robuste dans les environnements marins complexes où les obstacles dynamiques tels que d'autres navires ou les effets de courant sont fréquents. Les résultats démontrent une nette amélioration de la robustesse tout en maintenant des temps de calcul compatibles avec une utilisation en temps réel. De manière similaire, pour les environnements aériens tridimensionnels, l'approche multicouches présentée dans [19] intègre un algorithme de *Q-learning* [20-23] avec des courbes *B-spline*[18]. Cette solution hybride permet de tirer parti à la fois des informations globales fournies par les cartes statiques et des

données locales issues des capteurs embarqués. Le système est capable de générer des trajectoires optimisées et adaptatives montrant une flexibilité bien supérieure aux méthodes purement géométriques. Cette capacité d'adaptation est particulièrement précieuse dans des espaces fortement contraints et en constante évolution comme ceux rencontrés par les *UAV* dans des missions en environnement urbain ou naturel complexe.

Au fil des recherches, ces approches hybrides ont démontré qu'elles constituent un compromis efficace pour permettre aux systèmes autonomes de conjuguer planification stratégique et réactivité opérationnelle une exigence devenue incontournable en robotique industrielle et collaborative.

2.2.1.3. INNOVATIONS ALGORITHMIQUES RÉCENTES

Au cours de la dernière décennie, de nouveaux paradigmes algorithmiques ont émergé, repoussant les limites des méthodes traditionnelles de planification de trajectoire. Parmi ces avancées, l'intégration de techniques issues de l'apprentissage automatique et des méthodes bio-inspirées a permis de concevoir des solutions plus adaptatives et performantes. L'approche Neural A* [24] illustre parfaitement cette tendance. Elle repose sur un encodeur convolutif permettant d'entraîner un réseau de neurones qui apprend, à partir de données, à optimiser la recherche de trajectoire. Contrairement aux algorithmes classiques, qui dépendent de règles prédéfinies et de fonctions heuristiques statiques, Neural A* s'adapte dynamiquement aux environnements qu'il rencontre. Les résultats obtenus montrent qu'il surpasse les méthodes traditionnelles en termes d'optimalité et d'efficacité. Parallèlement, les algorithmes bio-inspirés ont démontré leur pertinence pour résoudre des problèmes de planification multi-objectifs. Les travaux menés autour de l'optimisation par colonie de fourmis [25-27] et par essaim de particules (*PSO*) [28-31] ont permis d'atteindre des résultats prometteurs. Une application particulièrement intéressante est présentée dans [25], où une colonie de fourmis est combinée avec l'algorithme A* afin d'optimiser simultanément la précision du contrôle et la consommation énergétique des *UAV* électriques. Cette

synergie permet d'améliorer la qualité des trajectoires tout en tenant compte des contraintes énergétiques critiques dans les systèmes embarqués.

Ces innovations illustrent l'évolution du domaine vers des méthodes plus souples et intelligentes, capables de s'adapter aux contraintes complexes des environnements réels et de répondre aux exigences croissantes en matière de performance, de sécurité et d'efficacité énergétique.

2.2.1.4. ÉVITEMENT D'OBSTACLES DYNAMIQUES

Dans de nombreux domaines d'application, l'évitement d'obstacles dynamiques constitue un défi clé pour les systèmes autonomes, particulièrement lorsqu'il doit s'intégrer à des cadres réglementaires stricts (vitesse maximale, distance maximale...). C'est notamment le cas dans le domaine maritime, où les manœuvres d'évitement doivent être conformes aux règles internationales *COLREGs*, imposant aux algorithmes de navigation des contraintes supplémentaires. Dans ce contexte, plusieurs travaux ont enrichi les méthodes traditionnelles pour les rendre plus efficaces et conformes. Ainsi, des améliorations de la *Dynamic Window Approach (DWA)* ont été proposées en y intégrant des techniques issues de l'optimisation bio-inspirée. Ainsi, l'algorithme *PSO-CCDWA* [32] combine *DWA* avec un algorithme d'optimisation par essaim de particules (*PSO*), réduisant significativement le temps de navigation dans divers scénarios de rencontre, tout en améliorant la capacité d'adaptation face aux obstacles en mouvement. De manière complémentaire, l'article[33] présente une version améliorée des *Velocity Obstacles (VO)*[34], qui intègre pleinement les contraintes des *COLREGs*. Le système comprend un module avancé de détection de collision et un algorithme de replanification, permettant un évitement simultané de multiples cibles dynamiques avec un haut degré d'efficacité. Au-delà de ces méthodes classiques, les avancées récentes en apprentissage par renforcement (*DRL*) [35-41] ouvrent de nouvelles perspectives pour un évitement d'obstacles plus adaptatif et intelligent. Par exemple, le *Smooth Convergence DRL (SCDRL)* [42], basé sur un *Deep Q-Network* amélioré, simplifie notablement la loi de commande tout en maintenant d'excellentes performances en navigation dynamique. En parallèle, l'approche *PPO* [43] s'appuie sur

un modèle mathématique du comportement des USV et un mécanisme de récompense adapté, permettant une prise de décision rapide et sécurisée en présence d'obstacles mouvants. Enfin, des travaux d'hybridation intelligente ont vu le jour. Le *CS-MPC* [44] illustre parfaitement cette tendance en combinant un contrôle prédictif (*MPC*) avec la logique des *Velocity Obstacles*, le tout dans un cadre unifié. Cette approche permet de renforcer la robustesse du système en environnement fortement encombré, les simulations montrant une stabilité améliorée même face à des obstacles imprévus. Ces avancées démontrent le potentiel du *DRL* et des approches hybrides pour repousser les limites des stratégies classiques d'évitement dynamique, en intégrant davantage d'intelligence et de flexibilité dans le comportement des robots autonomes.

2.2.1.5. OPTIMISATION ÉNERGÉTIQUE ET DISTANCE PARCOURUE POUR UNE AUTONOMIE MAXIMALE

Avec l'essor des systèmes autonomes déployés sur le terrain, la minimisation de la consommation énergétique est devenue un enjeu crucial pour maximiser l'autonomie et garantir des opérations prolongées, notamment pour les *UAV*, *USV*, ou les robots mobiles opérant en environnement industriel. Face à cet impératif, plusieurs approches algorithmiques ont été développées pour optimiser simultanément la trajectoire parcourue et l'efficacité énergétique. Les algorithmes génétiques se sont révélés particulièrement adaptés à ces problématiques multi-objectifs. Parmi eux, *NSGA-II* [45] s'avère être un excellent outils. Cet algorithme permet d'optimiser simultanément la forme de la trajectoire et la consommation énergétique des navires, tout en réduisant de manière significative le coût computationnel par rapport aux méthodes conventionnelles. L'approche multi-objectif de *NSGA-II* garantit ainsi un bon compromis entre performance énergétique et qualité de navigation. Dans le même esprit, l'algorithme *D* Lite amélioré* [46] apporte une contribution intéressante en combinant planification rapide et optimisation énergétique. En élargissant la zone de recherche adjacente et en intégrant une distance de sécurité, cette variante permet de réduire drastiquement le temps de planification, tout en conservant une trajectoire

optimale. Les tests effectués sur des USV réels confirment l'efficacité de cette méthode dans des environnements complexes, démontrant son potentiel pour les applications embarquées. Au-delà des algorithmes de planification pure, certaines approches plus récentes exploitent pleinement les données capteurs pour réaliser une optimisation en temps réel. L'article [47] en est un bon exemple, en utilisant un *Double Deep Q-Network (D2QN)* pour minimiser à la fois la consommation énergétique et la perte de données au cours de la mission. Cette approche met en lumière l'intérêt croissant pour des stratégies de contrôle basées sur les données, où l'intégration intelligente des informations capteurs permet de raffiner les décisions de navigation, en conciliant autonomie énergétique et qualité du service fourni. Ces travaux illustrent l'importance croissante accordée à l'optimisation énergétique dans le domaine de la planification de trajectoire, et ouvrent des perspectives prometteuses pour concevoir des robots autonomes capables d'opérer efficacement dans des scénarios contraints par l'autonomie énergétique.

2.2.1.6. ROBOTIQUE COLLABORATIVE, DÉFIS ET SOLUTIONS POUR LES BRAS INDUSTRIELS

Dans les environnements industriels modernes, les bras robotiques évoluent de plus en plus au sein d'espaces partagés avec les opérateurs humains, ce qui impose des exigences élevées en matière de sécurité, de réactivité et de précision. La planification de trajectoire dans ces contextes dynamiques doit permettre une adaptation en temps réel face à des interactions complexes et imprévisibles. Pour répondre à ces défis, les solutions basées sur une perception 3D avancée associée à des stratégies de replanification rapide se sont largement développées. Par exemple, la méthode proposée dans [48] intègre une caméra Kinect pour acquérir en temps réel un nuage de points 3D représentant l'environnement immédiat du robot. Ce système de perception est couplé à un mécanisme d'échappement aux minima locaux, permettant au bras robotique de naviguer de manière sécurisée parmi de multiples obstacles, fixes ou mobiles, tout en garantissant une fluidité du mouvement. Dans la même logique, l'approche *BP-RRT** [49] illustre une avancée notable en matière

de planification rapide. Cette solution combine un réseau neuronal à rétropropagation avec un *RRT** amélioré [50]. L'innovation clé réside dans la transformation de la recherche globale en une recherche locale, guidée de manière intelligente par le réseau. Cela permet une réduction radicale du temps de calcul, tout en assurant des trajectoires sans collision, cruciales dans les environnements industriels fortement contraints. Au-delà de la perception et de la génération de trajectoires, la sécurité de l'interaction humain-robot constitue un axe de recherche central. Les dernières avancées en cobotique exploitent des méthodes de modélisation dynamique de l'environnement. Par exemple, l'utilisation de champs de potentiel adaptatifs (*APF*) [13, 51-57] permet d'ajuster dynamiquement la force de répulsion en fonction de la configuration actuelle des obstacles. Cette capacité d'adaptation conduit à une replanification rapide et efficace, avec une réduction du temps de calcul de 55 % par rapport aux méthodes classiques. Les auteurs dans [58] combinent un planificateur RRT avec un contrôleur ERG pour gérer en temps réel les contraintes dynamiques, y compris la présence humaine mouvante. Validée expérimentalement sur un robot collaboratif, cette solution offre robustesse, sécurité et efficacité. Sa principale limite réside dans sa complexité d'implémentation et la nécessité de capteurs avancés. Enfin, la modélisation des intentions humaines constitue une avancée déterminante pour garantir des interactions naturelles et sécurisées. L'approche probabiliste proposée dans [59] permet au robot d'anticiper les mouvements des opérateurs humains, en tenant compte de leur comportement probable. Cette anticipation rend l'interaction plus fluide et intuitive, tout en renforçant la sécurité des opérations collaboratives.

2.2.1.7. ANALYSE CRITIQUE ET PERSPECTIVES

Malgré les avancées remarquables constatées au cours de cette revue, plusieurs limites persistent dans les approches actuelles de planification de trajectoire et d'évitement d'obstacles en particulier lorsqu'il s'agit d'applications dans des environnements réels complexes et dynamiques.

Tout d'abord, on observe un écart important entre les performances obtenues en simulation et celles mesurées sur le terrain. Nombre de méthodes performantes en environnement simulé peinent à maintenir leur niveau d'efficacité en conditions réelles, en raison de la variabilité et de l'imprévisibilité

des scénarios industriels. Ce gap simulation-réel constitue un défi majeur pour le transfert technologique des solutions de recherche vers l'industrie. Par ailleurs, la complexité algorithmique de certaines approches avancées, telles que *RMPflow* [60], limite leur déploiement sur des systèmes embarqués ou nécessitant une exécution temps réel stricte. Le coût computationnel élevé de ces méthodes, ainsi que la difficulté d'implémentation et de paramétrage fin, constituent encore des freins importants. De plus, les approches basées sur l'apprentissage par renforcement profond (*DRL*), bien que prometteuses, souffrent d'une forte dépendance à des volumes importants de données d'entraînement, qui doivent être variées et représentatives des situations réelles. Or, la constitution de tels jeux de données demeure complexe, coûteuse et peu reproductible selon les cas d'application. Face à ces constats, plusieurs axes de recherche prometteurs émergent. D'abord, le développement d'architectures neuro-symboliques, combinant la capacité de raisonnement logique des méthodes symboliques avec la souplesse d'apprentissage des réseaux de neurones, pourrait permettre de concevoir des planificateurs plus génériques et capables de mieux généraliser à des scénarios nouveaux. Ensuite, l'optimisation embarquée représente un enjeu clé : le développement d'algorithmes légers et efficaces, compatibles avec les contraintes des processeurs embarqués sur les plateformes robotiques industrielles, facilitera l'adoption des solutions de planification avancée dans le milieu productif. Dans cette perspective, l'approche adoptée dans ce projet, combinant les algorithmes *Thêta** et *D* Lite* avec l'utilisation d'une costmap dynamique 3D, constitue une piste de solution particulièrement adaptée aux environnements industriels évolutifs. En permettant une planification globale efficace avec *Thêta**, une replanification locale rapide et incrémentale avec *D* Lite*, et une intégration en temps réel des obstacles mobiles via la costmap dynamique, cette architecture répond aux exigences de réactivité et de sécurité indispensables en robotique collaborative. De plus, elle offre un compromis intéressant entre simplicité de mise en œuvre et robustesse, facilitant son intégration dans des systèmes embarqués. Le tableau 2 présente la comparaison de cette approche avec quelques travaux proches.

Enfin, la mise en place de benchmarks unifiés et de protocoles d'évaluation standardisés est devenue indispensable. Actuellement, la diversité des scénarios, des métriques et des méthodologies d'évaluation rend difficile la comparaison rigoureuse des performances entre les différentes

approches proposées dans la littérature. Des initiatives en ce sens permettront de renforcer la reproductibilité des résultats et de guider les futures évolutions du domaine.

En somme, si les progrès réalisés en matière de planification de trajectoire et d'évitement dynamique sont indéniables, de nombreux défis restent à relever pour garantir des systèmes autonomes sûrs, robustes et efficaces en conditions réelles. Les approches hybrides associées à des mécanismes de perception et de replanification dynamiques, comme celle proposée dans ce mémoire, constituent des pistes particulièrement prometteuses pour relever ces défis.

3. CONCLUSION

Cette revue de la littérature a permis de dégager les principales tendances et limites des approches actuelles en planification de trajectoires pour la robotique autonome, notamment en contexte collaboratif. Face à la complexité croissante des environnements industriels, les solutions hybrides combinant perception 3D, planification adaptative et sécurité renforcée apparaissent comme les plus prometteuses. L'architecture proposée dans ce projet, articulée autour de Thêta*, D* Lite et d'une costmap dynamique, s'inscrit dans cette perspective. Le chapitre suivant présente la méthodologie détaillée de mise en œuvre de ce système, depuis la modélisation de l'environnement jusqu'à l'exécution robotique.

Tableau 2 : Comparaison de quelques travaux de recherche avec notre approche

Références	Méthodes	Matériels	Présence humaine	Types d'obstacles		Commentaires
				Statique	Dynamique	
Notre approche	Thêta* + D*Lite, TEB	UR5	✓	✓	✓	Chemins optimaux, adaptation en temps réel aux changements, collaboration sûre et efficace entre les humains et les robots.
Ramy Meziane et al.2017 [59]	Réseau Neuronal, polynôme quintique		✓		✓	Trajectoires d'évitement d'obstacles fluides. Complexe, Inefficacité énergétique, la réactivité en temps réel limité, généralisation limitée des résultats
Merckaet et al.2024 [58]	RRT + ERG (planification en temps réel)	Franka Emika Panda	✓	✓	✓	Compatible avec des environnements dynamiques, maintient les contraintes de sécurité. Peut-être piégé dans des minima locaux dans des environnements complexes
Law-Kam Cio et al.2019 [48]	Stéreo vision +Suivioculaire	JACO (Kinova Robotics)	✓	✓	✓	Faible coût de mise en œuvre, intuitive pour l'utilisateur. Sensible aux conditions d'éclairage et à la complexité des scènes.
Kamali et al.2020 [49]	DRL (PPO) + Téléopération VR	ABB IRB 120	✓	✓	✓	Réactivité en temps réel, apprentissage par simulation. Nécessite un temps d'entraînement considérable, dépend de la qualité de la simulation.
C. A. Cheng et al.2021 [60]	RMPflow (planification géométrique)	Manipulateurs multi-tâches	✓	✓	✓	Modélisation géométrique flexible, stabilité garantie. Complexité mathématique élevée, difficile à implémenter.
S. Secil et al.2023 [51]	Planification par capsules	Manipulateur industriel	✓	✓	✓	Méthode simple et efficace pour éviter les collisions. Sensible aux erreurs de détection de position de l'humain.

CHAPITRE 3 MÉTHODOLOGIE

3.1. INTRODUCTION

Dans le domaine de la robotique industrielle, l'émergence des environnements collaboratifs soulève de nouveaux défis en matière de planification de trajectoires, notamment lorsqu'il s'agit de garantir la sécurité, l'adaptabilité et la précision des mouvements en présence d'obstacles dynamiques, tels que les opérateurs humains. C'est dans ce contexte que s'inscrit le présent projet, dont le but est de concevoir un système intelligent capable de générer et d'exécuter des trajectoires optimisées pour un bras robotique évoluant dans un espace partagé. Afin de répondre à cet enjeu, ce chapitre expose la méthodologie développée pour mettre en œuvre une architecture complète et fonctionnelle de planification de trajectoire. L'objectif principal est de permettre au robot de produire des trajectoires sûres, réactives et cinématiquement réalisables, même en présence de perturbations imprévues. Le système repose sur une approche hybride qui combine des techniques issues de la planification discrète, de la robotique mobile et de la perception 3D, tout en intégrant les spécificités liées aux bras manipulateurs industriels.

La démarche retenue se décline en plusieurs étapes structurantes, la modélisation tridimensionnelle de l'environnement à partir de capteurs stéréo (3.3.1, 3.3.2), la construction d'une costmap dynamique basée sur OctoMap (3.3.4), la planification multi-niveau combinant les algorithmes Thêta* et D* Lite pour le niveau global et TEB pour le niveau local (3.3.5), ainsi que l'exécution des trajectoires dans MoveIt2 via un module de cinématique inverse. Tous ces composants sont implémentés et orchestrés dans l'environnement ROS2 (3.3.3), garantissant une modularité, une interopérabilité et une évolutivité du système.

3.2. HYPOTHÈSES DE RECHERCHE

Afin d'orienter le développement méthodologique et d'évaluer la pertinence des choix technologiques, deux hypothèses principales ont été formulées dans le cadre de ce projet. Ces hypothèses sont issues d'une analyse approfondie de la littérature et s'inscrivent dans les objectifs spécifiques liés à l'optimisation des trajectoires robotiques en environnement industriel collaboratif.

Hypothèse 1 : La création d'une cartographie hybride, reposant sur la fusion d'une carte de coûts 3D statique issues d'OctoMap et d'une carte dynamique (mise à jour en temps réel), permet d'améliorer significativement la capacité d'évitement des collisions en fournissant une représentation plus riche et actualisée de l'environnement ;

Hypothèse 2 : L'utilisation conjointe des algorithmes Thêta* et D* Lite dans le processus de planification globale améliore la réactivité du système face aux changements d'environnement, tout en garantissant des trajectoires plus précises et mieux adaptées aux contraintes dynamiques du milieu industriel.

Dans le prolongement de ces hypothèses, la section suivante décrit en détail la méthodologie adoptée pour valider ces postulats, en mettant en œuvre une architecture logicielle intégrée combinant cartographie 3D dynamique, planification hybride et exécution robotique dans un environnement collaboratif simulé.

3.3. MÉTHODOLOGIE

La méthodologie mise en œuvre dans ce projet repose sur une architecture logicielle intégrée, conçue pour permettre la planification et l'exécution de trajectoires optimisées dans un environnement industriel collaboratif dynamique. Cette approche combine modélisation 3D de l'environnement, cartographie probabiliste, planification hybride multi-niveaux et contrôle robotique en temps réel, le tout sous ROS2 avec MoveIt2 et un bras robotique UR5 simulé.

L'environnement de travail est reconstruit en trois dimensions à l'aide d'une caméra stéréo Intel RealSense D455. Les données de profondeur acquises sont traitées par la bibliothèque OctoMap

pour générer une carte d'occupation probabiliste structurée en octree. Cette carte est ensuite transformée en une costmap dynamique 3D, intégrant en temps réel les obstacles fixes, les objets mobiles et les interactions humaines potentielles.

La planification globale s'appuie sur une synergie entre les algorithmes Thêta* et D* Lite. Thêta* est utilisé pour produire une trajectoire initiale directe et fluide tandis que D* Lite permet une replanification incrémentale rapide lors de modifications détectées dans l'environnement. Cette combinaison assure un compromis efficace entre optimalité initiale et adaptabilité en cours d'exécution.

Pour la planification locale, l'algorithme TEB (Timed Elastic Band) est employé afin d'ajuster dynamiquement la trajectoire à proximité du robot. Ce planificateur local intègre les contraintes cinématiques spécifiques au bras UR5 et permet une adaptation réactive face aux obstacles dynamiques en affinant le chemin planifié en continu.

Une fois la trajectoire validée, celle-ci est transmise à MoveIt2 via ROS2. Un module de calcul de la cinématique inverse convertit les points de la trajectoire en commandes articulaires, assurant ainsi une exécution fluide, précise et sécurisée par le robot.

Cette méthodologie illustrée à la figure 5, articulant perception 3D, planification hybride et exécution robotique, constitue le socle technique du système développé. Les sections suivantes détaillent les composants clés de cette architecture, en commençant par la cartographie tridimensionnelle de l'environnement.

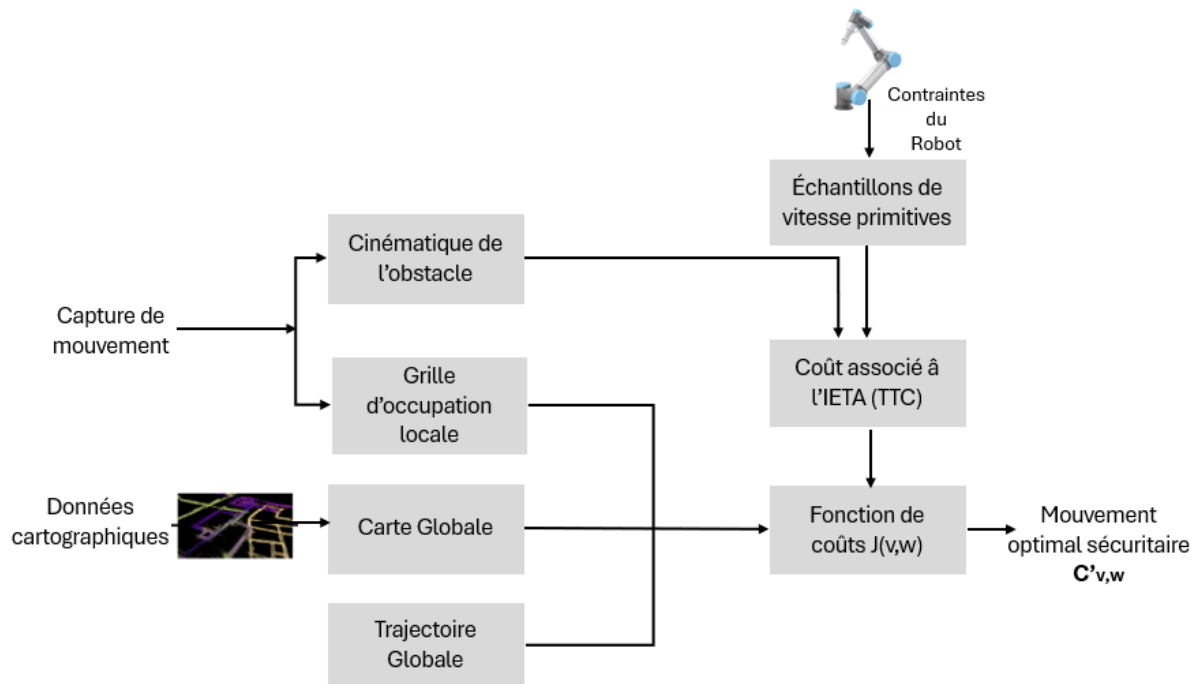


Figure 5 : Diagramme descriptif de la méthode

3.3.1. CARTOGRAPHIE 3D DE L'ENVIRONNEMENT

3.3.1.1. ACQUISITION DES DONNEES PAR LA CAMERA INTEL REALSENSE D455

Dans le cadre de ce projet, l'acquisition de données tridimensionnelles de l'environnement est assurée par la caméra Intel RealSense D455, un capteur stéréo actif spécialement conçu pour la perception 3D en temps réel dans des environnements dynamiques. Cette caméra intègre trois modules fonctionnels complémentaires :

- Deux caméras infrarouges (*IR*) munies de capteurs à obturateur global (*global shutter*), utilisées pour l'estimation de la profondeur via le principe de la stéréovision active ;
- Une caméra *RGB* dotée d'un capteur à obturateur roulant (*rolling shutter*) dédiée à la capture d'images couleur haute résolution, facilitant l'analyse visuelle et l'enregistrement contextuel ;

- Une unité de mesure inertielle (*IMU*), composée d'un accéléromètre et d'un gyroscope, fournissant des données de mouvement et de position permettant de stabiliser et synchroniser les données spatiales obtenues.

Cette architecture hybride permet de générer un nuage de points dense, précis et cohérent, tout en assurant une synchronisation temporelle des informations issues de la profondeur, de la couleur et du mouvement. Grâce à la stéréovision renforcée par un projecteur infrarouge, la caméra D455 est capable de maintenir une perception fiable même dans des conditions d'éclairage difficile ou sur des surfaces faiblement texturées. Les données collectées sont ensuite publiées sous forme de messages ROS2 de type *sensor_msgs/PointCloud2*, servant de base à la construction de la carte 3D de l'environnement et à la détection des obstacles en temps réel.

Les sous-sections suivantes détaillent deux aspects techniques essentiels du fonctionnement de ce capteur, le principe de mesure de profondeur et la gestion des zones de profondeur invalide.

3.3.1.1.1. PRINCIPE DE LA MESURE DE PROFONDEUR

La mesure de profondeur réalisée par la caméra Intel RealSense D455 repose sur le principe fondamental de la stéréovision. Ce procédé consiste à capter simultanément une scène à partir de deux caméras infrarouges placées à une distance connue ("*baseline*") l'une de l'autre. Chaque caméra enregistre une image de la scène sous un angle légèrement différent, ce qui permet de détecter une différence de position apparente d'un même point dans les deux images, appelée disparité. À partir de cette disparité, il est possible de calculer la distance d'un point par rapport à la caméra en utilisant la relation géométrique de l'équation (3.1) [61]. Dans cette relation, Z représente la profondeur du point (distance à la caméra), f est la distance focale de la caméra (en pixels), B est la distance entre les deux capteurs infrarouges (*baseline*) et d est la disparité, c'est-à-dire la différence de position du point entre les deux images.

$$Z = \frac{f \times B}{d}, \quad (3.1)$$

Cette méthode permet de générer une carte de profondeur pixel par pixel dans laquelle chaque valeur encode la distance entre la caméra et les objets de la scène. Pour renforcer la qualité de la mesure, la D455 est équipée d'un projecteur infrarouge qui projette un motif structuré sur la scène améliorant la précision du calcul de disparité notamment sur des surfaces peu texturées. Les données de profondeur obtenues sont ensuite combinées avec les informations issues de l'*IMU* et de la caméra *RGB* pour produire un nuage de points 3D cohérent, utilisé dans la suite du processus de cartographie et de planification de trajectoire. Ce mécanisme constitue une étape clé dans la perception de l'environnement en robotique mobile et collaborative. Elle permet également une estimation dense de la profondeur sur toute la scène observée, dans un champ de vision horizontal (FOV_H) et un champ de vision vertical selon les caractéristiques matérielles de la D455. L'angle de vision de la profondeur, déterminé par l'intersection de ces deux champs de vision gauche et droit, est défini par l'équation (3.2) [61].

$$Depth\ FOV(Z) = \frac{HFOV}{2} + \tan^{-1} \left(\tan \left(\frac{HFOV}{2} \right) - \frac{B}{Z} \right) \quad (3.2)$$

3.3.1.1.2. GESTION DE LA BANDE DE PROFONDEUR INVALIDE

L'un des défis majeurs associés à l'utilisation de caméras stéréo actives, telles que la RealSense D455, réside dans la gestion des zones de profondeur invalide, c'est-à-dire des portions d'image pour lesquelles aucune mesure de profondeur fiable ne peut être calculée. Ces zones, également appelées bande morte (*dead zone*), apparaissent notamment lorsque la disparité est trop faible ou trop élevée pour être détectée ou lorsque les surfaces observées sont uniformes, transparentes ou trop proches. La bande invalide est principalement liée à deux facteurs techniques :

- La distance minimale de mesure, qui dépend de la baseline du système. Pour la D455 cette distance est d'environ 0,4 mètre. Les objets situés en deçà de cette distance ne produisent pas une disparité suffisante pour être interprétée correctement ;

- Le manque de texture sur certaines surfaces qui rend difficile l'appariement de points entre les deux images IR. Ce problème est partiellement compensé par le projecteur IR embarqué mais reste présent dans des conditions particulières (scènes sombres, reflets, matériaux transparents).

Afin d'atténuer les effets de cette bande de profondeur invalide, plusieurs techniques sont mises en œuvre :

- L'interpolation spatiale pour estimer les valeurs manquantes à partir des pixels voisins valides, selon une logique de continuité spatiale ;
- Le filtrage temporel qui exploite la redondance entre les images successives pour stabiliser les mesures et combler les lacunes transitoires ;
- L'intégration avec l'IMU, qui permet de prédire la position relative de la caméra entre deux acquisitions améliorant ainsi la cohérence spatiale dans les zones à faible fiabilité.

Dans le contexte du présent projet, la qualité du nuage de points est essentielle pour la cartographie 3D et la planification de trajectoire. Une attention particulière est donc portée à la détection et au traitement de ces bandes invalides, afin d'assurer une représentation fiable de l'environnement du robot. Des paramètres spécifiques de la D455 ont été ajustés pour maximiser la densité des points utiles tout en minimisant les artefacts liés à la bande morte. Pour caractériser cette zone, on utilise un indicateur quantitatif appelé *Depth Band Ratio (DBR)* défini suivant l'équation (3.3) [61] avec soit l'utilisation du *HFOV* équation (3.4) ou soit avec l'utilisation de la focale *F* en pixels équation (3.5).

$$DBR = \frac{\text{Largeur invalide}}{\text{Largeur totale image}}, \quad (3.3)$$

$$DBR = \frac{B \times F}{2 \times Z \tan\left(\frac{HFOV}{2}\right)}, \quad (3.4)$$

$$DBR = \frac{B \times F}{Z \times HRES}, \quad (3.5)$$

où, B est la distance entre les deux capteurs infrarouges (*baseline*), F est la focale en pixels, $HRES$ la résolution horizontale de l'image, et $HFOV$ le champ de vision horizontal. Cela permet d'évaluer la largeur de la zone morte en pixels selon la distance Z , un paramètre crucial pour choisir les régions les plus fiables de l'image dans la génération du nuage de points.

3.3.2. CONSTRUCTION DE LA CARTE 3D AVEC OCTOMAP

Une fois les données de profondeur acquises par la caméra Intel RealSense D455, celles-ci sont converties en nuages de points tridimensionnels représentant fidèlement l'environnement immédiat du robot. Afin de structurer ces données brutes en une représentation exploitable pour la planification de trajectoires, une carte 3D probabiliste est générée à l'aide de la bibliothèque OctoMap couramment utilisée dans les systèmes ROS2 pour la modélisation volumétrique d'environnements [62, 63]. OctoMap repose sur une structure arborescente appelée octree, où chaque nœud peut être subdivisé en huit sous-nœuds. Cette hiérarchie permet une résolution adaptative, les zones homogènes (entièrement libres (en blanc) ou occupées (en bleu)) sont fusionnées, tandis que les régions complexes sont subdivisées pour plus de précision. Cette configuration est illustrée à la figure 6 **Erreur! Source du renvoi introuvable.** et basé sur les travaux de Hornung et al. [62] où le modèle volumique est présenté à gauche et la représentation arborescente correspondante à droite.

Chaque voxel est associé à une probabilité d'occupation $P(n)$ estimée de manière bayésienne à partir des observations fournies par les capteurs. Cette approche permet de représenter efficacement des environnements complexes tout en maintenant une charge mémoire modérée grâce à la compression implicite de l'arbre. La mise à jour de la probabilité d'occupation d'un voxel $P(n)$ après réception d'une mesure z_t repose sur la règle bayésienne [64] selon l'équation (3.6).

$$P(n|z_{1:t}) = \left[1 + \frac{1-P(n|z_t)}{P(n|z_t)} \frac{1-P(n|z_{1:t-1})}{P(n|z_{1:t-1})} \frac{P(n)}{1-P(n)} \right]^{-1} \quad (3.6)$$

Pour garantir une mise à jour additive, la probabilité est transformée en log-odds suivant les équations (3.7) et (3.8).

$$L(n|z_{1:t}) = L(n|z_{1:t-1}) + L(n|z_t), \quad (3.7)$$

$$L(n|z_{1:t}) = \log \frac{P(n)}{1-P(n)}, \quad (3.8)$$

Cette formulation permet de simplifier les calculs et d'intégrer les nouvelles observations de façon incrémentale tout en conservant une mémoire des données antérieures. Sur cette base, chaque nœud de l'octree peut être classé comme étant :

- Occupé, le voxel est détecté comme contenant un obstacle c'est-à-dire le faisceau du capteur s'arrête à cet endroit ;
- Libre, le voxel est traversé sans obstruction par le faisceau donc mesuré comme vide ;
- Inconnu, aucune mesure n'a encore été enregistrée ou les données sont jugées trop incertaines.

Cette structure permet une mise à jour efficace en temps réel de l'environnement observé ce qui est crucial pour les applications en robotique dynamique. En traitant les mesures bruitées de manière probabiliste et en intégrant l'historique des observations OctoMap offre une solution robuste et flexible pour la modélisation 3D d'environnements évolutifs élément fondamental pour une planification de trajectoire fiable.

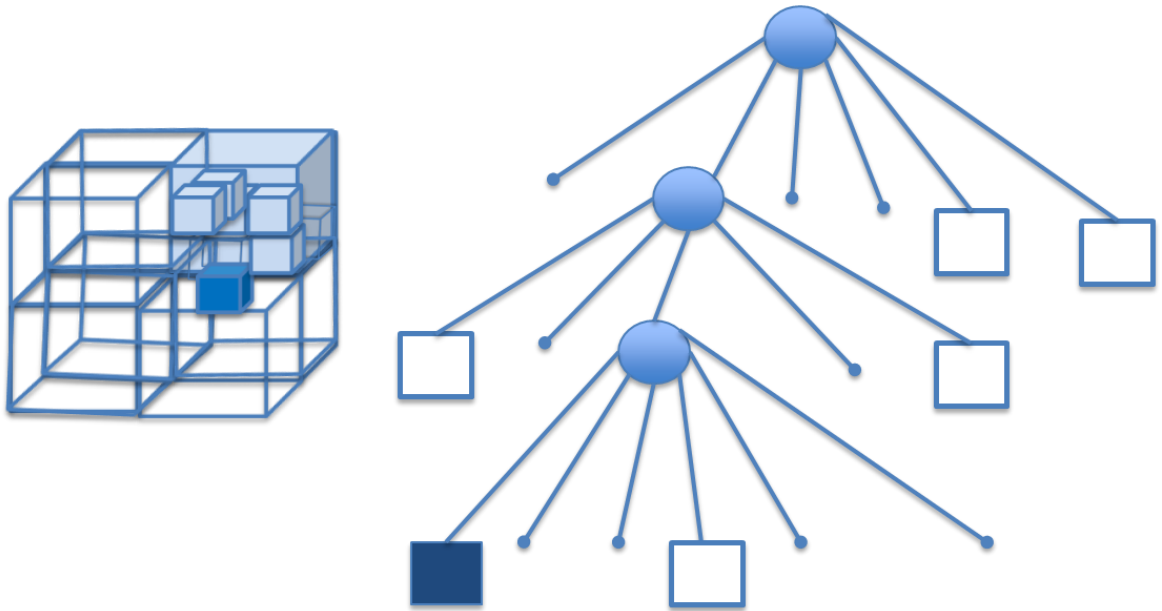


Figure 6 : Exemple d'octree stockant des cellules libres (blanc) et occupées (bleu) dans OctoMap (modifié)[62]

3.3.3. INTÉGRATION DANS ROS2 ET VISUALISATION

L'intégration de la carte 3D dans le système ROS2 constitue une étape clé pour permettre la perception, la planification et l'exécution de trajectoires robotisées dans un environnement en évolution. Cette étape vise à connecter de manière cohérente les modules de perception, de cartographie et de contrôle, tout en assurant une visualisation interactive des données en temps réel. Dans ce projet, les données issues de la caméra Intel RealSense D455 sont d'abord publiées sous forme de nuages de points au format *sensor_msgs/PointCloud2*, un standard ROS2 pour les données 3D. Ces points sont ensuite traités par le nœud *octomap_server* qui convertit en continu les flux de données en une carte OctoMap intégrée à l'environnement ROS2. Le serveur OctoMap publie plusieurs topics essentiels dont */octomap_binary*, */octomap_full* et */occupied_cells_vis_array*, permettant d'accéder à différentes représentations de la carte (compressée, complète, ou visualisable).

Pour assurer une visualisation intuitive, l'environnement est affiché dans RViz2, un outil de visualisation standard de ROS2. RViz2 permet de superposer en temps réel la carte 3D, le modèle

du robot UR5 (via *robot_state_publisher*), les *TF* (transformations de cadre de référence) ainsi que les trajectoires planifiées. Cela facilite à la fois le diagnostic du comportement du robot et l'analyse de la cartographie générée, notamment grâce à l'affichage des voxels occupés et libres dans l'environnement simulé. Un ensemble de *TF* statiques est également défini pour établir la chaîne de transformations spatiales entre les différents éléments de la scène, notamment entre les frames *map*, *base_link*, *tool0* et *camera_link*. Ces *TF* garantissent la cohérence spatiale des données perçues et permettent d'effectuer des projections géométriques précises entre les cartes, les capteurs et l'effecteur du bras robotique. Ainsi, l'intégration complète dans ROS2 fournit une infrastructure logicielle modulaire et extensible assurant l'interopérabilité entre la perception, la cartographie, la planification et la commande. Elle constitue un socle robuste pour le développement de trajectoires adaptatives et sécurisées dans un environnement collaboratif.

3.3.4. CRÉATION DU COSTMAP STATIQUE ET DYNAMIQUE

La génération d'une costmap (ou carte de coût) constitue une étape cruciale pour la planification de trajectoire en environnement semi-structuré ou dynamique. Dans ce projet, la costmap est utilisée comme support pour les algorithmes de planification afin de quantifier l'accessibilité spatiale autour du robot UR5 et de faciliter l'évitement d'obstacles. La construction de la costmap comme illustrée à la figure 7 s'appuie sur les données de la carte OctoMap elle-même dérivée des nuages de points fournis par la caméra stéréo RealSense D455. OctoMap offre une représentation 3D hiérarchique de l'environnement basée sur une structure d'arbre octal dans laquelle chaque voxel est associé à une probabilité d'occupation. À partir de cette carte, deux types de *costmaps* sont générés :

- La costmap statique, construite à partir des parties stables et persistantes de l'environnement. Elle sert de référence pour la planification globale, en considérant les obstacles fixes comme les murs, les machines ou les zones interdites ;

- La costmap dynamique, qui prend en compte les variations en temps réel détectées par la caméra (mouvements d'humains, objets mobiles, etc.). Elle est continuellement mise à jour à l'aide des données perçues, ce qui permet une adaptation rapide de la trajectoire du robot.

L'ensemble de la costmap est représenté sous forme de grilles tridimensionnelles dans lesquelles chaque cellule est affectée d'un coût codé sur une échelle de 0 (zone libre) à 100 (obstacle infranchissable), avec des zones tampon (inflation) autour des obstacles pour anticiper les erreurs de localisation ou les imprécisions mécaniques du robot. Cette représentation 3D est ensuite transformée en un format exploitable par les planificateurs, en particulier Thêta*, D* Lite et TEB. Elle permet de moduler le comportement de ces algorithmes en intégrant les contraintes spatiales de l'environnement perçu en temps réel. La capacité à maintenir cette costmap dynamique à jour constitue un levier essentiel pour garantir la sécurité, l'autonomie et la robustesse du système dans un contexte industriel collaboratif.

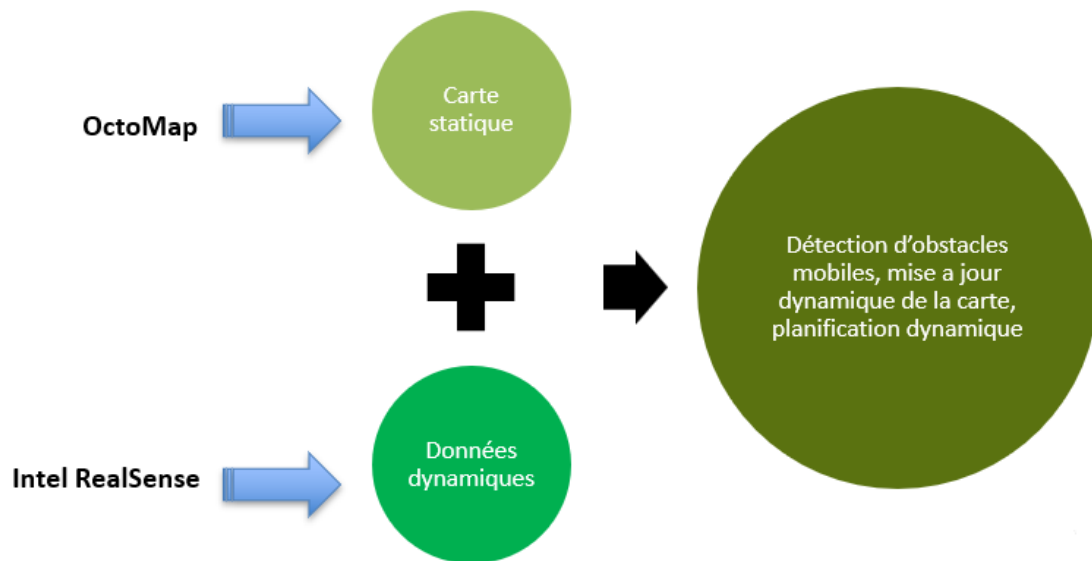


Figure 7 : Création du costmap dynamique

3.3.4.1. FONCTIONNEMENT ET ARCHITECTURE DU NŒUD DYNAMICCOSTMAPNODE

Le nœud *DynamicCostmapNode* constitue un composant central de l'architecture logicielle dédiée à la gestion de la costmap 3D dynamique. Intégré dans l'environnement ROS2, ce nœud assure la transformation en temps réel des données de perception en une carte de coûts exploitable par les planificateurs de trajectoires (Thêta*, D* Lite et TEB). Ce processus repose sur une chaîne logicielle modulaire interconnectée au système de perception composé de la caméra Intel RealSense D455 et de la bibliothèque OctoMap. Les données brutes issues du capteur, transmises sous forme de messages *sensor_msgs/PointCloud2* sont d'abord traitées par OctoMap pour produire une carte d'occupation probabiliste. Le nœud *DynamicCostmapNode* prend ensuite en charge ces données pour générer une grille de voxels discrète (*voxel grid*) dont chaque cellule est associée à un coût de navigation. Le fonctionnement de ce nœud s'articule autour des étapes suivantes :

- Réception et transformation des données, les messages */octomap_pointcloud* sont reçus puis convertis dans le repère commun *map* via le service *TF2* ;
- Filtrage spatial, un prétraitement élimine les points aberrants, redondants ou hors zone d'intérêt afin d'alléger la charge de calcul et de préserver la qualité de la carte ;
- Discrétisation et classification, les points filtrés sont projetés dans la grille 3D selon la résolution définie. Chaque voxel est classé comme libre, occupé ou inconnu, en fonction des probabilités d'occupation issues d'OctoMap ;
- Encodage des coûts, un poids est attribué à chaque voxel, en tenant compte de sa proximité avec les obstacles. Les zones proches des objets sont affectées d'un coût plus élevé, tandis que les zones libres ont un coût nul ;
- Publication ROS2, la costmap 3D finale est encapsulée dans un message *sensor_msgs/PointCloud2* et publiée sur le topic */fusion_costmap*. Elle est accessible par les modules de planification globaux et locaux ;

- Diagnostics, le nœud fournit des statistiques de fonctionnement, utiles pour la supervision et l'optimisation du système.

Cette architecture modulaire permet d'assurer une actualisation continue de la carte 3D avec une latence minimale, tout en garantissant la compatibilité avec les algorithmes de planification du système (Thêta*, D* Lite et TEB). En outre, elle offre une capacité d'extension permettant d'intégrer d'autres sources de perception (lidar, caméra RGB-D, capteurs ultrason) et de mettre en œuvre des logiques de fusion multisensorielle. Le système est ainsi capable de s'adapter dynamiquement aux modifications de l'environnement, assurant une planification robuste et sécurisée dans des contextes industriels collaboratifs.

Afin de renforcer la capacité du système à anticiper les risques de collision en contexte industriel collaboratif, la costmap dynamique est enrichie par l'intégration d'un module de prédiction du risque nommé IETA (*Interference Estimated Time of Arrival*), également interprétable comme une couche *Time-To-Collision* (TTC) simplifiée. La section suivante détaille le fonctionnement et le rôle stratégique de ce module au sein de l'architecture globale.

3.3.4.2. INTÉGRATION DU MODULE IETA SIMPLIFIÉ (OU TTC)

Dans un contexte de robotique collaborative, la sécurité des opérateurs humains constitue un enjeu fondamental. Pour renforcer cette sécurité tout en maintenant la réactivité du système, un module simplifié de type *TTC* (*Time To Collision*) [65, 66], inspiré du concept *IETA* (*Interference Estimated Time of Arrival*) [67] a été intégré dans l'architecture logicielle. Ce module agit comme une couche logique complémentaire, superposée à la costmap dynamique 3D, afin de fournir une évaluation continue du risque de collision entre le bras robotique et des obstacles mobiles, notamment les membres humains. Le principe de fonctionnement repose sur trois éléments d'entrée :

- Le nuage de points dynamique, publié sur `/fusion_costmap`, issu du nœud `DynamicCostmapNode` ;

- Les contraintes cinématiques du robot UR5, incluant la vitesse maximale admissible et l'accélération ;
- Le chemin planifié, récupéré depuis le topic */planned_path* (type *nav_msgs/Path*), mis à jour en continu par le planificateur global.

À partir de ces données, le module calcule une estimation du temps minimal nécessaire pour qu'un obstacle mobile entre en contact avec l'une des parties du robot. Cette estimation est effectuée pour chaque segment du bras UR5 et comparée au temps estimé d'arrivée du robot à la même position. Si une collision potentielle est détectée, une alerte est générée, deux messages ROS sont alors publiés :

- */ieta_feedback*, un niveau de risque continu, codé sous forme numérique ;
- */ieta_trigger_replan*, un booléen déclenchant une replanification immédiate si un seuil critique est franchi (ici 0.5).

Ce mécanisme est particulièrement utile pour, renforcer la sécurité dans des environnements partagés (présence humaine à proximité), prévenir les risques d'interférence dans les zones à forte incertitude dynamique et d'alimenter les algorithmes de planification avec un retour contextuel orienté sécurité.

Bien que ce module soit qualifié de « simplifié » dans cette version, il pose les bases pour une extension future vers un module IETA complet, capable d'intégrer des modèles prédictifs du comportement humain, ou des mécanismes d'apprentissage adaptatif. Il s'agit donc d'un composant clé pour la réactivité contextuelle du robot dans une architecture collaborative temps réel.

Figure 8 supprimée.

Dans la continuité de cette architecture de perception et de cartographie adaptative, la prochaine section s'attache à décrire les mécanismes de planification de trajectoire qui exploitent ces données en temps réel. Elle détaille la complémentarité entre les planificateurs globaux (Thêta* et D* Lite) et le planificateur local TEB, assurant une navigation efficace et sécurisée du robot UR5 dans un environnement industriel collaboratif en constante évolution.

3.3.5. PLANIFICATEUR GLOBAL ET LOCAL

La planification de trajectoire repose sur une stratégie hiérarchisée combinant deux niveaux complémentaires, un planificateur global, responsable de la génération d'un chemin optimal à l'échelle de l'environnement complet et un planificateur local, chargé d'ajuster dynamiquement ce chemin en fonction des obstacles immédiats et des contraintes cinématiques du robot UR5. Cette séparation permet de conjuguer raisonnement stratégique à long terme et réactivité opérationnelle en temps réel, répondant ainsi aux exigences des environnements industriels collaboratifs.

3.3.5.1. PLANIFICATEUR GLOBAL

Le planificateur global repose sur l'articulation de deux algorithmes complémentaires. Thêta*, utilisé pour générer une trajectoire initiale fluide et D* Lite, chargé d'actualiser cette trajectoire de manière incrémentale en cas de modifications de l'environnement. Cette double approche permet de répondre aux exigences d'optimalité, de réactivité et d'efficacité nécessaires dans un contexte industriel dynamique.

3.3.5.1.1. ALGORITHME THÊTA*

Thêta* est une extension du célèbre algorithme A* qui introduit un mécanisme de ligne de visée directe pour contourner la rigidité des chemins en escalier générés sur des grilles. Cela permet de produire des trajectoires plus courtes et plus naturelles, mieux adaptées aux contraintes de mouvement des bras robotiques. Comme A*, Thêta* repose sur une fonction de coût total définie par l'équation (3.9) [68].

$$f(n) = g(n) + h(n) \quad (3.9)$$

Où, $g(n)$ est le cout réel du point de départ au nœud courant n et $h(n)$ la fonction heuristique qui représente le cout estimé du point intermédiaire n au point d'arrivée.

Dans le processus de mise à jour de Θ^* , pour chaque nœud s , au lieu de considérer uniquement son voisin immédiat s' , Θ^* tente de relier s au parent $parent(s')$ de s' via une ligne de visée directe notée $lineOfSight(parent(s'), s)$. Si cette liaison est libre (c'est-à-dire sans obstacle), la mise à jour est donnée par l'équation (3.10). Ce mécanisme réduit le nombre de virages et permet des trajectoires plus fluides.

$$g(s) = \min(g(s), g(parent(s')) + dist(parent(s'), s)) \quad (3.10)$$

3.3.5.1.2. ALGORITHME D* LITE

D* Lite est un algorithme de planification incrémentale [69], conçu pour mettre à jour efficacement un chemin existant lorsqu'une modification locale de l'environnement (obstacle ajouté ou retiré) est détectée. Contrairement à A^* , il ne redémarre pas toute la recherche mais ne met à jour que les nœuds affectés ce qui le rend particulièrement efficace dans des environnements dynamiques. La structure centrale de D* Lite repose sur deux fonctions clés, le coût estimé depuis l'origine $g(s)$ et le coût d'un chemin optimal en tenant compte de changements futurs $rhs(s)$, défini par l'équation (3.11).

$$rhs(s) = \begin{cases} 0, & s = s_{goal}, \\ \min_{s' \in Succ(s)} (g(s') + c(s, s')), & \text{sinon}, \end{cases} \quad (3.11)$$

où $c(s, s')$ est le coût de déplacement de s à s' et $Succ(s)$ désigne l'ensemble des successeurs de s . La priorité d'un nœud dans la file d'attente est déterminée par un vecteur clé à deux composantes déterminé par l'équation (3.12). Ainsi, le nœud ayant la plus petite clé est traité en priorité. Et pour la mise à jour de l'algorithme si $g(s)$ est différent de $rhs(s)$, le nœud est inconsistant et doit être replanifié sinon, il est cohérent. L'algorithme continue jusqu'à ce que tous les nœuds soient cohérents

ou que le chemin soit mis à jour. D* Lite est donc particulièrement adapté pour une replanification en ligne et son intégration dans le système permet une réactivité élevée face aux obstacles dynamiques détectés par la costmap 3D.

$$Key(s) = [\min(g(s), rhs(s)) + h(start, s), \min(g(s), rhs(s))] \quad (3.12)$$

3.3.5.2. PLANIFICATEUR LOCAL

Le planificateur *TEB* (*Timed Elastic Band*) [70] constitue la solution de planification locale retenue dans cette architecture. Son rôle principal est de transformer le chemin global, souvent discret et idéalisé en une trajectoire lissée, réalisable et dynamique tenant compte à la fois des contraintes cinématiques du robot UR5 et des obstacles présents dans l'environnement immédiat. Le principe fondamental de *TEB* repose sur une modélisation de la trajectoire sous forme de bande élastique composée de plusieurs points consécutifs, appelés poses. Ces poses représentent des positions successives dans l'espace tridimensionnel auxquelles sont associées des intervalles de temps modulables. Cette structure flexible permet d'ajuster dynamiquement la forme de la trajectoire en fonction des contraintes détectées comme la proximité d'un obstacle ou une limitation de vitesse. Le planificateur *TEB* agit en minimisant un ensemble de coûts pondérés incluant la distance aux obstacles, la déviation par rapport au chemin global, la régularité temporelle entre les poses et la faisabilité au regard des limites physiques du robot (vitesse, accélération, rayon de courbure). Ce processus d'optimisation est itératif et s'exécute en temps réel garantissant une adaptation rapide à tout changement dans l'environnement.

Dans le contexte spécifique du projet, *TEB* est directement connecté à la costmap dynamique 3D issue du module de perception ce qui lui permet d'intégrer les informations sur les obstacles mobiles ou soudainement apparus. Grâce à cela, le robot UR5 peut ajuster son mouvement au fil de l'exécution tout en maintenant une trajectoire fluide et sécurisée.

En résumé, *TEB* agit comme un optimiseur local, assurant que le chemin global planifié est non seulement suivable par le robot mais également ajusté en permanence aux conditions réelles du terrain. Sa capacité à produire des trajectoires continues, compatibles avec les contraintes du bras robotique en fait un élément central pour garantir une navigation sûre et efficace en environnement industriel collaboratif.

3.4. CONCLUSION

Ce chapitre a présenté l'ensemble des choix méthodologiques ayant permis de concevoir un système robuste et flexible pour la planification de trajectoires dans un environnement collaboratif. En combinant modélisation 3D, cartographie dynamique, planification hybride et exécution articulée, les objectifs initiaux ont été atteints sur le plan fonctionnel et technique. Les fondements posés ici constituent le socle du système expérimental. Le chapitre suivant est consacré à l'évaluation et à la discussion des résultats obtenus lors des tests simulés, mettant en lumière les performances, les limites et les perspectives d'amélioration du système développé.

CHAPITRE 4 RÉSULTATS ET DUSCUSSIONS

4.1. INTRODUCTION

Ce chapitre présente les résultats expérimentaux issus de la mise en œuvre du système de planification de trajectoire développé dans ce projet. L'objectif principal est d'évaluer l'efficacité, la robustesse et la réactivité des algorithmes implémentés dans des environnements simulés complexes, incluant la présence d'obstacles dynamiques et des variations de la position cible.

L'analyse se structure en deux volets. La première partie examine les performances comparées des planificateurs Thêta*, D* Lite ainsi que leur combinaison (D* Lite raffiné) à travers des indicateurs clés tels que le temps de planification, la longueur du chemin, la distance minimale à l'obstacle et le nombre de poses générées. Quatre scénarios distincts sont étudiés afin de mettre en évidence le comportement des algorithmes dans des contextes plus ou moins contraints. La seconde partie explore l'impact de l'intégration du contrôleur local *TEB* (*Timed Elastic Band*) sur les trajectoires produites en évaluant son apport en termes d'optimisation, de fluidité et de sécurité.

Ce chapitre est structuré en deux grandes parties. La première (section 4.2) est consacrée à l'analyse comparative des performances des planificateurs Thêta*, D Lite* et de leur combinaison D Lite raffiné*, selon cinq scénarios types allant de l'absence d'obstacles à des configurations très contraintes (pose derrière obstacle). La seconde (section 4.3) explore l'effet de l'ajout du contrôleur local TEB (*Timed Elastic Band*) sur les trajectoires générées, en mettant en évidence ses apports en termes de fluidité, compacité et maintien de la sécurité.

Les résultats obtenus permettent de vérifier les hypothèses formulées, de cerner les atouts et les limites de chaque composant et d'identifier les conditions dans lesquelles leur complémentarité s'avère la plus bénéfique.

4.2. PERFORMANCES DES PLANIFICATEURS

Afin d'évaluer la pertinence des algorithmes de planification mis en œuvre, une série de tests a été conduite dans un environnement simulé intégrant une costmap dynamique 3D illustrée à la figure 8. L'objectif de cette analyse est de comparer les performances individuelles de Thêta*, de D* Lite ainsi que de leur combinaison (notée D* Lite raffiné), en fonction de la position de la cible par rapport aux obstacles (un exemple de trajectoire de ces algorithmes est présenté à la figure 8).

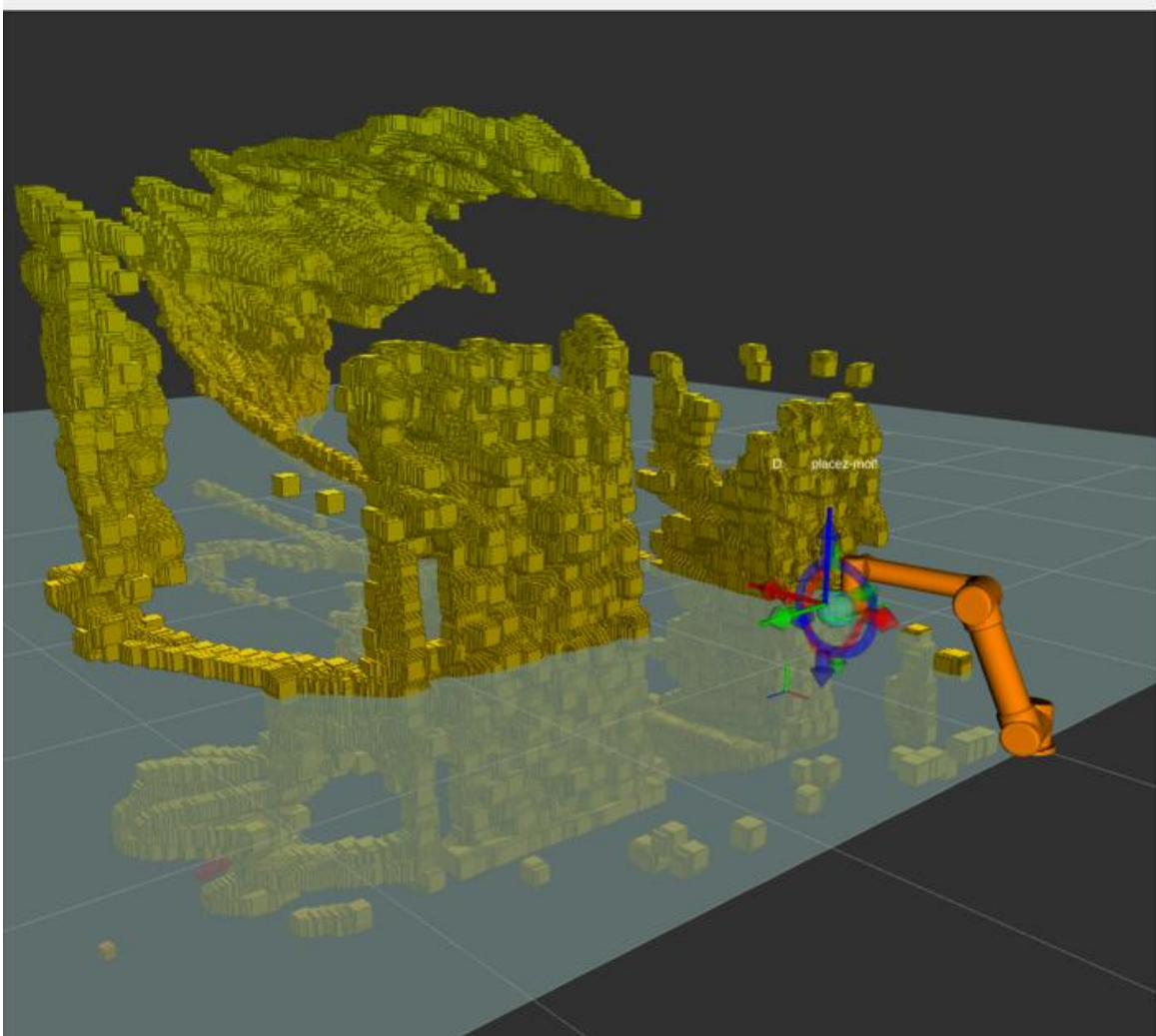


Figure 8 : Illustration du costmap 3D dynamique

L'évaluation s'appuie sur quatre indicateurs quantitatifs représentatifs à savoir le temps de planification, la longueur totale de la trajectoire, la distance minimale à l'obstacle et le nombre de poses générées. Ces paramètres permettent de caractériser à la fois l'efficacité, la sécurité et la complexité des trajectoires produites.

Pour une analyse fine, les tests ont été organisés selon cinq configurations types de placement de la cible avec une complexité progressive. Ainsi, le placement de la cible est fait :

- sans obstacle direct ;
- éloignée des obstacles ;
- proche d'un obstacle ;
- très proche ;
- presque derrière un obstacle.

Chaque configuration expérimentale permet de révéler les atouts et les limites spécifiques des différentes approches de planification, notamment en ce qui concerne leur robustesse et leur capacité d'adaptation face à des environnements dynamiques. Le banc d'essai utilisé pour ces évaluations est illustré à la figure 10. Dans ce scénario, une barre cylindrique mobile est intégrée afin de simuler la présence d'un bras humain en mouvement, introduisant ainsi une contrainte dynamique représentative des interactions potentielles en contexte collaboratif. Les résultats de ces différents tests sont présentés dans les sections suivantes.

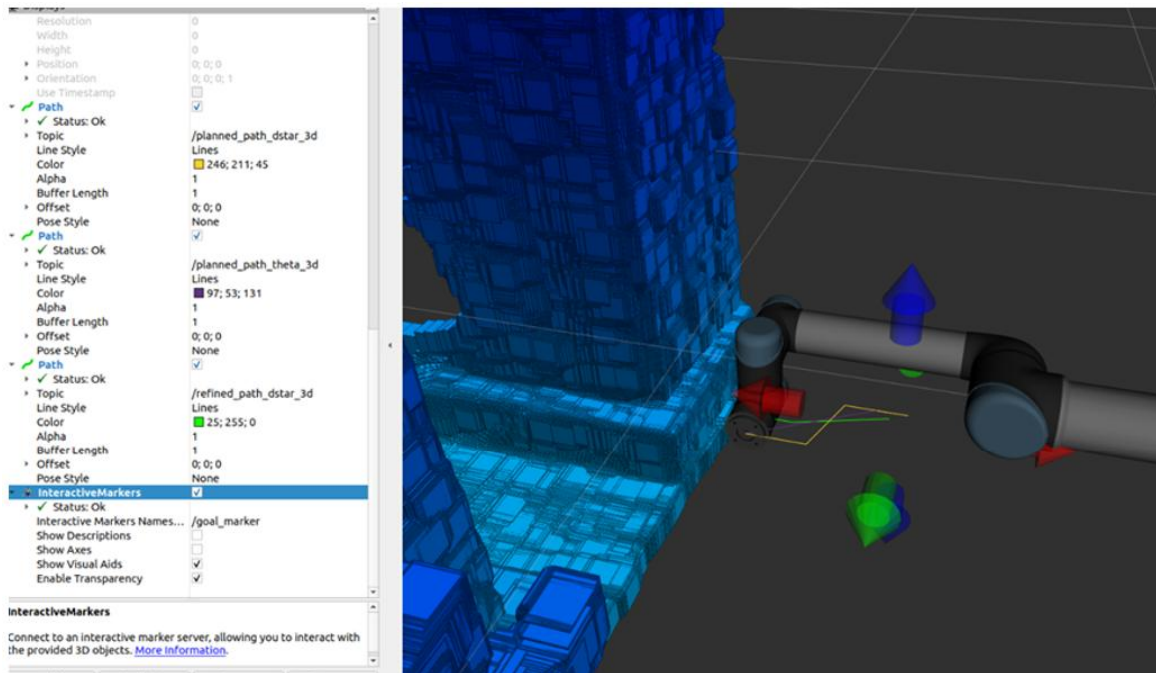


Figure 9 : Exemple de trajectoire pour les trois algorithmes



Figure 10 : Banc d'essai utilisé pour les tests

4.2.1.POSE SANS PRÉSENCE D'OBSTACLES

Dans des conditions idéales (figure 11), où aucun obstacle n'est présent dans l'environnement (statique ou dynamique), les trois planificateurs évalués Thêta*, D* Lite et D* Lite raffiné présentent des comportements différenciés. Comme attendu, Thêta* affiche les performances les plus stables avec un temps de planification court et constant, ce qui confirme son efficacité dans des environnements non contraints. À l'inverse, D* Lite subit une hausse drastique de son temps de calcul pour certaines poses (Goal_ID 2 et 3), avec un pic dépassant les 25 secondes, ce qui témoigne d'une surcharge algorithmique injustifiée en l'absence de complexité environnementale. La version raffinée (Thêta* & D* Lite) parvient à maintenir un compromis, limitant ces pics tout en restant plus rapide que D* Lite seul.

En ce qui concerne la longueur des trajectoires et le nombre de poses, Thêta* génère les chemins les plus directs et compacts. D* Lite, en revanche, allonge inutilement les trajets même sans obstacle, et multiplie les poses intermédiaires. D* Lite raffiné montre une performance intermédiaire, proche de Thêta*, mais légèrement plus coûteuse en poses.

Enfin, la distance minimale à l'obstacle est inexistante ici puisque l'environnement est totalement libre, confirmant que les planificateurs n'ont pas à gérer d'évitement. Cette configuration met donc en lumière l'intérêt d'utiliser Thêta* comme base rapide et efficace dans les environnements peu ou pas encombrés.

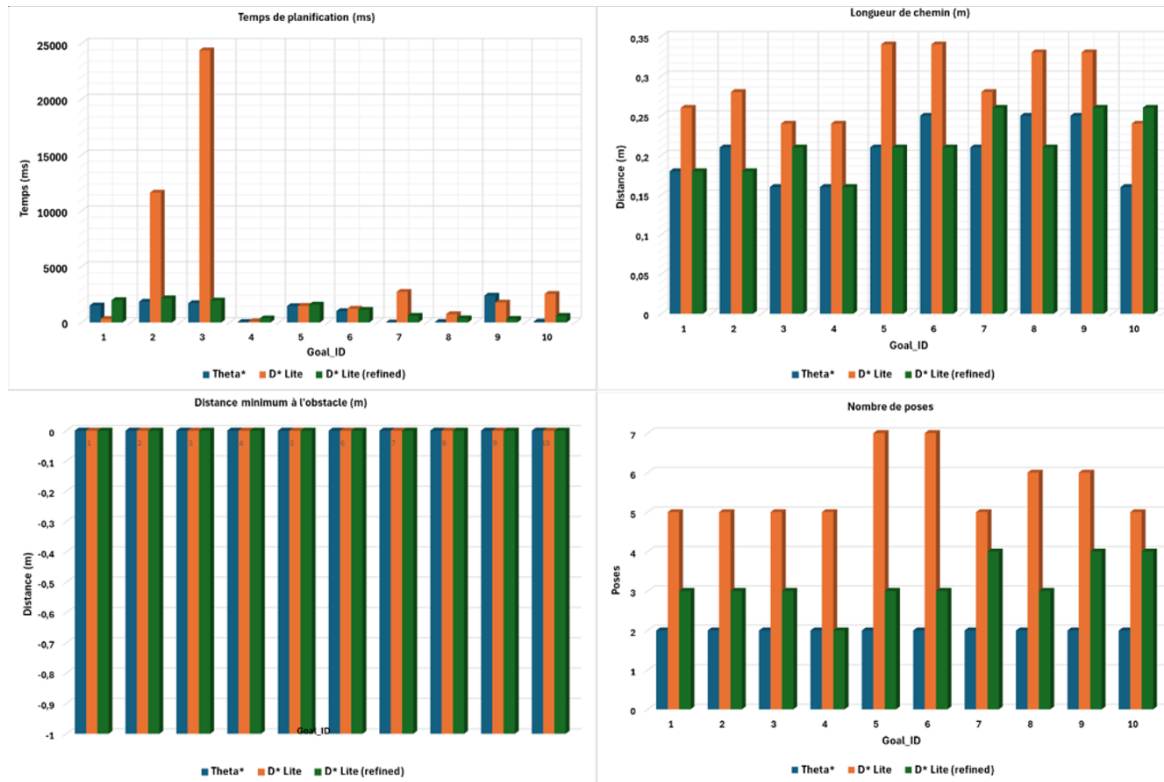


Figure 11 : Performances des algorithmes pour une pose sans obstacles

Après avoir examiné le comportement des algorithmes dans un environnement entièrement dégagé, il est désormais pertinent d'évaluer leurs performances lorsque la pose cible se situe dans une zone où des obstacles, bien que distants, sont présents. Cette configuration, plus représentative d'un environnement industriel réel, permet d'analyser la capacité des planificateurs à anticiper les contraintes spatiales tout en maintenant des trajectoires efficaces.

4.2.2.POSE ELOIGNEE DES OBSTACLES

Lorsque l'objectif est situé dans une zone dégagée, éloignée de toute contrainte spatiale significative, les différences entre les algorithmes se traduisent essentiellement par des variations en termes de performance et d'efficacité. La figure 12 illustre ces résultats à travers quatre indicateurs, temps de planification, distance minimale à l'obstacle, longueur de chemin et nombre de poses.

Sur le plan du temps de planification, l'algorithme Thêta* (en bleu) se distingue par sa rapidité constante, affichant des temps inférieurs à 300 ms pour la majorité des objectifs. En comparaison, D* Lite (orange) présente des durées de calcul significativement plus longues en général, allant jusqu'à 2200 ms pour certains objectifs (Goal_ID 17). La version combinée Thêta* & D* Lite (vert) montre un compromis intéressant, limitant le surcoût computationnel tout en restant performant.

Concernant la distance minimale à l'obstacle, les trois méthodes maintiennent un seuil de sécurité relativement similaire, proche de 0.15 à 0.17 m. Cela confirme que dans des contextes dégagés, la sécurité n'est pas un facteur discriminant entre les planificateurs. On note toutefois une légère amélioration pour Thêta* & D* Lite sur certains objectifs (Goal_ID 24, 25), suggérant un comportement plus prudent.

La longueur du chemin révèle une performance nettement supérieure de Thêta* avec des trajectoires plus courtes que celles produites par D* Lite. Ce dernier tend à générer des trajectoires plus longues, en raison de sa nature incrémentale et prudente. L'approche hybride Thêta* & D* Lite reste proche de Thêta* ce qui indique que dans des zones peu contraignantes, le raffinement apporté par D* Lite est peu sollicité.

Enfin, pour le nombre de poses générées, les tendances sont similaires, Thêta* génère systématiquement deux poses environ tandis que D* Lite en produit jusqu'à cinq. L'approche hybride conserve un nombre limité de poses, démontrant sa capacité à simplifier l'exécution sans altérer la précision.

En résumé, pour des poses éloignées des obstacles, l'approche combinée Thêta* & D* Lite hérite des avantages des deux méthodes, rapidité, concision des trajectoires, tout en conservant un bon niveau de sécurité. Elle constitue ici le choix optimal, confirmant l'hypothèse selon laquelle une planification globale hybride améliore les performances dans un environnement dégagé.

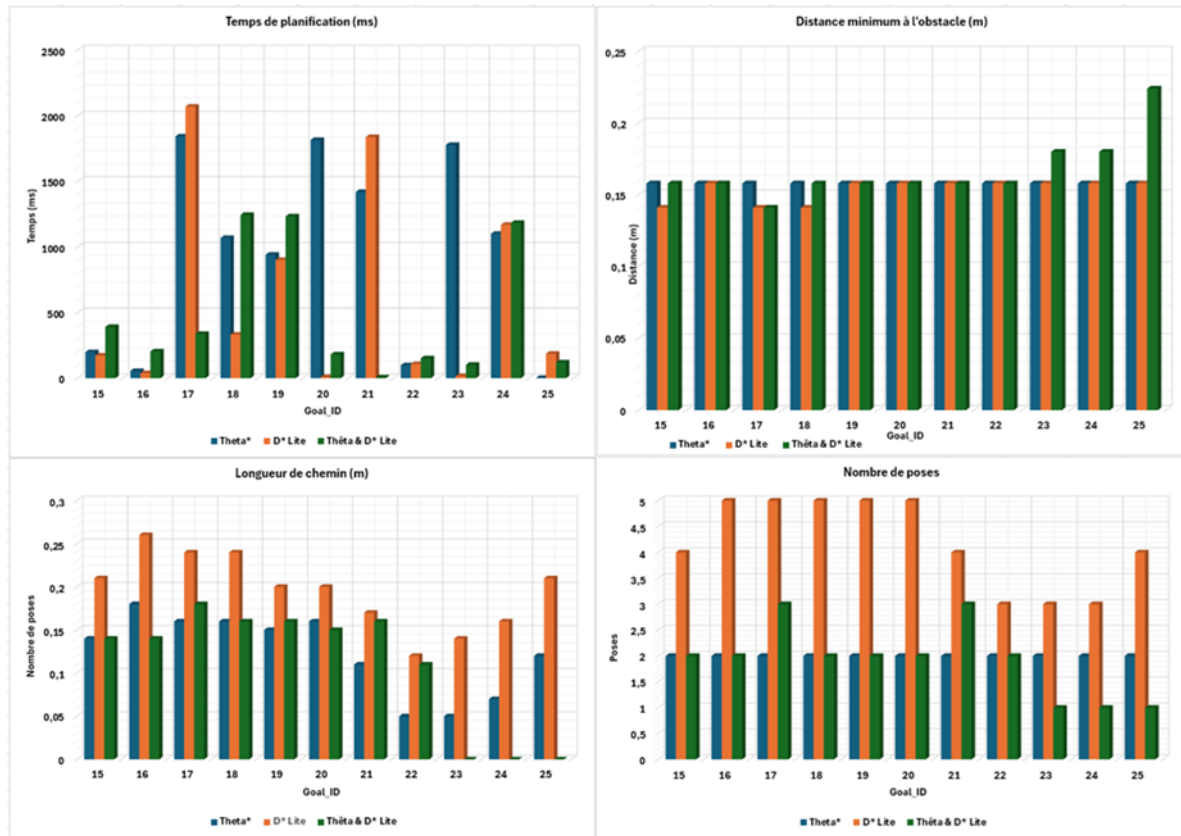


Figure 12 : Performances des algorithmes pour une pose éloignée des obstacles

4.2.3.POSE PROCHE D'UN OBSTACLE

Lorsque la cible à atteindre est située à proximité immédiate d'un obstacle, les différences entre les algorithmes deviennent plus marquées, notamment en termes de temps de calcul, de complexité de la trajectoire et de sécurité. La figure 13 présente les résultats pour les Goal_IDs de 1 à 11 à travers quatre indicateurs, temps de planification, distance minimale à l'obstacle, longueur de chemin et nombre de poses.

Du point de vue du temps de planification, Thêta* (en bleu) reste globalement rapide, même dans un contexte plus contraint. Toutefois, certains objectifs (ex. Goal_IDs 2) montrent un temps de planification plus élevé atteignant jusqu'à 1800 ms, ce qui suggère une sensibilité accrue à la densité locale. En revanche, D* Lite (orange) devient sensiblement plus lent, dépassant souvent les 2000

ms. La version combinée Thêta* & D* Lite (vert) montre un comportement intermédiaire, offrant un bon compromis entre réactivité et raffinement.

La distance minimale à l'obstacle constitue un critère critique dans cette configuration. On observe ici que Thêta* et D* Lite maintiennent une distance de sécurité relativement stable, aux alentours de 0.155 m. En revanche, Thêta* & D* Lite adopte un comportement plus conservateur, bien que sa distance soit légèrement réduite (~ 0.14 m), elle reflète une volonté d'approcher au plus près de l'obstacle, ce qui pourrait être problématique dans certains contextes industriels.

Concernant la longueur du chemin, D* Lite produit systématiquement les trajectoires les plus longues. Thêta* génère des chemins plus courts, mais parfois au prix d'un manque de prudence. L'approche hybride (Thêta* & D* Lite) montre des résultats cohérents et optimisés, équilibrant longueur réduite et sécurité.

Enfin, en termes de nombre de poses, D* Lite affiche des valeurs élevées, autour de cinq poses, traduisant une trajectoire segmentée et prudente. Thêta* reste autour de deux poses, confirmant sa capacité à générer des chemins plus directs. Thêta* & D* Lite conserve ici un nombre raisonnable de poses (en moyenne 2 à 3), ce qui facilite l'exécution tout en assurant une couverture sécurisée.

Dans l'ensemble, cette configuration démontre que la combinaison Thêta* & D* Lite offre un compromis intéressant, elle conserve une bonne efficacité de planification tout en adaptant la trajectoire à la présence d'obstacles proches. Cela confirme son intérêt pour les environnements partiellement contraints, tout en soulignant la nécessité d'un ajustement local pour préserver les marges de sécurité.

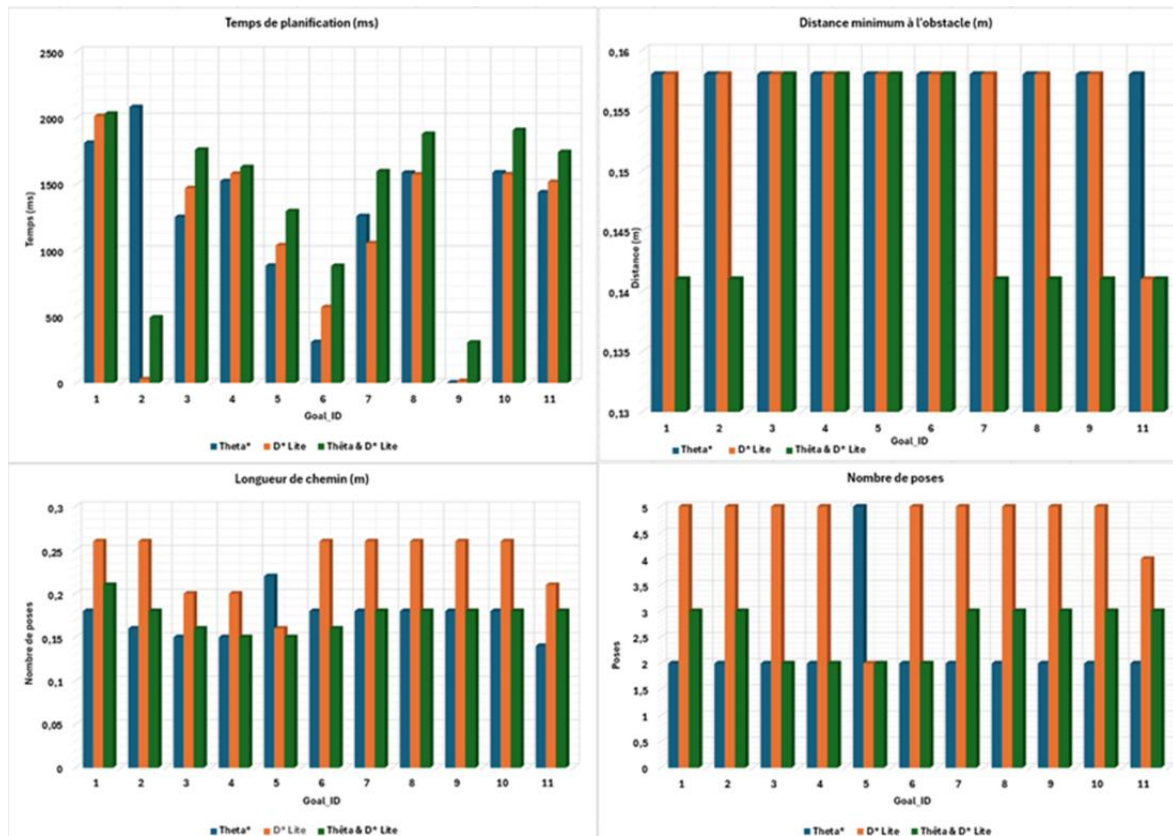


Figure 13 : Performances des algorithmes pour une pose proche d'un obstacle

4.2.4. POSE TRÈS PROCHE DE L'OBSTACLE

Lorsque la cible se situe dans une zone très proche d'un obstacle, la planification devient significativement plus exigeante. Cette situation accentue les écarts de performances entre les différents algorithmes testés, tant au niveau du temps de calcul, de la qualité de la trajectoire que de la sécurité. Les résultats sont présentés à la figure 14 pour les Goal_IDs de 27 à 36 à travers ces quatre indicateurs.

En ce qui concerne le temps de planification, D* Lite (orange) affiche des durées de calcul extrêmement élevées, atteignant jusqu'à 6000 ms (Goal_ID 28) dans certains cas. Cela souligne sa difficulté à gérer les environnements densément encombrés, surtout lorsqu'un grand nombre de replanifications est nécessaire. À l'inverse, Thêta* (bleu) et la combinaison Thêta* & D* Lite (vert)

restent plus stables avec un temps de planification moyen autour de 1200 à 1500 ms, ce qui les rend nettement plus adaptés aux environnements contraints.

La distance minimale à l'obstacle reste globalement constante pour Thêta* et D* Lite, autour de 0.15 m. En revanche, l'approche combinée montre un pic remarquable pour le Goal_ID 27, atteignant 0.22 m, ce qui traduit un comportement plus prudent du planificateur hybride dans certaines configurations critiques. Cela démontre sa capacité à maximiser les marges de sécurité lorsque l'environnement le permet.

En termes de longueur de chemin, D* Lite produit encore une fois les trajectoires les plus longues. Thêta* reste le plus efficace tandis que Thêta* & D* Lite conserve un compromis équilibré entre concision et contournement des obstacles. Cette tendance confirme que l'algorithme hybride ajuste localement la trajectoire lorsque nécessaire sans générer des excès de distance.

Enfin, le nombre de poses générées est un excellent indicateur de la complexité du chemin. D* Lite affiche systématiquement 5 poses, ce qui traduit une segmentation excessive, difficile à exécuter sur un robot réel. À l'opposé, Thêta* reste proche de 2 poses. Le planificateur hybride se situe entre les deux avec 3 à 4 poses, ce qui assure un bon compromis entre précision du chemin et fluidité d'exécution.

En résumé, dans un contexte de planification très contrainte, la combinaison Thêta* & D* Lite conserve sa capacité d'adaptation tout en limitant le coût temporel et en maintenant un niveau de sécurité satisfaisant. Elle se positionne ainsi comme la meilleure option pour des environnements où l'espace libre est réduit, et où les exigences de précision et de robustesse sont élevées.

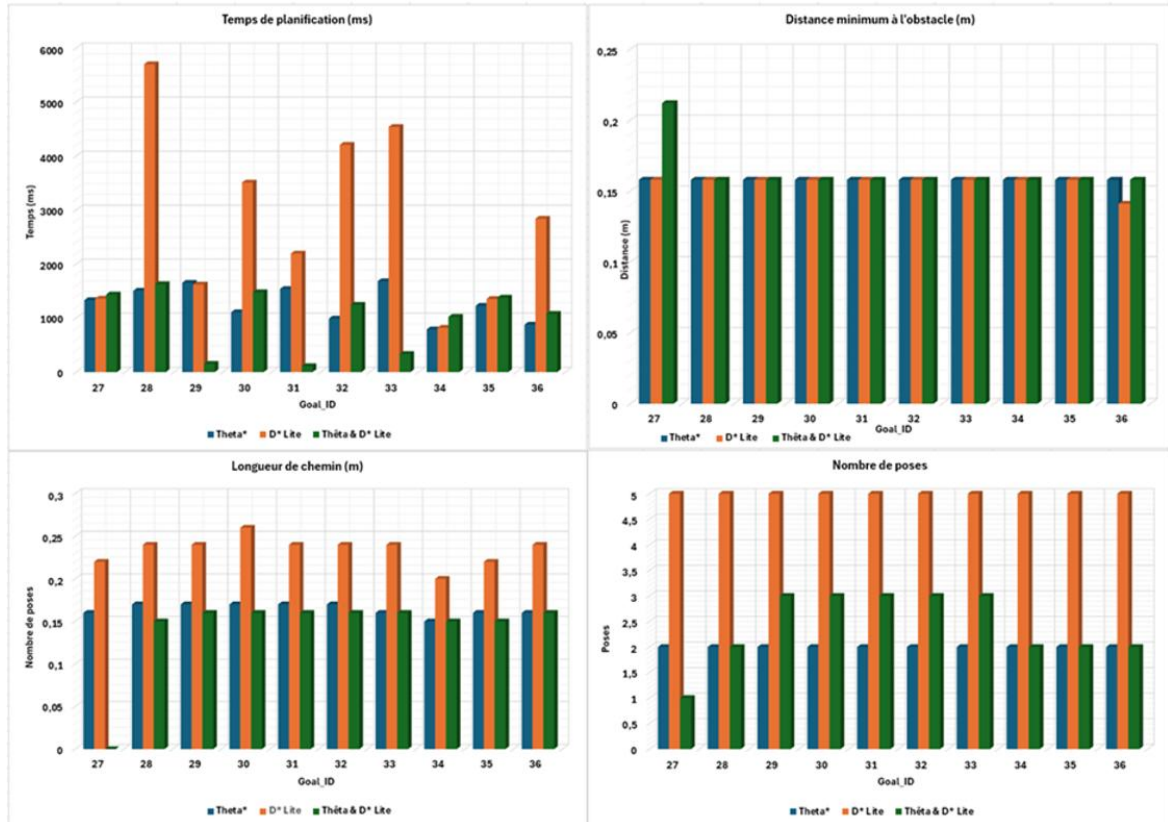


Figure 14 : Performances des algorithmes pour une pose très proche d'un obstacle

4.2.5.POSE PRESQUE DERRIÈRE UN OBSTACLE

Ce scénario représente un cas extrême, où la cible est placée dans une position quasi-occulte, nécessitant un contournement complet de l'obstacle. Cette situation teste la capacité des planificateurs à explorer efficacement l'espace de recherche tout en respectant les contraintes de sécurité. Les résultats de ces tests sont présentés à la figure 15 pour les Goal_IDs de 38 à 47 à travers les quatre mêmes indicateurs. Les résultats obtenus révèlent que D* Lite échoue systématiquement à générer un chemin valide dans cette situation, ne parvenant pas à surmonter la contrainte d'occultation causée par la proximité immédiate de l'obstacle. Ce comportement met en lumière une limitation importante de l'algorithme dans des environnements fortement contraints où l'accessibilité visuelle et topologique est restreinte.

Ainsi, concernant le temps de planification, l'approche Thêta* & D* Lite (en vert) montre des temps particulièrement élevés pour certains objectifs, atteignant près de 12 secondes pour le Goal_ID 45 et plus de 8 secondes pour le Goal_ID 38. Cela indique que la complexité du raffinement apporté par D* Lite devient coûteuse en cas de chemins très contraints. Thêta* seul (en bleu), bien qu'un peu plus lent que dans les cas précédents, reste beaucoup plus stable (autour de 1 à 2 secondes) et montre une meilleure réactivité dans ce type de scénario.

La distance minimale à l'obstacle révèle une nette différence, Thêta* parvient à maintenir une distance constante et sécuritaire d'environ 0.16 m pour tous les objectifs, tandis que la version raffinée réduit dangereusement cette distance pour plusieurs Goal_IDs (notamment 38,43, 44 et 45, descendant en dessous de 0.05 m). Cela montre que le raffinement par D* Lite bien qu'utile pour l'optimisation locale peut parfois compromettre la marge de sécurité.

La longueur du chemin est également significativement augmentée par l'approche hybride pour les Goal_IDs les plus contraints. Dans certains cas (comme Goal_IDs 40 à 44), la trajectoire générée dépasse 0.35 m, tandis que Thêta* maintient une longueur plus courte et plus directe. Cette inflation du chemin est souvent la conséquence de contournements excessifs ou de chemins replanifiés trop finement.

Enfin, le nombre de poses montre un contraste net, Thêta* génère systématiquement 2 à 3 poses, alors que la version combinée peut monter jusqu'à 6 poses, doublant ainsi la segmentation du chemin. Ce comportement, bien qu'indicatif d'un raffinement plus précis, complique considérablement la commande du bras robotique et augmente la durée d'exécution.

En résumé, dans les cas où la cible est presque complètement masquée par un obstacle, Thêta* se montre plus rapide, plus fluide et plus sécuritaire au prix parfois d'une trajectoire moins optimisée localement. À l'inverse, *Thêta** & D* Lite est capable de trouver des chemins plus fins et adaptatifs, mais au prix d'un coût temporel élevé, d'une proximité plus risquée avec les obstacles et d'un nombre élevé de poses, ce qui peut nuire à la fluidité de l'exécution.

Ce scénario met en lumière les limites du raffinement local, notamment lorsque les marges de manœuvre sont très faibles. Il souligne également l'intérêt d'une régulation contextuelle du niveau de

raffinement pour éviter une surcharge computationnelle inutile et préserver la sécurité. Cette situation souligne également à la fois la robustesse de Thêta* en environnement très contraint et les limites de D* Lite en l'absence de visibilité ou de passage direct. La solution hybride, bien qu'utile, doit être utilisée avec prudence dans de telles configurations, au risque d'un coût computationnel élevé et d'une trajectoire complexe difficilement exécutable.

Afin de compléter cette analyse comparative des planificateurs globaux, il est désormais pertinent d'examiner l'impact du module de planification locale *TEB* (*Timed Elastic Band*) sur la qualité et la fluidité des trajectoires générées, notamment dans leur exécution par le bras robotique UR5.

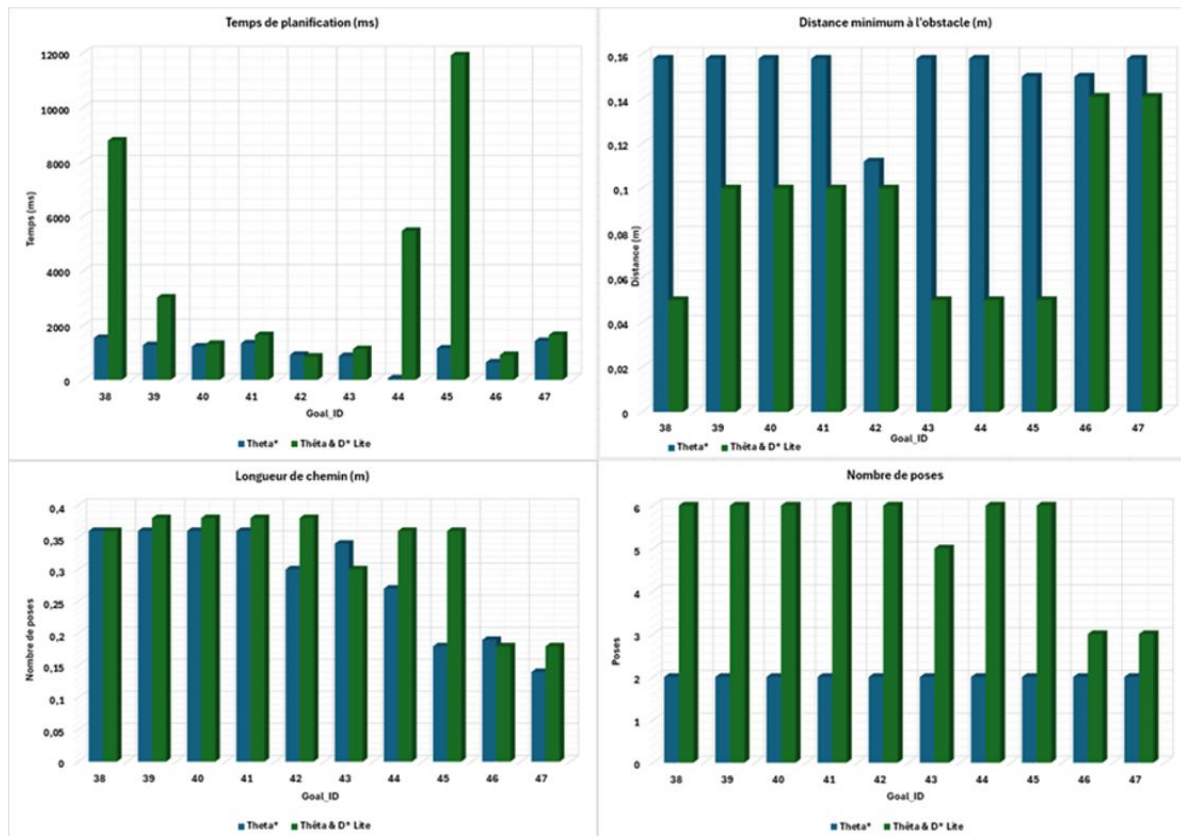


Figure 15 : Performances des algorithmes pour une pose presque derrière un obstacle

4.3. INFLUENCE DU TEB SUR L'ALGORITHME D* LITE RAFFINÉ

L'intégration du planificateur local *TEB (Timed Elastic Band)* au sein du pipeline de planification basé sur *D* Lite raffiné* a été évaluée afin d'examiner son impact sur la qualité des trajectoires générées dans un environnement industriel dynamique. Cette étude a porté sur une série de 20 objectifs, répartis dans différents contextes de complexité, et a utilisé quatre indicateurs clés, le temps de planification, la longueur du chemin, le nombre de poses intermédiaires et la distance minimale à l'obstacle.

Les résultats obtenus (figure 16) montrent que l'ajout du *TEB* n'induit qu'une légère surcharge en temps de planification dans certaines situations complexes. Par exemple, pour l'objectif 11, on observe un pic à plus de 4000 ms avec *TEB*, contre environ 1500 ms pour *D* Lite raffiné* seul. Toutefois, dans la majorité des cas, cette augmentation reste négligeable, démontrant que le module *TEB* s'intègre efficacement sans altérer la réactivité globale du système.

En ce qui concerne la longueur du chemin, l'optimisation locale assurée par *TEB* permet dans presque tous les cas une réduction mesurable de la distance parcourue. Cette amélioration, bien que parfois marginale, se révèle constante, en particulier dans les zones à forte densité d'obstacles, confirmant l'efficacité de *TEB* pour produire des trajectoires plus courtes et plus efficaces.

Le nombre de poses générées est également réduit dans plusieurs configurations, traduisant une simplification du plan d'exécution. Cette diminution du nombre de points de contrôle facilite la commande du bras UR5, améliore la fluidité du mouvement et réduit le risque d'oscillations inutiles.

Enfin, sur le critère de la sécurité, les résultats indiquent que *TEB* conserve rigoureusement les marges prévues par le planificateur initial. Les distances minimales aux obstacles restent stables et conformes aux contraintes imposées garantissant ainsi un respect strict des limites de sécurité dans l'environnement collaboratif.

Dans l'ensemble, cette évaluation met en évidence l'apport significatif du module *TEB* en termes de performance sans compromettre ni la sécurité ni la réactivité du système. Il constitue ainsi une

extension pertinente pour renforcer la qualité d'exécution des trajectoires planifiées par D* Lite raffiné, notamment dans des scénarios industriels où la précision et l'adaptabilité sont primordiales.

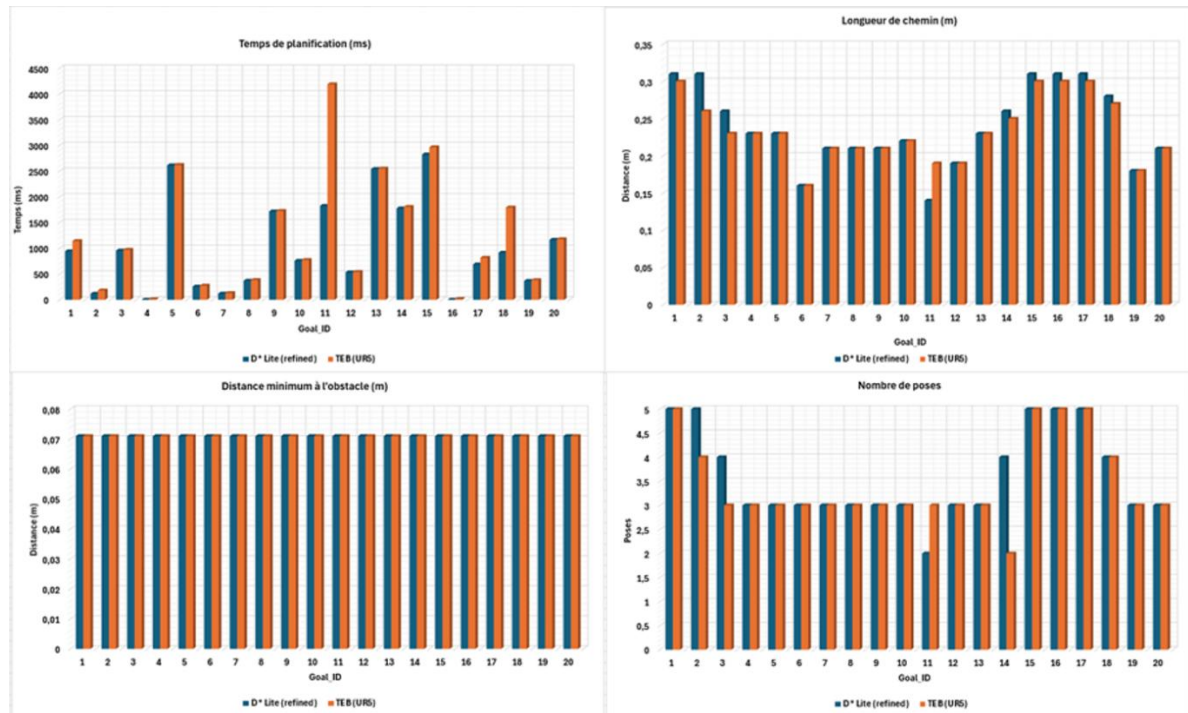


Figure 16 : Influence du TEB sur l'algorithme D* Lite raffiné

4.4. CONCLUSION

L'analyse approfondie des performances des différents planificateurs présentés dans ce chapitre a permis de mettre en lumière les forces et les limites des algorithmes Thêta*, D* Lite, ainsi que de leur combinaison optimisée (D* Lite raffiné) en contexte de planification globale. Dans les environnements simples, sans obstacle, Thêta* démontre des performances rapides et stables, avec des trajectoires courtes et peu de poses. Toutefois, dès que la complexité de l'environnement augmente, avec des obstacles statiques ou dynamiques proches de la cible, D* Lite montre des signes de fragilité notamment un temps de planification fortement pénalisé voire une incapacité à produire un chemin valide dans certains cas extrêmes.

La stratégie hybride basée sur la combinaison de Thêta* et de D* Lite offre une réponse pertinente à ces limites. Elle combine la rapidité initiale du Thêta* pour une planification directe et fluide avec la

capacité du D* Lite à effectuer une replanification locale incrémentale en s'adaptant aux variations de l'environnement. Ce mécanisme de raffinement progressif permet d'assurer une plus grande robustesse dans les environnements dynamiques et encombrés, en maintenant des temps de planification raisonnables et des distances sécuritaires vis-à-vis des obstacles.

L'intégration du planificateur local TEB (Timed Elastic Band) sur les trajectoires issues du D* Lite raffiné constitue une optimisation supplémentaire. Les résultats montrent que TEB contribue à raccourcir la longueur des trajectoires, à réduire le nombre de poses intermédiaires, tout en maintenant une distance minimale constante par rapport aux obstacles. Ce lissage local est obtenu sans surcharge computationnelle excessive, ce qui confirme la pertinence de son intégration dans des architectures robotiques temps réel.

Les résultats obtenus dans ce chapitre confirment ainsi les hypothèses de recherche formulées en amont, la combinaison d'une costmap dynamique 3D avec une stratégie de planification hybride Thêta* et D* Lite, raffinée localement par TEB, permet d'atteindre des performances élevées en termes de réactivité, de sécurité et d'optimalité des trajectoires. Cette approche s'adapte efficacement aux contraintes spécifiques des environnements industriels collaboratifs, notamment dans les situations de cohabitation homme-robot.

CHAPITRE 5

CONCLUSION

Ce mémoire a présenté une approche méthodologique intégrée pour la planification et l'exécution de trajectoires de bras robotiques dans des environnements industriels collaboratifs, dynamiques et partagés avec des opérateurs humains. L'objectif principal était de concevoir une architecture logicielle capable de générer des trajectoires sûres, réactives et optimisées, en prenant en compte à la fois la variabilité de l'environnement et les contraintes cinématiques du robot UR5.

La solution développée repose sur la synergie entre plusieurs technologies complémentaires : la perception tridimensionnelle par caméra stéréo Intel RealSense D455, la cartographie probabiliste à l'aide de la bibliothèque OctoMap, la conversion en costmap dynamique 3D via un nœud ROS2 dédié, et la planification multi-niveau. Plus précisément, la combinaison des algorithmes Thêta* et D* Lite permet une planification globale fluide et adaptative, tandis que l'algorithme TEB (Timed Elastic Band) assure une optimisation locale tenant compte des contraintes cinématiques du robot. En complément, un module IETA simplifié (Time To Collision) assure une évaluation continue du risque de collision avec les obstacles dynamiques, renforçant la sécurité dans les scénarios de collaboration homme-robot.

Les résultats expérimentaux en environnement simulé ont permis de valider les hypothèses de recherche. Ils ont mis en évidence :

- la complémentarité efficace entre Thêta* et D* Lite dans des situations variées ;
- la contribution de TEB à l'optimisation locale des trajectoires ;
- la pertinence du module IETA simplifié (TTC) pour la prévention proactive des collisions.

L'architecture proposée s'est révélée modulaire, extensible et bien adaptée à l'écosystème ROS2, répondant aux exigences de réactivité, de robustesse et de sécurité propres à la robotique collaborative moderne. Elle constitue ainsi une contribution méthodologique significative pour le pilotage de bras robotiques dans des environnements dynamiques.

Cependant, certaines limites ont été identifiées, notamment :

- des difficultés de planification dans les situations fortement contraintes ou avec des poses très proches d'obstacles ;
- un surcoût computationnel dans les cas de raffinement excessif du chemin.

Ces constats ouvrent plusieurs perspectives concrètes d'amélioration :

- Renforcement du module IETA : vers une version avancée intégrant des modèles prédictifs (apprentissage supervisé, modèles probabilistes) pour une anticipation plus fine des interactions humaines ;
- Amélioration de la planification locale : exploration de contrôleurs plus adaptatifs, tels que *MPPI (Model Predictive Path Integral)*, pouvant compléter ou remplacer TEB dans des scénarios à forte contrainte ;
- Planification multi-objectifs : développement de fonctions de coût pondérées intégrant simultanément les critères de sécurité, de rapidité d'exécution et de consommation énergétique ;
- Validation expérimentale en conditions réelles : déploiement du système sur un robot physique dans un environnement industriel réel, avec objets manipulables, opérateurs humains multiples et perturbations imprévues.

En résumé, ce travail pose les fondations d'une architecture robuste, évolutive et orientée vers la sécurité, alignée avec les principes de l'industrie 5.0. Il contribue à faire avancer la robotique collaborative dans la direction d'une meilleure interaction homme-robot, tout en garantissant performance et fiabilité dans des environnements complexes et évolutifs.

LISTE DE RÉFÉRENCES

- [1] M. Calzavara, M. Faccio, et I. Granata, "Multi-objective task allocation for collaborative robot systems with an Industry 5.0 human-centered perspective," *International Journal of Advanced Manufacturing Technology*, vol. 128, no. 1-2, pp. 297-314, 2023. doi: 10.1007/s00170-023-11673-x.
- [2] L. Tomidei, M. Guertler, N. Sick, G. Paul, et M. Carmichael, "Design Principles for Safe Human Robot Collaboration," *Interaction Design and Architecture(s)*, no. 61, pp. 66-97, 2024. doi: 10.55612/s-5002-061-002.
- [3] J. Xiao, J. Gao, N. Anwer, et B. Eynard, "Multi-Agent Reinforcement Learning Method for Disassembly Sequential Task Optimization Based on Human-Robot Collaborative Disassembly in Electric Vehicle Battery Recycling," *Journal of Manufacturing Science and Engineering*, vol. 145, no. 12, Article 121001, 2023. doi: 10.1115/1.4062235.
- [4] K. Niquet, N. Patel, F. Schrodell, M. Jahn, et S. Varelmann, "Prototype of an intelligent textile based robot skin to expand potential applications for cobot's," Dans *2023 8th International Conference on Mechanical Engineering and Robotics Research, ICMERR 2023*, 2023, pp. 12-17.
- [5] W. Li, Y. Peng, Y. Zhu, D. T. Pham, A. Y. C. Nee, et S. K. Ong, "End-of-life electric vehicle battery disassembly enabled by intelligent and human-robot collaboration technologies: A review," *Robotics and Computer-Integrated Manufacturing*, vol. 89, Article 102758, 2024. doi: 10.1016/j.rcim.2024.102758.
- [6] S. Keivanpour, "Integrating Lean Management Principles into Human-Robot Collaboration in Disassembly Cell," Dans *Manufacturing Driving Circular Economy*, Cham, 2023, pp. 339-348: Springer International Publishing.
- [7] D. Kamath, S. Moore, R. Arsenault, et A. Anctil, "A system dynamics model for end-of-life management of electric vehicle batteries in the US: Comparing the cost, carbon, and material requirements of remanufacturing and recycling," *Resources, Conservation and Recycling*, vol. 196, Article 107061, 2023. doi: 10.1016/j.resconrec.2023.107061.
- [8] J. Hathaway, A. Shaarawy, C. Akdeniz, A. Aflakian, R. Stolkin, et A. Rastegarpanah, "Towards reuse and recycling of lithium-ion batteries: tele-robotics for disassembly of electric vehicle batteries," *Frontiers in Robotics and AI*, vol. 10, Article 1179296, 2023. doi: 10.3389/frobt.2023.1179296.
- [9] S. Mateo, "A procedure for conduction of a successful literature review using the PRISMA method," *Kinesitherapie*, vol. 20, no. 226, pp. 29-37, 2020. doi: 10.1016/j.kine.2020.05.019.
- [10] C. Zammit et E. J. Van Kampen, "Comparison Between A* and RRT Algorithms for 3D UAV Path Planning," *Unmanned Systems*, vol. 10, no. 2, pp. 129-146, 2022. doi: 10.1142/S2301385022500078.
- [11] C. Van Dang, H. Ahn, D. S. Lee, et S. C. Lee, "A Path Planning Method Based on Theta-star Search for Non-Holonomic Robots," Dans *2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems, SCIS and ISIS 2022*, 2022.
- [12] S. R. Bassolillo, G. Raspaolo, L. Blasi, E. D'Amato, et I. Notaro, "Path Planning for Fixed-Wing Unmanned Aerial Vehicles: An Integrated Approach with Theta* and Clothoids," *Drones*, vol. 8, no. 2, Article 62, 2024. doi: 10.3390/drones8020062.

- [13] W. Zhao, L. Li, Y. Wang, H. Zhan, Y. Fu, et Y. Song, "Research on A Global Path-Planning Algorithm for Unmanned Arial Vehicle Swarm in Three-Dimensional Space Based on Theta*-Artificial Potential Field Method," *Drones*, vol. 8, no. 4, Article 125, 2024. doi: 10.3390/drones8040125.
- [14] P. Muñoz et M. D. R-Moreno, "S-Theta*: Low steering path-planning algorithm," Dans *Res. and Dev. in Intelligent Syst. XXIX: Incorporating Applications and Innovations in Intel. Sys. XX - AI 2012, 32nd SGAI Int. Conf. on Innovative Techniques and Applications of Artificial Intel.*, 2012, pp. 109-121.
- [15] P. Muñoz, M. D. R-Moreno, et B. Castaño, "3Dana: A path planning algorithm for surface robotics," *Engineering Applications of Artificial Intelligence*, vol. 60, pp. 175-192, 2017. doi: 10.1016/j.engappai.2017.02.010.
- [16] Z. Chen, Y. Zhang, Y. Zhang, Y. Nie, J. Tang, et S. Zhu, "A Hybrid Path Planning Algorithm for Unmanned Surface Vehicles in Complex Environment with Dynamic Obstacles," *IEEE Access*, vol. 7, pp. 126439-126449, Article 8808914, 2019. doi: 10.1109/ACCESS.2019.2936689.
- [17] Y. Zhang, Z. Xiao, X. Yuan, S. Li, et S. Liang, "Obstacle Avoidance of Two-Wheeled Mobile Robot based on DWA Algorithm," Dans *Proceedings - 2019 Chinese Automation Congress, CAC 2019*, 2019, pp. 5701-5706.
- [18] L. Wan, J. Li, et M. Zhou, "A hybrid path planning strategy unified with COLREGS for the unmanned surface vehicle," *Ships and Offshore Structures*, 2025. doi: 10.1080/17445302.2025.2486203.
- [19] Z. Cui et Y. Wang, "UAV Path Planning Based on Multi-Layer Reinforcement Learning Technique," *IEEE Access*, vol. 9, pp. 59486-59497, Article 9406006, 2021. doi: 10.1109/ACCESS.2021.3073704.
- [20] G. M. Nayeem, M. Fan, et G. M. Daiyan, "Adaptive Q-Learning Grey Wolf Optimizer for UAV Path Planning," *Drones*, vol. 9, no. 4, Article 246, 2025. doi: 10.3390/drones9040246.
- [21] Y. S. Yang et J. G. Juang, "Path Planning and Obstacle Avoidance of Formation Flight," *Sensors*, vol. 25, no. 8, Article 2447, 2025. doi: 10.3390/s25082447.
- [22] M. S. Khojasteh et A. Salimi-Badr, "Autonomous Quadrotor Path Planning Through Deep Reinforcement Learning With Monocular Depth Estimation," *IEEE Open Journal of Vehicular Technology*, vol. 6, pp. 34-51, 2025. doi: 10.1109/OJVT.2024.3502296.
- [23] B. Saravana Kumar et A. Suyampulingam, "Comparative evaluation of path planning algorithms in a simulated disaster environment," Dans *MysuruCon 2022 - 2022 IEEE 2nd Mysore Sub Section International Conference*, 2022.
- [24] R. Yonetani, T. Taniai, M. Barekatain, M. Nishimura, et A. Kanezaki, "Path Planning using Neural A* Search," Dans *Proceedings of Machine Learning Research*, 2021, vol. 139, pp. 12029-12039.
- [25] Z. Changxin *et al.*, "UAV Electric Patrol Path Planning Based on Improved Ant Colony Optimization-A* Algorithm," Dans *2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms, EEBDA 2022*, 2022, pp. 1374-1380.
- [26] S. R. Bassolillo, E. D'Amato, I. Notaro, L. D'Agati, G. Merlino, et G. Tricomi, "Bridging ACO-Based Drone Logistics and Computing Continuum for Enhanced Smart City Applications," *Drones*, vol. 9, no. 5, Article 368, 2025. doi: 10.3390/drones9050368.

- [27] N. Tenniche et B. Mendil, "A New Hybrid Water Cycle Algorithm for Path Planning with Obstacle Avoidance for Indoor Assistant Autonomous UAV Navigation," *Periodica Polytechnica Electrical Engineering and Computer Science*, vol. 69, no. 2, pp. 175-187, 2025. doi: 10.3311/PPee.39765.
- [28] S. J. Fusic, R. Sitharthan, S. S. Masthan, et K. Hariharan, "Autonomous vehicle path planning for smart logistics mobile applications based on modified heuristic algorithm," *Measurement Science and Technology*, vol. 34, no. 3, Article 034004, 2023. doi: 10.1088/1361-6501/aca708.
- [29] Q. Cheng, Z. Zhang, Y. Du, et Y. Li, "Research on Particle Swarm Optimization-Based UAV Path Planning Technology in Urban Airspace," *Drones*, vol. 8, no. 12, Article 701, 2024. doi: 10.3390/drones8120701.
- [30] J. Xin, J. Kim, S. Chu, et N. Li, "OkayPlan: Obstacle Kinematics Augmented Dynamic real-time path Planning via particle swarm optimization," *Ocean Engineering*, vol. 303, Article 117841, 2024. doi: 10.1016/j.oceaneng.2024.117841.
- [31] V. T. Hoang, M. D. Phung, T. H. Dinh, et Q. P. Ha, "Angle-Encoded Swarm Optimization for UAV Formation Path Planning," Dans *IEEE International Conference on Intelligent Robots and Systems*, 2018, pp. 5239-5244.
- [32] D. Gao, P. Zhou, W. Shi, T. Wang, et Y. Wang, "A Dynamic Obstacle Avoidance Method for Unmanned Surface Vehicle under the International Regulations for Preventing Collisions at Sea," *Journal of Marine Science and Engineering*, vol. 10, no. 7, Article 901, 2022. doi: 10.3390/jmse10070901.
- [33] D. Wang, J. Zhang, J. Jin, et X. Mao, "Local collision avoidance algorithm for a unmanned surface vehicle based on steering maneuver considering colregs," *IEEE Access*, vol. 9, pp. 49233-49248, Article 3058288, 2021. doi: 10.1109/ACCESS.2021.3058288.
- [34] J. Zhuang, Y. Zhang, P. Xu, Y. Zhao, J. Luo, et S. Song, "Multiple Moving Obstacles Avoidance for USV using Velocity Obstacle Method," Dans *Proceedings of 2021 IEEE International Conference on Unmanned Systems, ICUS 2021*, 2021, pp. 140-145.
- [35] G. Sharma et S. Jain, "Flight Evolution: Decoding Autonomous UAV Navigation—Fundamentals, Taxonomy, and Challenges," *Transactions on Emerging Telecommunications Technologies*, vol. 36, no. 4, Article e70111, 2025. doi: 10.1002/ett.70111.
- [36] Z. Fan et al., "UAV Collision Avoidance in Unknown Scenarios with Causal Representation Disentanglement," *Drones*, vol. 9, no. 1, Article 10, 2025. doi: 10.3390/drones9010010.
- [37] S. Sabzekar, M. Samadzad, A. Mehditabrizi, et A. N. Tak, "A Deep Reinforcement Learning Approach for UAV Path Planning Incorporating Vehicle Dynamics with Acceleration Control," *Unmanned Systems*, vol. 12, no. 3, pp. 477-498, 2024. doi: 10.1142/S2301385024420044.
- [38] S. Zhang, Y. Li, F. Ye, X. Geng, Z. Zhou, et T. Shi, "A Hybrid Human-in-the-Loop Deep Reinforcement Learning Method for UAV Motion Planning for Long Trajectories with Unpredictable Obstacles," *Drones*, vol. 7, no. 5, Article 311, 2023. doi: 10.3390/drones7050311.
- [39] X. Chen, Y. Qi, Y. Yin, Y. Chen, L. Liu, et H. Chen, "A Multi-Stage Deep Reinforcement Learning with Search-Based Optimization for Air–Ground Unmanned System Navigation," *Applied Sciences (Switzerland)*, vol. 13, no. 4, Article 2244, 2023. doi: 10.3390/app13042244.

- [40] B. Sangiovanni, G. P. Incremona, M. Piastra, et A. Ferrara, "Self-Configuring Robot Path Planning with Obstacle Avoidance via Deep Reinforcement Learning," *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 397-402, Article 9119090, 2021. doi: 10.1109/LCSYS.2020.3002852.
- [41] C. Yan, X. Xiang, et C. Wang, "Towards Real-Time Path Planning through Deep Reinforcement Learning for a UAV in Dynamic Environments," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 98, no. 2, pp. 297-309, 2020. doi: 10.1007/s10846-019-01073-3.
- [42] L. Li, D. Wu, Y. Huang, et Z. M. Yuan, "A path planning strategy unified with a COLREGS collision avoidance function based on deep reinforcement learning and artificial potential field," *Applied Ocean Research*, vol. 113, Article 102759, 2021. doi: 10.1016/j.apor.2021.102759.
- [43] J. Xia, X. Zhu, Z. Liu, Y. Luo, Z. Wu, et Q. Wu, "Research on Collision Avoidance Algorithm of Unmanned Surface Vehicle Based on Deep Reinforcement Learning," *IEEE Sensors Journal*, vol. 23, no. 11, pp. 11262-11273, 2023. doi: 10.1109/JSEN.2022.3222575.
- [44] Y. Tao et J. Du, "Agile collision avoidance for unmanned surface vehicles based on collision shielded model prediction control algorithm," *Journal of Navigation*, vol. 75, no. 5, pp. 1243-1267, 2022. doi: 10.1017/S0373463322000315.
- [45] S. Yang, J. Huang, X. Xiang, et W. Li, "Optimization of USV area coverage path planning based on confidence ellipsoid," *Xi Tong Gong Cheng Yu Dian Zi Ji Shu/Systems Engineering and Electronics*, vol. 44, no. 7, pp. 2263-2269, 2022. doi: 10.12305/j.issn.1001-506X.2022.07.22.
- [46] X. Zhu, B. Yan, et Y. Yue, "Path planning and collision avoidance in unknown environments for usvs based on an improved d* Lite," *Applied Sciences (Switzerland)*, vol. 11, no. 17, Article 7863, 2021. doi: 10.3390/app11177863.
- [47] N. Su, J. B. Wang, C. Zeng, H. Zhang, M. Lin, et G. Y. Li, "Unmanned-Surface-Vehicle-Aided Maritime Data Collection Using Deep Reinforcement Learning," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 19773-19786, 2022. doi: 10.1109/JIOT.2022.3168589.
- [48] Y. S. Law-Kam Cio, M. Raison, C. Leblond Menard, et S. Achiche, "Proof of Concept of an Assistive Robotic Arm Control Using Artificial Stereovision and Eye-Tracking," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 12, pp. 2344-2352, Article 8887462, 2019. doi: 10.1109/TNSRE.2019.2950619.
- [49] K. Kamali, I. A. Bonev, et C. Desrosiers, "Real-time Motion Planning for Robotic Teleoperation Using Dynamic-goal Deep Reinforcement Learning," Dans *Proceedings - 2020 17th Conference on Computer and Robot Vision, CRV 2020*, 2020, pp. 182-189.
- [50] F. Liu, D. Wang, T. Zhao, et Y. Y. Zhou, "Improved RRT * Algorithm Combined with an Adaptive Probability-Adjusted Goal-Directed Node Strategy," Dans *2023 5th International Conference on Robotics, Intelligent Control and Artificial Intelligence, RICAI 2023*, 2023, pp. 1121-1124.
- [51] S. Secil et M. Ozkan, "A collision-free path planning method for industrial robot manipulators considering safe human-robot interaction," *Intelligent Service Robotics*, vol. 16, no. 3, pp. 323-359, 2023. doi: 10.1007/s11370-023-00465-7.

- [52] L. Ren, Y. Kang, L. Yang, H. Jia, et S. Wang, "Optimization Algorithm for 3D Smooth Path of Robotic Arm in Dynamic Obstacle Environments," *Applied Sciences (Switzerland)*, vol. 15, no. 4, Article 2116, 2025. doi: 10.3390/app15042116.
- [53] J. Mao *et al.*, "Robust UAV Path Planning with Obstacle Avoidance for Emergency Rescue," Dans *IEEE Wireless Communications and Networking Conference, WCNC*, 2025.
- [54] J. Goricanec, A. Milas, L. Markovic, et S. Bogdan, "Collision-Free Trajectory Following With Augmented Artificial Potential Field Using UAVs," *IEEE Access*, vol. 11, pp. 83492-83506, 2023. doi: 10.1109/ACCESS.2023.3303109.
- [55] A. Batinovic, J. Goricanec, L. Markovic, et S. Bogdan, "Path Planning with Potential Field-Based Obstacle Avoidance in a 3D Environment by an Unmanned Aerial Vehicle," Dans *2022 International Conference on Unmanned Aircraft Systems, ICUAS 2022*, 2022, pp. 394-401.
- [56] Y. Yan, Z. Lv, J. Yuan, et S. Zhang, "Obstacle avoidance for multi-UAV system with optimized artificial potential field algorithm," *International Journal of Robotics and Automation*, vol. 36, no. 10, 2021. doi: 10.2316/J.2021.206-0610.
- [57] F. A. L. Lifen, B. S. Ruoxin, C. L. Shuandao, et D. W. Jiang, "Path planning for UAVS based on improved artificial potential field method through changing the repulsive potential function," Dans *CGNCC 2016 - 2016 IEEE Chinese Guidance, Navigation and Control Conference*, 2017, pp. 2011-2015.
- [58] K. Merckaert, B. Convens, M. M. Nicotra, et B. Vanderborght, "Real-time constraint-based planning and control of robotic manipulators for safe human–robot collaboration," *Robotics and Computer-Integrated Manufacturing*, vol. 87, Article 102711, 2024. doi: 10.1016/j.rcim.2023.102711.
- [59] R. Meziane, M. J. D. Otis, et H. Ezzaidi, "Human-robot collaboration while sharing production activities in dynamic environment: SPADER system," *Robotics and Computer-Integrated Manufacturing*, vol. 48, pp. 243-253, 2017/12/01/ 2017. doi: <https://doi.org/10.1016/j.rcim.2017.04.010>.
- [60] C. A. Cheng *et al.*, "RMPflow: A Geometric Framework for Generation of Multitask Motion Policies," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 968-987, Article 9388869, 2021. doi: 10.1109/TASE.2021.3053422.
- [61] I. RealSense, "Intel® RealSense™ Product Family D400 Series," *Datasheet, Rev*, vol. 15, 2023.
- [62] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, et W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189-206, 2013/04/01 2013. doi: 10.1007/s10514-012-9321-0.
- [63] A. Alzadjali, M. Mushtaq, F. Esposito, C. Fiandrino, et J. Deogun, "OctoMap: Supporting Service Function Chaining via Supervised Learning and Online Contextual Bandit," Dans *2021 IEEE 7th International Conference on Network Softwarization (NetSoft)*, 2021, pp. 133-141.
- [64] Y. Yue *et al.*, "Probabilistic 3D Semantic Map Fusion Based on Bayesian Rule," Dans *Proceedings of the IEEE 2019 9th International Conference on Cybernetics and Intelligent Systems and Robotics, Automation and Mechatronics, CIS and RAM 2019*, 2019, pp. 542-547.
- [65] S. Zhu, Y. Tang, B. Li, X. Wang, et H. Chen, "Angle-Weighted Time-to-Collision Algorithm for Enhanced Obstacle Avoidance in Autonomous Driving," Dans *SAE Technical Papers*, 2025.

- [66] F. Gomes, T. Hormigo, et R. Ventura, "Vision based real-time obstacle avoidance for drones using a time-to-collision estimation approach," Dans *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR 2020*, 2020, pp. 90-95.
- [67] M. J. D. Otis, T. L. Nguyen-Dang, D. Laurendeau, et C. Gosselin, "Interference estimated time of arrival on a 6-DOF cable-driven haptic foot platform," Dans *Proceedings - IEEE International Conference on Robotics and Automation*, 2009, pp. 1067-1072.
- [68] Y. Zhang, Y. Hu, J. Lu, et Z. Shi, "Research on Path Planning of Mobile Robot Based on Improved Theta* Algorithm," *Algorithms*, vol. 15, no. 12 2022. doi: 10.3390/a15120477.
- [69] Y. Gao, Q. Han, S. Feng, Z. Wang, T. Meng, et J. Yang, "Improvement and Fusion of D*Lite Algorithm and Dynamic Window Approach for Path Planning in Complex Environments," *Machines*, vol. 12, no. 8 2024. doi: 10.3390/machines12080525.
- [70] W. Zai, Q. Lin, S. Wang, et X. Lin, "Path Planning for Wheeled Robots Based on the Fusion of Improved A* and TEB Algorithms," Dans *2023 China Automation Congress (CAC)*, 2023, pp. 3257-3261.