



Contributions à la coordination de tâches et de mouvements pour un système multi-robots

Zhi Yan

► **To cite this version:**

Zhi Yan. Contributions à la coordination de tâches et de mouvements pour un système multi-robots. Robotique [cs.RO]. Université Paris 8, 2012. Français. <tel-01151617>

HAL Id: tel-01151617

<https://hal.archives-ouvertes.fr/tel-01151617>

Submitted on 13 May 2015

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

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

Université Paris 8
Laboratoire d'Informatique Avancée de Saint-Denis

THÈSE

pour l'obtention du

Doctorat de l'Université Paris 8
spécialité Informatique

présentée par

Zhi YAN

Contributions à la coordination de tâches et de mouvements pour un système multi-robots

Thèse soutenue le 10 décembre 2012 devant le jury :

Raja CHATILA	Directeur de recherche CNRS à l'ISIR	Rapporteur
Zahia GUESSOUM	MCF HDR, Université de Reims	Rapporteur
Lakhdar SAIS	Professeur, Université d'Artois	Examineur
Patrick GREUSSAY	Professeur, Université Paris 8	Examineur
Arab ALI CHÉRIF	Professeur, Université Paris 8	Directeur
Nicolas JOUANDEAU	MCF, Université Paris 8	Codirecteur

A mes grands-parents maternels

Remerciements

Je remercie Patrick GREUSSAY d'avoir accepté de présider le jury.

Je remercie vivement Arab ALI CHÉRIF pour sa confiance et son soutien. Je tiens à exprimer toute ma reconnaissance à mon directeur de thèse, pour avoir su orienter mes aspirations en me laissant une grande liberté.

Je dois beaucoup à Nicolas JOUANDEAU qui depuis plusieurs années à l'Université Paris 8 m'a fait profiter de son immense expérience et m'a accordé avec générosité, aide et amitié. Je ne saurais assez le remercier pour le rôle majeur qu'il a joué durant plusieurs années : pour la qualité de sa direction, pour ses nombreuses remarques et appréciations d'une extrême pertinence. Je tiens en cette circonstance à lui exprimer ma profonde reconnaissance.

Je remercie Zahia GUESSOUM et Raja CHATILA de m'avoir fait l'honneur d'accepter d'être rapporteurs de ma thèse, de participer au jury et de s'être intéressés à mes travaux.

Je remercie Lakhdar SAIS pour sa participation au jury.

Je remercie Lynda SEDDIKI et Youcef TOUATI qui ont contribué, au cours de nombreuses discussions à l'achèvement de ce travail.

Je remercie tout particulièrement mon frère Youssef RIDENE pour son amitié, son aide et ses encouragements constants.

Enfin je tiens à remercier sincèrement mon épouse Xiao Fang LIU pour son soutien, sa patience et ses encouragements constants.

Que tous en soient remerciés.

Résumé

Nous étudions dans ce mémoire le problème de la coordination de tâches et de mouvements pour un système multi-robots du point de vue de la planification. Nos travaux se concrétisent par quatre nouvelles approches : une approche contrarienne centrée sur les échanges, une approche heuristique utilisant un modèle empirique des résultats escomptés, une approche par échantillonnage dédié à la décision et une approche par échantillonnage adaptatif. Toutes les approches proposées dans ce mémoire ont été testées et évaluées par des expériences de simulation. Le mémoire s'articule autour de cinq chapitres.

Le chapitre 1 introduit nos objectifs et présente un état de l'art des systèmes multi-robots, par comparaison avec les systèmes mono-robot, par comparaison de la nature des environnements pouvant être coopératifs ou compétitifs. Partant des problèmes inhérents aux systèmes multi-robots, nous présentons les solutions existantes dans le domaine de la planification de tâches et de la planification de mouvements. Nous concluons ce chapitre par un positionnement de nos travaux.

Le chapitre 2 présente nos deux premières contributions pour la planification de tâches. La première est une approche contrarienne, inspirée de principes de microéconomie, centrée sur l'identification de règles d'interaction proposant un modèle d'exécution simple et robuste. La seconde est une approche heuristique basée sur une estimation empirique des résultats escomptés permettant de minimiser le temps total d'exécution des missions tout en conservant une faible consommation énergétique des robots.

Le chapitre 3 présente deux autres contributions pour la planification de mouvements. Notre troisième approche est réalisée par échantillonnage dédié à la décision. Cette approche permet la synthèse de nouveaux points de passage : le *diagramme de Voronoï* assure la synthèse de points de passage primaires, enrichie par un échantillonnage permettant de créer des points de passage secondaires, formant un ensemble représentatif des contraintes topologiques de l'environnement. Notre quatrième approche est réalisée par échantillonnage adaptatif probabiliste de l'espace des configurations des robots. Pour chaque robot, l'espace des configurations est composé d'obstacles statiques et de robots mobiles. L'échantillonnage de cet espace ainsi réalisé avant exécution de la mission permet de réduire la dépendance

entre la dimension de l'espace et la complexité du problème.

Le chapitre 4 présente les expérimentations menées pour valider nos contributions. Ces expérimentations sont réalisées avec le simulateur *Player/Stage*. Notre première approche est validée par comparaison avec la *méthode hongroise* dans un problème d'exploration en environnement inconnu. Les résultats montrent la simplicité et la robustesse de notre première approche. Notre deuxième approche est validée dans une expérience de transport de marchandises. Les marchandises apparaissent aléatoirement sur un site producteur et chaque marchandise est transportable par un unique robot vers un site consommateur. Les résultats de notre approche sont comparés avec ceux de l'approche *re-planner*. Le caractère novateur de notre approche réside dans l'application d'un modèle heuristique de l'estimation de l'achèvement de la mission. Notre troisième approche est validée dans une expérience d'exploration dont les résultats sont comparés à l'utilisation d'un *diagramme de Voronoï*. Cette expérience compare notre approche avec une solution optimale en distance et sous-optimale en temps d'exécution. Les résultats de notre approche sont quasi équivalents en distance et meilleurs en temps d'exécution. Notre quatrième approche est validée dans une expérience de transport de marchandises par comparaison avec une approche générale. Les résultats de notre approche montrent des temps de transport inférieurs à ceux de l'approche générale. L'originalité de notre quatrième approche réside dans la proposition d'un ensemble de solutions envisageables pour chaque robot, permettant de multiples scénarios d'exécution et une réduction des possibles situations d'attente.

Le chapitre 5 présente une conclusion de l'ensemble de nos contributions et les différentes perspectives faisant suite à nos travaux.

Table des matières

1	Introduction	1
1.1	Systèmes multi-robots	1
1.2	Mono-robot vs multi-robots	4
1.3	Coopératifs vs compétitifs	7
1.4	Problèmes inhérents	9
1.5	Coordination : statique vs dynamique	12
1.6	Communication : explicite vs implicite	13
1.7	Planification	15
1.7.1	Planification de tâches	16
1.7.2	Planification de mouvements	20
1.8	Prise de décision : centralisée vs décentralisée	23
1.9	Positionnement de nos travaux	25
2	Contributions à la planification de tâches	27
2.1	Approche contrarienne centrée sur les échanges	28
2.1.1	Méthodes existantes	29
2.1.2	Contributions	31
2.2	Approche heuristique basée sur un modèle empirique	36
2.2.1	Méthodes existantes	37
2.2.2	Contributions	38
2.3	Conclusion	42
3	Contributions à la planification de mouvements	45
3.1	Méthodes existantes	46
3.2	Situations d'attente	48
3.3	Approche par échantillonnage dédié à la décision	50
3.3.1	Diagramme de Voronoï	51
3.3.2	Feuille de route probabiliste	53
3.3.3	Méthode hongroise	56
3.3.4	Limite de l'approche	58
3.4	Approche par échantillonnage adaptatif	59

3.4.1	Échantillonnage de l'espace des configurations	60
3.4.2	Construction de feuille de route	62
3.4.3	Planification de mouvements	63
3.4.4	Limite de l'approche	64
3.5	Conclusion	65
4	Expérimentations	67
4.1	Approche contrarienne et exploration multi-robots	67
4.2	Approche heuristique et transport de marchandises	72
4.3	Approche par échantillonnage et exploration multi-robots	77
4.4	Approche par échantillonnage et transport de marchandises	79
4.5	Conclusion	82
5	Conclusion et perspectives	83
A	Exemples de solutions	87
B	Articles publiés	91
	Bibliographie	93

Table des figures

1.1	RHINO, ASIMO, NAO, MER-A et BigDog	6
2.1	Organigramme de prise de décision d'un robot acheteur	35
2.2	Organigramme de prise de décision d'un robot vendeur	36
2.3	Comparaison de deux plans de répartition des tâches	40
2.4	Relations entre comportements pour la commande d'un robot . . .	42
3.1	Exemples type d'exclusion mutuelle de point de passage	49
3.2	Trajectoires de robots évitant les situations d'attente	50
3.3	Exemple de <i>diagrammes de Voronoï</i> de deux cartes	52
3.4	Points de passage résultants du processus d'identification	53
3.5	Processus de génération d'une feuille de route probabiliste	55
3.6	Positions générées par échantillonnage sur la base du <i>diagramme de Voronoï</i>	56
3.7	Cas limite par échantillonnage dédié à la décision	59
3.8	Processus de génération d'une feuille de route à l'aide d'ACS-PRM	63
3.9	Cas limite par échantillonnage adaptatif	64
3.10	Résultats en environnement irrégulier avec l'approche ACS-PRM	65
4.1	Robot <i>Pioneer 2-DX</i> avec télémètre laser	68
4.2	Deux cartes utilisées dans un problème d'exploration	68
4.3	Exemple d'exécution de l'approche contrarienne	69
4.4	Robot <i>Pioneer 2-DX</i> avec pince	72
4.5	Quatre environnements expérimentaux de simulation	73
4.6	Exemple d'exécution de l'approche heuristique	74
4.7	Résultats comparés de l'approche heuristique et de l'approche <i>re-planner</i>	75
4.8	Exemple d'exécution de l'approche par échantillonnage dédié à la décision	78
4.9	Gain en temps dans un problème d'exploration	78
4.10	Deux cartes utilisées dans un problème de transport	80

4.11	Exemple d'exécution de l'approche par échantillonnage adaptatif .	81
4.12	Gain en temps dans un problème de transport	81
A.1	Trajectoires de robots obtenues avec l'algorithme OBO	88
A.2	Trajectoires de robots obtenues avec l'algorithme VFH+	89
A.3	Trajectoires de robots obtenues avec l'algorithme ND	90

Liste des tableaux

1.1	Comparaison entre trois méthodes de répartition de tâches	19
2.1	Comparaison entre deux méthodes de répartition de tâches	42
4.1	Temps d'exécution des missions d'exploration	70
4.2	Résultats avec PPI variable	71
4.3	Nombre moyen des tâches complétées et des énergies consommées dans l'environnement A	76
4.4	Nombre des exclusions mutuelles dans un problème d'exploration	79
4.5	Nombre des exclusions mutuelles dans un problème de transport .	82

Liste des algorithmes

2.1	Répartition des rôles pour un robot r	33
2.2	Répartition des tâches pour un robot vendeur	35
2.3	Répartition heuristique des tâches	41
3.1	Échantillonnage sur l'axe médian	56
3.2	Exploration par échantillonnage dédié à la décision	58
3.3	Échantillonnage croisé adaptatif	62

Chapitre 1

Introduction

Nous décrivons dans ce mémoire la modélisation et l'implémentation de quatre nouvelles approches pour la planification de tâches et de mouvements pour un systèmes multi-robots¹. Cette modélisation est associée à son implémentation sur le simulateur *Player/Stage* [46], qui est un simulateur reconnu et utilisé pour l'implémentation des algorithmes sur les robots mobiles. Dans ce premier chapitre, nous présentons l'état de l'art des systèmes multi-robots. Ces systèmes sont présentés par comparaison aux systèmes mono-robot qui les ont précédés dans le temps. Ce chapitre présente la dualité des interactions caractérisant les systèmes multi-robots comme des organisations soient coopératives soient compétitives. Ce chapitre présente également la problématique qu'ils soulèvent dans les domaines de la planification de tâches et de la planification de mouvements. Cependant l'utilisation de plusieurs robots implique la considération de problèmes inhérents dus aux possibilités d'un partage d'exécution, d'un partage de tâches et d'un partage d'espace. Dans ce contexte, la coordination pourra être statique ou dynamique et la communication pourra être explicite ou implicite.

1.1 Systèmes multi-robots

La notion de systèmes multi-robots ou *Multi-Robot Systems* (MRS) en anglais débute dans les années 1990, notamment dans des travaux regroupant des robots mobiles rassembleurs d'objets [97], des colonies de robots marcheurs [25, 26]. D'une part, une série de projets ont été réalisés avec succès, tel que les projets

1. Le mot robot désigne une machine créée par l'homme pour accomplir des tâches. Dans cette optique, les recherches et les développements en robotique sont destinés à assister, remplacer, prolonger ou améliorer le travail de l'homme. Les domaines d'application les plus courants sont l'industrie manufacturière, les applications militaires, l'exploration spatiale, le transport de marchandises et les applications médicales.

suivants : ACTRESS² [7], GOFER³ [21], CEBOT⁴ [39], ALLIANCE⁵ [90], COMETS⁶ [87], M+⁷ [17], MURDOCH⁸ [40] et ASyMTRe⁹ [118]. D'autre part, la quantité importante des travaux de recherche publiés sur le sujet des systèmes multi-robots permet ainsi de produire une taxonomie et une analyse des solutions présentées dans les travaux existants. Dudek *et al.* [35] ont présenté une taxonomie qui classe les systèmes multi-agents ou *Multi-Agent Systems* (MAS) en anglais notamment selon la communication et la capacité de calcul. Ils présentent également des résultats supplémentaires par comparaison des méthodes existantes afin d'illustrer l'utilité de la taxonomie présentée et de démontrer qu'un collectif peut être plus puissant qu'une seule unité (*i.e.* qu'un ensemble multi-robots sera meilleur qu'un simple robot dans certains cas). Cao *et al.* [22] ont donné un aperçu de la littérature en robotique mobile coopérative jusqu'à la moitié des années 1990. Ils classent la littérature selon cinq catégories correspondant à des points de vues qui sont : l'architecture du groupe, le conflit de ressources, l'ori-

2. ACTRESS (ACTor-based Robot and Equipments Synthetic System) est un système décentralisé, qui comprend trois robots et trois postes de travail. Ce système forme un groupe hétérogène conçu pour effectuer des tâches qui ne peuvent pas être accomplies par un seul robot comme les problèmes du type boîte-poussant.

3. GOFER comprend un module central pour la planification de tâches. Ce module communique avec tous les robots et propose une vue globale des tâches à effectuer et des robots disponibles.

4. CEBOT (CELLular roBOTics System) est un système décentralisé, inspiré par l'architecture hiérarchique de l'organisation cellulaire des entités biologiques. Ce système est dynamiquement reconfigurable au niveau de ses "cellules" (*i.e.* robots). Les cellules sont physiquement couplables entre elles. Ceci permet de reconfigurer dynamiquement leur structure pour une configuration optimale en réponse aux modifications de l'environnement.

5. ALLIANCE est un système décentralisé, dans lequel les robots sont supposés pouvoir, avec une probabilité, sentir les effets de leurs propres actions et les actions des autres agents au travers de perceptions et de communications explicites.

6. COMETS est un système hétérogène, conçu pour l'exploration des milieux naturels au travers d'une coopération entre plusieurs robots aériens. L'originalité de ces travaux étend le problème de la coopération multi-robots aux architectures hétérogènes (composées de deux hélicoptères et d'un dirigeable) et ouvre ainsi de nouvelles perspectives d'applications dans le domaine civil.

7. M+ est un système décentralisé, dans lequel chaque robot possède sa propre connaissance du monde et peut décider de ses actions futures en tenant compte de son contexte actuel et de la tâche à accomplir.

8. MURDOCH est un système décentralisé, conçu pour traiter le problème de la répartition de tâches. Ce système utilise un protocole de négociation simple distribué qui répartit les tâches via une séquence d'enchère au premier prix.

9. ASyMTRe (*Automated Synthesis of Multi-robot Task solutions through software Reconfiguration*) est un système décentralisé, qui reconfigure automatiquement les connexions de schémas (y compris le schéma de la cartographie d'environnement, le schéma de la perception et le schéma du contrôle de moteur) au sein et entre les robots pour synthétiser les comportements de robots pour accomplir les objectifs de l'équipe.

gine de la coopération, l'apprentissage et la géométrie. Ils présentent également les contraintes découlant des limitations technologiques et les lacunes possibles dans l'ensemble des travaux existants. Stone et Veloso [115] ont présenté quatre scénarios dans le cadre de l'utilisation d'un système multi-agents, associés à l'utilisation de différents types d'agents : des agents homogènes non communicants, des agents hétérogènes non communicants, des agents homogènes communicants et des agents hétérogènes communicants. Ils illustrent les différents scénarios en utilisant le problème de la poursuite et décrivent les travaux existants du domaine. Les techniques présentées sont biaisées en faveur des méthodes d'apprentissage. Arai *et al.* [6] ont identifié sept thèmes de recherche primaires dans les systèmes multi-robots : les inspirations biologiques, la communication, les architectures, la localisation / la cartographie / l'exploration, le transport et la manipulation d'objet, la coordination de mouvements et les robots reconfigurables. Ils discutent également d'articles traitant de problèmes spécifiques et suggèrent plusieurs questions de recherche supplémentaires. Farinelli *et al.* [36] ont présenté un horizon des travaux réalisés jusqu'au début des années 2000 dans le domaine de la coopération et la coordination des systèmes multi-robots. Par ailleurs, ils proposent une taxonomie pour la classification des systèmes multi-robots fondée sur la notion de coordination, dont la complexité se caractérise selon deux dimensions : les dimensions de l'espace de coordination et les dimensions de l'espace des configurations du système utilisé.

Les travaux étudiés dans ce chapitre se concentrent sur les robots mobiles où un robot mobile est considéré comme un agent *mobile, autonome* et physiquement indépendant. Notre thème d'investigation est ainsi limité aux systèmes multi-robots mobiles ou *Multiple Mobile Robot Systems* (MMRS) en anglais. Par ailleurs, il est important de ne pas confondre les systèmes multi-robots avec les systèmes multi-agents ou l'intelligence artificielle distribuée ou *Distributed Artificial Intelligence* (DAI) en anglais. Parce que :

- les systèmes multi-robots se réfèrent à des systèmes composés de robots, dans lesquels les robots mobiles se déplacent dans le monde physique et doivent interagir les uns avec les autres physiquement ;
- les systèmes multi-agents se réfèrent aux systèmes traditionnels d'ordinateurs distribués dans lesquels les individus sont des nœuds stationnaires, parfois dotés de capacités d'introspection permettant réorganisation et passage à l'échelle [50] ;
- et l'intelligence artificielle distribuée concerne des agents stationnaires (en position fixe, par opposition aux robots roulants, en positions mobile) impliquant des agents logiciels.

Ce chapitre est organisé comme suit : la section 1.2 décrit les systèmes multi-robots par comparaison aux systèmes mono-robot ; la section 1.3 présente les différences entre systèmes multi-robots coopératifs et systèmes multi-robots compé-

titifs ; la section 1.4 présente le conflit de ressources, qui est un problème inhérent des systèmes multi-robots ; la section 1.5 présente la notion de coordination comme pierre angulaire de l'utilisation des systèmes multi-robots permettant de tirer parti de leur nature distribuée ; la section 1.6 présente deux types de mécanisme de communication : l'un explicite et l'autre implicite ; la section 1.7 présente le problème de la planification basée sur la coordination, qui comprend la planification des tâches et la planification de mouvements ; la section 1.8 présente deux mécanismes de décision : l'un centralisé et l'autre décentralisé ; le chapitre se conclut par une discussion dans la section 1.9.

1.2 Mono-robot vs multi-robots

Un système mono-robot ou *Single-Robot System* (SRS) en anglais ne contient qu'un seul robot capable de modéliser lui-même, l'environnement, et les interactions entre les deux [115]. Un robot d'un tel système est souvent conçu pour faire un travail pour son propre compte. De tels robots sont de fait des systèmes complexes et coûteux. Bien qu'un robot puisse avoir un rendement élevé, certaines tâches sont complexes, voir impossibles pour être effectuées comme des tâches séparées dans l'espace. Par conséquent, il existe une restriction inhérente aux systèmes mono-robot, qui est la limitation dans l'espace. L'éventail des tâches possibles donne lieu à une diversité de plateformes telles que RHINO¹⁰ [19],

10. RHINO est un robot mobile présenté par Université rhénane Frédéric-Guillaume de Bonn (Allemagne) en 1994 pendant la compétition et l'exposition de robot AAI. Le matériel du robot est fabriqué par l'entreprise *Real World Interface*. Il est équipé de 24 capteurs de proximité à ultrasons, d'un système à double caméras couleurs et de deux ordinateurs i486 embarqués. Le projet RHINO est une réalisation logicielle de systèmes d'apprentissage autonomes et complexes.

ASIMO¹¹ [105], MER-A¹² [1], BigDog¹³ [96] et NAO¹⁴ [48] (cf. figure 2.3).

L'utilisation d'un système multi-robots présente plusieurs avantages par comparaison à un système mono-robot :

- Les systèmes multi-robots présentent une meilleure capacité de répartition spatiale. Dudek *et al.* [35] ont donné un exemple de tâche de lancement de missiles¹⁵, qui exige une sorte de synchronisation.
- Les systèmes multi-robots présentent de meilleurs résultats d'exécution. Les indicateurs les plus courants sont le temps total nécessaire à la terminaison d'une tâche [20] ou la consommation énergétique des robots [126]. La comparaison de la somme des consommations d'énergie dans un système multi-robots et de la consommation d'un robot dans un système mono-robot nous place donc dans un contexte d'économie globale de l'énergie.
- Les systèmes multi-robots introduisent une notion de robustesse résultante de la fusion de données et du partage d'informations entre les robots. Cette notion de robustesse est synonyme de tolérance aux pannes résultante de la redondance de l'information. Plusieurs robots peuvent se localiser plus efficacement s'ils échangent des renseignements sur leur position respective au fil de l'exécution d'une mission¹⁶ [38, 101, 75].

11. ASIMO est un robot humanoïde développé par la société japonaise *Honda*. La première version fut finalisée en 2000 avec 26 degrés de liberté de mouvement et 30 minutes d'autonomie. La dernière version avec 57 degrés de liberté de mouvement et 40 minutes d'autonomie en marchant, qui date de 2011, mesure 130 cm pour 48 kg et possède une vitesse de course de 9 km/h.

12. MER-A (abréviation de *Mars Exploration Rover-A*), également connu sous le nom *Spirit*, est un astromobile envoyé sur Mars par la *NASA* en 2003 par une des deux sondes du programme *Mars Exploration Rover*. Ce robot possède six roues, est alimenté par énergie solaire, mesure 1,5 m de hauteur pour 2,3 m de largeur et 1,6 m de long en pesant 180 kg. Chaque roue possède un moteur indépendant. Le véhicule est dirigeable à l'avant et à l'arrière. Il est conçu pour fonctionner en toute sécurité à des inclinaisons maximales de 30 degrés. Sa vitesse maximale est de 0,18 km/h.

13. BigDog est un robot quadrupède à l'allure d'un grand chien créé en 2005 par la société américaine *Boston Dynamics*. Chaque jambe du BigDog comporte quatre vérins hydrauliques contrôlant les articulations et d'un cinquième degré de liberté passif. Il mesure 91 cm de long pour 76 cm de haut et pèse 110 kg soit la taille d'une petite mule. Il est capable de traverser un terrain difficile à 6,4 km/h, de porter un chargement de 150 kg et de grimper des pentes de 35 degrés. Il a une autonomie de 2,5 heures pour 10 kilomètres en marchant.

14. NAO est un robot humanoïde développé par la société française *Aldebaran Robotics*, présenté pour la première fois au public fin 2006. Il existe en différents modèles qui comportent de 14 à 25 degrés de liberté et dispose d'environ 45 minutes d'autonomie.

15. Il existe deux touches qui sont séparées par une grande distance dans l'espace, et qui doivent être activés en même temps.

16. Quand un robot est assigné à une mission, il se voit doté d'un comportement (ou d'une pensée) individuel(le). Quand les robots sont assignés en groupe à la réalisation d'une mission (comportant une ou plusieurs tâches) dans un espace de travail commun, il apparaît un comportement (ou une pensée) collectif(ctive). Dans un cadre d'exécution multi-robots, il s'agit de trouver le comportement collectif propice à la synergie dans l'exécution de la mission choisie, ce qui pré-suppose de l'existence d'une relation de coopération (ayant une contribution globalement positive)



FIGURE 1.1 – RHINO, ASIMO, NAO, MER-A et BigDog

- Les systèmes multi-robots présentent dans certains cas des coûts inférieurs, dans le sens où la construction, la mise au point, le déploiement et la maintenance d'un système complexe peut devenir plus difficile que pour un en-

entre les robots. Dans ce premier chapitre, nous passons en revue l'état de l'art de la coordination multi-robots. Par ailleurs, nous présentons les implications associées à ces travaux, sur les questions de planification pour un système multi-robots, vues comme un problème de planification de tâches (avec une commande par action) et vu comme un problème de planification de mouvements (avec une commande par asservissement des actionneurs de chaque robot).

semble de robots (supposés plus simples¹⁷).

Un système multi-robots est homogène ou hétérogène. Dans les équipes homogènes, les capacités des robots individuels sont identiques (même si les structures physiques ne sont pas nécessairement les mêmes). Des travaux pertinents utilisant des équipes homogènes sont présentés dans [123, 142, 12, 31, 126]. Dans les équipes hétérogènes, les capacités des robots sont différentes, dans le sens où les robots sont spécialisés pour des tâches particulières. Des travaux pertinents utilisant des équipes hétérogènes sont présentés dans [7, 90, 40, 75, 16]. Les systèmes hétérogènes sont admis comme plus complexes que les systèmes homogènes.

1.3 Coopératifs vs compétitifs

La notion de comportement collectif dans les systèmes multi-robots se définit par similitude avec le principe de travail collectif dans notre société. Popenoe [95] a défini un comportement collectif comme un comportement dans lequel il apparaît une réponse à une influence ou à un stimulus commun dans une situation spontanée, imprédictible, non-structurée et instable. Le comportement collectif comprend deux sous-catégories de comportements associées aux deux notions de coopération (appelé comportement coopératif) et de compétition (appelé comportement compétitif). Les environnements multi-robots sont ainsi soit *coopératifs*, soit *compétitifs* [102].

Une première définition présente la coopération par l'identification de situations dans lesquelles il apparaît un besoin d'interaction entre plusieurs robots. Cette coopération devient utile quand le résultat de l'exécution de la tâche est estimé meilleur avec plusieurs robots. En effet, la coopération est l'interaction entre des robots qui travaillent dans un intérêt commun [11]. Parker [91] a introduit le concept de couverture, qui est un indice du niveau de coopération requis pour l'accomplissement d'une tâche : si la couverture est grande, alors les tâches peuvent être accomplies sans coopération ; si la couverture est faible, alors une réelle coopération est nécessaire. La couverture est maximale dans les groupes homogènes et elle diminue avec l'ajout de caractères hétérogènes. Des robots en coopération possèdent par essence un but commun, qui donne lieu à l'identification de sous-buts. Cet aspect de la décomposition d'un but en sous-buts est abondamment traité dans les travaux relatifs au problème de planification multi-robots (présenté en détails dans la section 1.7). Les exemples les plus représentatifs de coopération s'expriment dans les travaux de localisation multi-robots [38, 101, 75, 79],

17. Les architectures SWARM sont une approche multi-robots centrée sur la coordination d'un grand nombre de robots très simples. L'objectif est ici l'émergence de comportements par interaction entre les robots ou entre robots et environnement. Cette approche trouve de nombreuses inspirations dans l'étude des sociétés d'insectes[34].

d'exploration multi-robots [20, 99, 142, 108, 129, 130, 55, 30, 133, 134], de recherche et de sauvetage multi-robots [61, 141, 8, 12] et de transport multi-robots [4, 123, 126, 135, 136].

La notion de comportements compétitifs présente les situations et les problèmes à résoudre sous la condition que les robots soient en compétition les uns contre les autres pour satisfaire leur propre intérêt. Chaque robot possède dans ce cas, un ensemble d'intérêts partiellement ou totalement en concurrence avec les intérêts d'autres robots. Cette notion d'intérêt est définie par des fonctions d'utilité en accord ou en contradiction avec les autres robots [102]. Les exemples classiques de systèmes multi-robots compétitifs sont les jeux à deux joueurs à somme nulle tels que les échecs [2] et les compétitions de robots footballeurs présentés dans la RoboCup¹⁸ [3, 138, 63]. Dans les systèmes multi-robots mobiles, l'environnement est partiellement coopératif et partiellement compétitif car chaque robot doit éviter les collisions avec les autres robots (donc coopérer) et chaque robot cherchant à remplir des tâches peut occuper un espace de stationnement (donc être en compétition). Par conséquent, dans un environnement multi-robots coopératif ou compétitif, la mobilité des robots dans un espace commun implique un consensus :

- vu comme un mécanisme de communication (cf. section 1.6) ;
- ou encore vu comme un mécanisme de prise de décision (cf. section 1.8).

Une seconde définition présente la coordination comme un processus cognitif nouveau, permettant de nouveaux résultats idéalement supra-linéaire (*i.e.* avec un gain supérieur à la simple agglomération de systèmes mono-robot). Les notions de coopération et de compétition donnent lieu à des évaluations divergentes. Afin de pouvoir utiliser des indicateurs divergents, [36] considère un angle de vue par indicateur et définit la notion de relations hiérarchiques pour chaque angle de vue du problème. La coordination est ainsi substituée à la notion de communication, qui est utilisée comme un comportement rationnel dans les environnements multi-robots (la coordination est présentée en détails dans la section 1.5 et la communication dans la section 1.6).

18. La RoboCup regroupe un ensemble de catégories, dont les plus populaires sont les compétitions de robots footballeurs. Les catégories sont divisées en trois grands groupes, qui sont les robots footballeurs, les robots sauveteurs et les robots à domicile. Une catégorie accessible aux enfants regroupe également les thèmes de ces groupes. Pour chacun des groupes, simulation et plateformes réelles donnent lieu à différentes sous parties, dont les robots footballeurs sont une des plus populaires.

1.4 Problèmes inhérents

Si plusieurs demandes visant la même ressource apparaissent simultanément, il se produit un *conflit de ressources*. Ce problème a été étudié sous de nombreuses formes : un cas bien connu de problème de ce type est le problème de l'*exclusion mutuelle* [33] des systèmes de calculs distribués ou des réseaux multi-accès, où seulement une seule ressource est disponible à chaque instant. Dans le cas des robots mobiles, la notion de conflit de ressources apparaît lorsque plusieurs robots ont besoin de partager un même espace, de manipuler un même objet ou d'utiliser un même support de communication. Pour résoudre ce problème, les robots doivent se coordonner. La coordination peut être réfléchi à l'aide d'un plan (dont les deux niveaux conceptuels courants sont la planification des tâches¹⁹ et la planification de mouvements²⁰).

Les problèmes de partage de média de communication sont souvent associés à des limitations de bande passante. Ye *et al.* [139] ont présenté une méthode pour évaluer les stratégies de contrôle et de communication pour un groupe des robots dans les réseaux sans fil. Ils prennent en compte la largeur du canal de communication et comparent deux conceptions de contrôleurs dans la communication établie et comparent des modèles de propagation des ondes radio. Rybski *et al.* [103] ont décrit une architecture de contrôle de logicielle distribuée conçue pour le contrôle de plusieurs robots sur un canal de communication à faible bande passante. Cette étude décrit le cas d'un groupe de robots miniatures disposant d'une très faible capacité de fréquence radio. Les contraintes dimensionnelles de ces robots limitent la capacité de fréquence radio mais également la quantité de puissance de calcul possible, les forçant ainsi à s'appuyer sur des processus de décision déportés sur d'autres ressources de calculs distantes plus puissantes. Ainsi, toutes les données des capteurs sont diffusées vers un ordinateur distant ou vers un robot plus grand (*i.e.* plus puissant) et les commandes des actionneurs sont relayées vers les robots miniatures. Le fonctionnement de ces robots est complètement dépendant des canaux de communication radio qu'ils emploient. Mais les canaux de communication n'étant pas toujours partagés, il devient nécessaire de la part des robots d'obtenir un accès exclusif à ces canaux de communication. En outre, ils ont présenté des résultats expérimentaux sur une tâche de surveillance dans laquelle plusieurs robots patrouillent sur une zone. L'étude constate que le mouvement et la bande passante limitée affecte les performances du système dans l'accomplissement de la tâche. Une analyse de la dépendance entre largeur de la bande passante et nombre de robots impliqués est également présentée dans leurs travaux.

19. Pour une pensée de haut niveau, très abstraite, par action.

20. Pour une pensée de bas niveau, très concrète, proche des consignes de commandes à appliquer à chaque actionneur sur chaque robot.

La question de l'identification des sous-buts et de leur répartition dans un système multi-robots est équivalente au problème du type boîte-poussant multi-robots, dans lequel les robots partagent des tâches de manipulation. Mataric *et al.* [80] ont étudié le problème de la coopération dans le contexte du type boîte-poussant avec deux robots autonomes à six pattes²¹. Chaque robot communique avec les autres robots afin de coordonner la livraison de chaque boîte. Kube et Bonabeau [68] ont présenté des observations empiriques de transport coopératif chez les fourmis et décrivent une étude de cas des problèmes du type boîte-poussant par un essaim de robots. Gerkey et Mataric [42] ont proposé un système de répartition des tâches basé sur la notion d'enchères appelé MURDOCH qui est proposé pour la résolution de problème du type boîte-poussant. Ce système est capable de répartir et de coordonner les tâches de manipulation avec une tolérance aux fautes. Les résultats montrent une coopération étroitement couplée avec le nombre de robots. Miyata *et al.* [83] ont proposé une méthode d'affectation des tâches pour le transport coopératif multi-robots dans un environnement inconnu statique. Cette méthode divise les actions de chaque robot en petites unités à réaliser et affecte des tâches en examinant si le nombre de robots disponibles coïncide avec la tâche à réaliser. Cette méthode observe également la distribution des tâches dans le temps et dans l'espace, afin de minimiser les conflits. Cette observation présuppose des cas de répartition dans le temps et dans l'espace. Wang *et al.* [124] ont étudié le problème du transport multi-robots en présentant un algorithme de test rapide de clôture d'objet²².

Le problème du partage de l'espace a été étudié principalement dans les problèmes de planification de mouvements multi-robots et dans les problèmes d'évitement de collision, de congestion et d'interblocage. Jäger et Nebel [60] ont décrit une méthode décentralisée pour coordonner des trajectoires indépendantes, en impliquant plusieurs robots mobiles tout en évitant les collisions entre eux et tout en évitant l'apparition d'impasses dans les chemins des robots (les impasses ayant pour conséquence l'implication possible de points de rebroussement dans les chemins des robots, l'implication possible de chemins communs entre plusieurs robots et l'implication possible de points de croisements entre plusieurs robots). L'idée présentée dans leurs travaux est la suivante : chaque fois que la distance

21. Les robots à six pattes sont des hexapodes. Ces robots sont plus stables que les robots bipèdes du fait qu'ils soient statiquement stables dans la plupart des positions. De ce fait, ils ne dépendent pas de contrôleurs temps-réel pour rester debout ou pour marcher à petite allure. Il a été démontré qu'à grande vitesse de déplacement, les insectes sont dépendants de facteurs dynamiques. Dans l'absolu, il n'y a pas de frontières à la complexité de la morphologie d'une patte. En pratique, les pattes sont construites selon un modèle insecte avec deux à six degrés de liberté (créant ainsi de 12 à 36 degrés de libertés).

22. La clôture d'objet indique l'existence d'un chemin partant de la configuration actuelle permettant la saisie d'un objet fixé. Pour un objet fixé, elle définit la distance à partir de laquelle toute configuration admet un chemin se rapprochant de cet objet et permettant de s'en saisir.

entre deux robots est inférieure à une valeur fixée, chaque robot transmet les informations concernant sa trajectoire planifiée et évalue les collisions possibles. Si une collision possible est détectée, alors ils surveillent leurs mouvements et insèrent si nécessaire des temps d'inactivité entre des portions de leurs trajectoires afin d'éviter ces collisions. Chaque fois qu'une impasse est détectée, les planificateurs locaux de chacun des robots impliqués génèrent successivement une trajectoire alternative jusqu'à ce que l'impasse disparaisse. Luo *et al.* [74] ont proposé une approche de réseau de neurones pour la planification de trajectoire multi-robots. Dédiés à des tâches de nettoyage, les trajectoires des robots sont générées en temps réel en utilisant un réseau de neurones. Marcolino et Chaimowicz [76] ont proposé un algorithme de coordination décentralisé pour contrôler le trafic d'un essaim de robots, en évitant les situations de congestion lorsque les grands groupes de robots se déplacent en sens opposé. L'algorithme proposé permet aux robots de percevoir la possibilité d'une collision et d'avertir leurs coéquipiers grâce à des communications locales. Les groupes concernés modifient ainsi leur trajectoire pour éviter les congestions. Ils ont également proposé un algorithme de coordination [77] pour le contrôle de trafic quand les robots ont pour objectif local d'atteindre une même cible. Dans cette solution, les robots contrôlent leurs actions en utilisant une machine à états finis probabiliste, en comptant sur les mécanismes de détection de collision et de communication locale pour la coordination. Luna et Bekris [73] ont proposé un algorithme de planification de trajectoire multi-robots dans un graphe topologique comportant à chaque instant au moins deux sommets non alloués. L'algorithme proposé utilise deux primitives : l'une nommée "push", dans laquelle un robot se déplace vers son objectif jusqu'à ce qu'aucun progrès ne puisse être fait et l'autre nommée "swap", qui permet à deux robots d'échanger leurs positions, sans altérer la position de tout autre robot.

Le problème du partage de l'espace est un problème important en planification de mouvements. Ce problème apparaît quand plusieurs robots se déplacent vers un même *point de passage* de l'espace de travail, provoquant ainsi une congestion et par la même une collision de l'espace et un éventuel interblocage, qui peuvent compromettre les performances globales du système. Ce problème fait apparaître une pause, un état dans lequel tout ou partie du système (*i.e.* un ou plusieurs robots) arrête par exclusion des points de passage en exclusion mutuelle. Cet état de pause ou d'attente est appelé *situation d'attente*. Il s'agit selon les cas, d'attendre le passage d'un autre robot et dans l'hypothèse d'une congestion excessive, d'attendre une nouvelle planification complète de l'ensemble des trajectoires des robots du système. Dans nos travaux, nous nous concentrons sur l'atténuation possible des apparitions de *situation d'attente*. Nous présentons dans la section 3.3 une approche d'échantillonnage itérative associée à une carte de type grille d'occupation permettant de construire une carte topologique pour l'ensemble des robots [133]. Nous présentons également dans la section 3.4 une seconde approche

utilisant une feuille de route probabiliste basé sur un échantillonnage adaptatif croisé (*i.e.* ACS-PRM) [135].

1.5 Coordination : statique vs dynamique

La coordination est une tâche essentielle dans les systèmes multi-robots. La performance globale du système est directement influencée par la qualité de la coordination et la qualité du contrôle de l'exécution de chaque mouvement de chaque robot impliqué. Cette coordination peut être *statique* ou *dynamique*. La coordination statique (également connue sous le nom de coordination délibérative [59], ou coordination hors-ligne [121]), se réfère à l'adoption d'une convention entre plusieurs robots avant de s'engager dans une tâche. Dans [64], certaines règles comme "garder la droite", "s'arrêter aux intersections" et "garder suffisamment d'espace devant soi" sont utilisées pour résoudre les problèmes de contrôle de trafic. La coordination dynamique (également connue sous le nom de coordination réactive [59], ou coordination en ligne [121]) est faite lors de l'exécution d'une tâche et basée sur l'analyse et sur la synthèse de l'information. Les informations sont obtenues par des moyens de communication inter-robots. La méthode statique permet de gérer des tâches complexes, mais son contrôle en temps réel est faible. La méthode dynamique favorise la possibilité de réponses en temps réel, mais elle permet de traiter des tâches plus simples. La combinaison des deux approches basées sur les caractéristiques spécifiques de la tâche à accomplir permet de conserver le bénéfice des deux approches.

La coordination dynamique peut également être divisée en deux catégories : la coordination explicite et la coordination implicite. Gerkey et Mataric [44] ont défini :

- La coordination explicite comme l'ensemble des techniques employant une communication intentionnelle et des méthodes de collaboration très semblables à celles employées dans les systèmes multi-agents. Une coordination explicite est naturellement plus adaptée aux robots sophistiqués.
- La coordination implicite comme l'ensemble des techniques employant la dynamique possible au travers de l'interaction entre les robots et l'environnement afin d'atteindre la performance collective souhaitée. Une coordination implicite permet souvent l'émergence de comportements.

La coordination explicite correspond à l'utilisation d'une communication explicite, qui résulte elle-même d'un comportement actif d'un ou plusieurs robots. La coordination implicite correspond à l'utilisation d'une communication implicite, nécessitant des robots percevant d'autres robots et proposant ainsi des actions à ces autres robots. En fusionnant les informations explicites et implicites raisonnablement, la notion de système mixte apparaît. Les communications explicites et

implicites sont présentées en détails dans la section 1.6.

1.6 Communication : explicite vs implicite

La communication, comme moyen de coordination, apparaît souvent comme un comportement rationnel dans les environnements multi-robots. Dans les faits, la communication est un mode d'interaction entre les robots. Par cette interaction, les robots peuvent d'une part partager les informations de position, l'état de l'environnement et les données des capteurs avec les autres dans le système. D'autre part, chaque robot peut individuellement obtenir des informations sur les intentions, les objectifs et les actions de chacun des autres robots.

Cao *et al.* [22] ont classé la structure de communication en trois types selon le mode d'interaction, qui comprend : l'interaction via l'environnement, l'interaction via les perceptions et l'interaction via la communication (explicite). Farinelli *et al.* [36] ont distingué deux types de communication en fonction des modes d'échange entre les robots, qui comprend : la communication directe et indirecte.

La communication explicite se réfère à la nature et aux procédures suivies pour la réalisation des échanges d'informations entre les robots. Ces échanges sont directs : ils sont faits de robot à robot sans passer par un tiers. Ce mode exige souvent un module de communication sur une carte dédiée. Les communications sont des requêtes unicast ou broadcast correspondant à des messages intentionnels. Balch et Arkin [10] ont présenté une stratégie de contrôle de formation basée sur le comportement pour les équipes multi-robots. Afin de communiquer à l'unité au centre de la formation, chaque robot communique sa position à l'autre via un réseau sans fil. Gerkey et Mataric [41] ont présenté un framework général pour la communication inter-robot et l'allocation dynamique de tâches pour les équipes de robots mobiles coopératifs. Klavins [66] a introduit une notion de complexité de communication comme moyen d'étude de l'extensibilité des algorithmes multi-robots. Ces travaux proposent de quantifier les besoins de coordination nécessaires. Madhavan *et al.* [75] ont décrit un algorithme basé sur le filtre de Kalman étendu ou *Extended Kalman Filter* (EKF) en anglais pour la localisation d'une équipe de robots à travers un réseau local sans fil. Nett et Schemmer [85] ont présenté une architecture basée sur la communication sans fil pour programmer l'intersection de deux chemins en temps réel. Ces travaux permettent à un ensemble de robots mobiles de traverser une intersection en minimisant le temps d'exécution et en évitant toute collision. Wang *et al.* [125] ont présenté un schéma de communication sans fil de robot en mode ad-hoc pour un grand système composé de plusieurs ensembles de robots mobiles permettant ainsi d'échanger des informations entre ces ensembles. Pimentel et Campos [93] ont développé un algorithme permettant à un groupe de robots coopérants d'établir et de maintenir un réseau sans fil en mode ad-hoc

pour échanger des données liées à une tâche. Rekleitis *et al.* [98] ont examiné le problème de planification de trajectoire multi-robots en définissant la notion de couverture multi-robots avec une communication limitée. Dans leurs travaux, les robots impliqués fonctionnent sous la restriction que la communication entre deux robots est disponible seulement quand les deux robots considérés sont dans leur ligne de mire respective. Berna-Koes *et al.* [15] ont proposé l'utilisation de la langage de communication d'agent ou *Agent Communication Language* (ACL) en anglais et ont décrit une architecture à deux niveaux de communication. Rooker et Birk [100] ont présenté une approche pour l'exploration multi-robots prenant en compte les contraintes des réseaux sans fil. Bicchi *et al.* [16] ont proposé une plate-forme pour le contrôle décentralisé du trafic multi-robots avec des communications sécurisées entre les véhicules en utilisant des clés cryptographiques. Ces travaux permettent également de recomposer les politiques de contrôle de trafic au fil de l'exécution. Bayram et Bozma [13] ont présenté une approche de navigation dynamique pour les robots communiquants via un modèle d'interactions bidirectionnel entre le réseau (établis par communication entre robots) et les états continus du système multi-robots.

La communication implicite se réfère au mode selon lequel un robot reçoit des informations d'autres robots au travers de l'environnement. Ce mode est réalisé en intégrant différents types de capteurs. Il peut faire usage de la notion d'auto-organisation et se divise en deux catégories : la communication implicite active (en accord avec l'interaction via l'environnement) et la communication implicite passive (en fonction des possibilités d'interaction via les perceptions).

La communication implicite active désigne les cas dans lesquels les robots communiquent en recueillant des informations laissées par d'autres robots dans l'environnement. L'utilisation de cette forme de communication est liée au domaine de la biomimétique [9]. La communication implicite passive désigne les cas dans lesquels la perception des changements de l'environnement correspondent à une communication entre les robots. Les perceptions sont propres à chaque robot. Chaque robot calcule les informations de contexte (par exemple, la position et l'attitude) des autres robots et son raisonnement est basé sur les données perçues. Pagello *et al.* [88] ont présenté une approche de coordination pour une équipe de robots footballeurs au travers d'une communication implicite. La coopération entre robots est réalisée par observation des comportements des autres robots. Stulp *et al.* [116] ont présenté un modèle d'évaluation de la coordination implicite pour la coordination de robots footballeurs. Yamada et Saito [131] ont décrit une méthode de sélection d'action adaptative sans communication explicite pour la résolution des problèmes du type boîte-poussant multi-robots dynamique. Pereira *et al.* [92] ont étudié le problème de la coordination multi-robots dans une tâche de transport de ressources en utilisant uniquement l'information de capteurs locaux. Hollinger *et al.* [57] ont présenté un algorithme d'approximation en utilisant une

coordination implicite, cherchant à résoudre le problème de la planification de trajectoire multi-robots dans des environnements intérieurs connus.

L'utilisation d'une communication explicite permet de garantir l'exactitude des informations échangées entre les robots. Toutefois, la charge de communication augmentant avec le nombre de robots impliqués, la saturation des communications peut conduire à une diminution des performances et dans un cas extrême, à l'échec de l'ensemble du système. En utilisant une communication implicite, les informations obtenues par chaque robot sont peu fiables mais la stabilité, la fiabilité et la tolérance aux pannes sont renforcées. L'utilisation conjointe de ces deux méthodes permet un enrichissement mutuel des deux approches et produit une méthode mixte très robuste aux pannes.

1.7 Planification

Le principe de prévoir une séquence d'actions permettant d'atteindre un objectif est appelé planification [102]. D'un point de vue opérationnel, il s'agit d'exécuter ce plan définissant une succession d'actions pour atteindre l'objectif recherché. Dans un système multi-robots, la planification est utilisée pour coordonner des robots leur permettant ainsi d'accomplir une mission. Dans le cadre des systèmes multi-robots, la planification optimale est un problème NP-difficile [32]. Le défi consiste alors à réaliser une planification suffisamment souple pour produire de bonnes solutions, même dans un contexte de suivi d'exécution difficile. La planification multi-robots se divise en deux sous-problèmes (correspondant à deux niveaux de considération du problème) : la planification des tâches et la planification de mouvements. La planification de tâches est conçue pour résoudre le problème de l'affectation de chaque robot à une éventuelle tâche, qui implique la décomposition du problème en tâches, le séquençage et la répartition de ces tâches. La planification de mouvements est conçue pour répondre à une problématique de génération de chemin pour chaque robot. Les notions de géométrie et d'espace de travail apparaissent avec la notion de mouvements et il s'agit alors pour chaque robot, de prendre en compte le chemin des autres robots afin d'éviter les collisions, les possibles situations de congestion et les possibles interblocages.

L'incertitude est une caractéristique importante des problèmes en robotique. Elle découle de l'observabilité partielle de l'environnement et d'une éventuelle modélisation stochastique du résultat des actions de chaque robot [102]. L'omniprésence de l'incertitude rend difficile l'analyse comparative des résultats de deux projets distincts. A l'inverse, la modélisation de robots mobiles sans incertitude facilite la comparaison entre les différentes approches.

1.7.1 Planification de tâches

La planification de tâches multi-robots comprend deux principaux problèmes : la décomposition en tâches et la répartition de ces tâches vers un ensemble de robots. Les résultats de la répartition des tâches sont directement dépendants de la première étape de décomposition.

La décomposition en tâches multi-robots se réfère à la décomposition de la globalité de la mission ou à la décomposition d'une tâche de cette mission. Cette mission ou cette tâche est décomposée en plusieurs sous-tâches simples qui peuvent être réalisées indépendamment par un ou plusieurs robots, selon la caractéristique de la mission, l'exigence recherchée et la répartition des ressources de la mission [24]. Stone et Veloso [114] ont réalisé une collaboration entre agents par introduction de formation qui décompose l'espace de travail des rôles possibles. L'espace de travail est composé d'un ensemble de rôles. A chaque rôle, il correspond à un comportement. Botelho et Alami [17] ont présenté un système décentralisé destiné à décrire et à effectuer une planification de tâches, une décomposition et une répartition des tâches dans les environnements multi-robots. Ce système décentralisé est appelé protocole M+. Ces travaux ont été développés à partir d'un projet européen appelé MARTHA [4]. Zlot et Stentz [141] ont concentré leurs travaux sur les tâches complexes pouvant être décomposées en plusieurs sous-tâches liées entre elles. Ils ont étudié le problème de la décomposition en tâches en généralisant les tâches à des arbres de tâches au sein d'un marché d'échange pair-à-pair. Tang et Parker [118] ont considéré que, dans les approches classiques de l'équipe de travail multi-robots, la décomposition de la tâche d'équipe en sous-tâches est définie par un concepteur humain à l'avance. Leurs travaux décrivent une méthodologie pour synthétiser automatiquement des solutions à partir d'un ensemble hétérogène de tâches avec des équipes multi-robots.

La répartition des tâches dans un système multi-robots est considérée comme une instance du problème d'affectation optimale ou *Optimal Assignment Problem* (PAO) en anglais [43]. Dans sa formulation classique, il est présenté comme suit :

Un nombre arbitraire d'agents est disponible pour exécuter un nombre arbitraire de tâches. Chacun de ces agents peut être affecté à une tâche, en encourageant un coût variant en fonction de l'affectation agent-tâche. Il est nécessaire pour effectuer toutes les tâches d'assigner au moins un agent à chaque tâche. L'objectif est de minimiser le coût total, résultant de la somme des coûts de chaque affectation.

Dans les problèmes connexes à la question de la répartition de tâches, les dynamiques de groupe ont un effet significatif sur le résultat final et la complexité de ces problèmes est NP-difficile [45, 31]. La répartition de tâches dans les systèmes multi-robots consiste en l'attribution de tâches à un ensemble de robots

afin de maximiser un critère prédéfini. Le problème la répartition devient ainsi un problème de décision dynamique variant dans le temps. Ainsi la méthode d'attribution statique des tâches aux robots n'est plus applicable telle que. Une solution alternative consiste à résoudre itérativement le problème d'affectation statique par une résolution successive des questions d'attribution.

Gerkey et Mataric [43, 45] ont donné une analyse formelle et une taxonomie indépendante du domaine des problèmes de répartition des tâches multi-robots, dans laquelle l'ensemble des problèmes sont classés en sept catégories selon la capacité de chaque robot à effectuer les différentes tâches, selon le nombre de robots nécessaires à la réalisation d'une tâche et selon la méthode d'attribution des tâches. Ils ont également analysé et comparé différentes architectures d'affectation par itération : ALLIANCE [90], BLE [127], M+ [17] et des architectures d'affectation en ligne : MURDOCH [42], enchères au premier prix [113] et attribution de rôle dynamique [23].

Le protocole de réseau contractuel ou *Contract Net Protocol* (CNP) en anglais a été développé par Smith [110]. Il vise à atteindre l'attribution des tâches via un contrôle distribué réalisé par un processus de négociation dans un système multi-agents. Aujourd'hui, la plupart des méthodes de résolution des problèmes de la répartition des tâches sont basées sur le modèle CNP. Botelho et Alami [17] ont présenté le système M+. C'est un système multi-robots pour la coopération permettant une répartition des tâches par négociation. Ce système est la première approche basée sur le protocole CNP proposée pour le problème de répartition des tâches. Stentz et Dias [113] ont présenté les idées de l'architecture du marché libre pour la coordination d'un groupe de robots afin d'atteindre un objectif donné. Cette architecture définit des fonctions de coût et des fonctions de revenu au travers des plans possibles pour l'exécution d'une tâche spécifiée. Une tâche est accomplie par division en sous-tâches. Cette solution permet aux robots de soumettre des requêtes et de négocier avec les autres robots pour décider de l'affectation de ces sous-tâches. Le système évolue parallèlement aux actions des robots en coopération ou en concurrence les uns avec les autres. Les actions des robots sont réalisées dans leurs propres intérêts et dans l'intérêt de chaque cercle de négociation. Zlot *et al.* [142] ont appliqué ces idées au problème de cartographie et d'exploration multi-robots. Leurs travaux empruntent l'architecture du marché et cherchent à maximiser des bénéfices tout en minimisant les coûts, visant ainsi à maximiser une fonction d'utilité. La notion de bénéfice est créée par apparition de l'information obtenue par la visite d'un point objectif. Un coût est défini par une fonction de distance estimant la valeur à parcourir pour atteindre l'objectif (en utilisant l'algorithme D*). L'utilité est la différence entre le bénéfice et le coût. Il est intéressant de souligner que l'architecture de coordination basée sur le marché a été appliquée à un scénario d'exploration de la planète Mars [47] (combiné avec l'algorithme D* pour la planification de mouvements de robot). Gerkey

et Matarić [42] ont présenté la première architecture d'affectation en ligne appelée MURDOCH. Cette approche utilise une enchère au premier prix à attribuer chaque tâche. Le mécanisme utilise une vente aux enchères, composée de cinq étapes : l'annonce des tâches, l'évaluation métrique, le dépôt des soumissions, la fermeture des enchères et le suivi des progrès/renouvellements de contrat. Le système MURDOCH a été testé sur deux types de problèmes : le premier est composé de tâches faiblement couplées à long terme et le second est une instance des problèmes du type boîte-poussant composé de tâches étroitement couplées à court terme. Les principales différences entre cette approche basée sur l'enchère et l'approche basée sur le marché sont les suivantes : 1) l'approche basée sur l'enchère utilise une soumission basée sur un coût estimé, tandis que l'approche basée sur le marché prend en compte à la fois un coût et un bénéfice ; 2) l'approche basée sur l'enchère ne permet pas la réaffectation des tâches, tandis que l'approche basée sur le marché permet une réaffectation ultérieure.

Dahl *et al.* [31] ont présenté un algorithme pour la répartition des tâches dans des groupes de robots homogènes, basé sur l'utilisation des chaînes de vacance avec une stratégie de répartition des ressources communes. Dans la structure sociale définie par les chaînes de vacance, une notion de récompense permet de distribuer les tâches entre les robots. Cette approche est inspirée des relations observées dans les sociétés humaines et animales. Cet algorithme utilise une sélection de tâches (la sélection utilisée est dite locale, dans la mesure où elle est valide pour un ensemble de robots pouvant être un sous ensemble des robots du système), avec un apprentissage par renforcement pour l'estimation de l'utilité des tâches. Hanna [53] a proposé une approche permettant aux robots de prendre en compte l'incertitude de l'exécution des tâches. Leurs travaux décomposent le problème de répartition des tâches en deux étapes : dans la première étape, chaque robot sélectionne ses propres tâches sur une estimation du bénéfice attendu en utilisant un processus de décision markovien ou *Markov Decision Process* (MDP) en anglais ; dans la deuxième étape, un mécanisme basé sur la notion d'enchère est appliqué. Michael *et al.* [81] ont proposé un algorithme de coordination distribué basé sur la notion de marché dans lequel les agents sont capables de soumettre des requêtes d'attribution de tâches avec l'hypothèse que les agents ont une connaissance de toutes les tâches en cours d'enchère, tout en connaissant le nombre maximum d'agents pouvant être affecté à chaque tâche. Chaque enchère est effectuée parmi les groupes voisins des agents et ne nécessite qu'une communication locale. L'algorithme proposé est utilisé pour résoudre le problème du contrôle de formation des systèmes multi-robots. Shiroma et Campos [109] ont proposé un framework appelé CoMutaR, conçu pour régler à la fois les problèmes de répartition de tâches et les problèmes de coordination dans les systèmes multi-robots. Ce framework permet à un seul robot d'effectuer plusieurs tâches simultanément en vérifiant les informations liées aux tâches simultanément à leur mise en œuvre. Ce frame-

TABLE 1.1 – Comparaison entre trois méthodes de répartition de tâches

	Approche basée sur le marché	Approche basée sur l'enchère	Approche basée sur l'échange
Modèle de négociation (communication)	publier/ souscrire	publier/ souscrire	demander/ répartir
Capacité de répartition des tâches par itération	une seule tâche	une seule tâche	plusieurs tâches
Détermination de rôle des robots	volontaire	volontaire	négociation
Algorithme de répartition des tâches	algorithme glouton	algorithme glouton	algorithme glouton
Considération d'utilité	coût et bénéfice	coût	coût
Réaffectation de tâche	autorisée	non autorisée	autorisée
Complexité de communication	$O(1)$ /soumissionnaire $O(n)$ /commissaire-preneur	$O(1)$ /soumissionnaire $O(n)$ /commissaire-preneur	$O(1)$ /acheteur $O(n)$ /vendeur
Complexité de calcul	$O(n)$	$O(n)$	$O(n)$

work a été testé et évalué en simulation dans des problèmes de transport d'objets, de surveillance de zone et du type boîte-poussant multi-robots. Wawerla et Vaughan [126] ont présenté deux stratégies d'allocation de tâches pour un système de transport multi-robots. L'une est basée sur un planificateur centralisé, utilisant les connaissances spécifiques du domaine du problème, pour trouver des solutions d'affectation en temps linéaire en fonction du nombre de tâches. La seconde stratégie permet aux robots de prendre des décisions individuelles d'attribution des tâches en utilisant une information locale obtenue par une communication minimaliste (à une seule valeur). Nous avons développé une approche décentralisée, légère et robuste, basée sur des règles commerciales pour le problème d'exploration multi-robots coordonnée [134]. Cette approche est conçue pour simuler la relation entre acheteurs et vendeurs dans un système d'affaires commerciales, à réaliser une répartition dynamique de tâches en utilisant un mécanisme d'*offre non sollicitée*. Une comparaison entre l'approche basée sur le marché, l'approche basée sur l'enchère et notre approche basée sur l'échange est présentée dans le tableau 1.1.

La répartition des tâches pour les systèmes hétérogènes et homogènes est différente. Dans les systèmes hétérogènes, la répartition des tâches est déterminée par les capacités individuelles des robots. Parker [91] a introduit le concept de

couverture de tâches, qui mesure la capacité d'un membre d'une équipe à réaliser une tâche donnée. Cette notion de couverture peut être utilisée comme indice d'organisation d'une équipe de robots pour effectuer une mission à partir d'une réserve disponible de robots hétérogènes. La couverture de tâches atteint une valeur maximale dans les équipes homogènes et diminue à mesure que les équipes deviennent plus hétérogènes. Dans les systèmes homogènes, les agents ont besoin de se différencier dans les rôles auxquels ils aspirent, par symétrie à leur homogénéité au regard des interactions possibles entre les robots [134]. Dans le cas simple d'un travail en binôme, il sera naturel de penser à un acteur avec un assistant. Dans [133, 134], une mission d'exploration de régions inconnues est décomposée en plusieurs sous-tâches pouvant être identifiées par la construction d'une grille contenant des informations topologiques. Des variations apparaissent dans les résultats en fonction des politiques d'assignation choisies au début de chaque mission.

1.7.2 Planification de mouvements

Le problème de la planification de mouvements correspond à la question de la génération d'une série de mouvements continus d'une configuration initiale à une configuration finale dans l'espace des configurations²³ (*i.e.* l'espace des états du robot défini par l'emplacement, l'orientation et les angles des articulations [102]), tout en évitant les collisions avec les obstacles. Le mouvement est représenté comme un trajet dans l'espace de configuration. La planification de mouvements est éminemment nécessaire pour un robot mobile, puisque par définition, un robot accomplit des tâches en se déplaçant dans le monde réel [71].

La planification de mouvements multi-robots tient compte non seulement des obstacles (statiques ou dynamiques) de l'environnement, mais aussi des interférences possibles entre les robots. Répartis en équipe et destinés à effectuer des tâches indépendantes dans un espace de travail partagé, les robots deviennent mutuellement des obstacles mobiles pour chacun. Ainsi, chaque robot doit prendre en compte le mouvement des autres robots pour mener à bien sa mission. Un exemple bien étudié est le problème du partage de l'espace multi-robots (cf. section 1.4). Par conséquent, les environnements multi-robots sont des environnements dyna-

23. En robotique, une configuration est une position accessible par un effecteur du robot. Un ensemble d'effecteurs donnera un ensemble de positions, toujours appelé configuration. Dans la pratique, il est possible d'associer nombre d'effecteurs et dimension de l'espace des configurations. Pour les robots non-redondants, le nombre d'effecteurs définit le nombre de degré de liberté et définit donc la dimension de l'espace des configurations. Pour les robots redondants, le nombre d'effecteurs est supérieur à la dimension de l'espace des configurations. Dans le cas de robots déformables, la correspondance bidirectionnelle entre encombrement dans l'espace physique et position dans l'espace des configurations est perdue.

miques dans lesquels la planification de mouvements est intrinsèquement difficile. Même pour un cas simple à deux dimensions (les robots évoluant selon deux coordonnées x et y dans le plan), le problème de planification de mouvements est NP-difficile et n'est pas résoluble en temps polynomial [52].

Les principales approches de planification de mouvements sont une extension de travaux précédents obtenus dans le cas des systèmes mono-robot. Bennewitz *et al.* [14] ont présenté une approche basée sur l'algorithme de planification de chemin A* [54], optimisant les priorités pour la méthode planification de chemin par découplage des sous-tâches et par définition d'une valeur de priorité à chaque sous-tâche, pour des groupes de robots mobiles. L'approche proposée est une méthode aléatoire qui redéfinit itérativement l'organisation des robots pour trouver une séquence correspondant au plan minimisant les longueurs de l'ensemble des chemins à parcourir pour l'ensemble des robots. Guo et Parker [52] ont proposé une approche basée sur l'algorithme de planification de chemin D* [112]. Cette approche contient deux modules : planification de chemin et planification en vitesse. Chaque robot planifie son propre chemin indépendamment (*i.e.* le système contient donc N modules de planification de chemin pour N robots), puis un diagramme de coordination est construit sur la base des possibles situations de collision entre tous les chemins de robots (*i.e.* le module de planification en vitesse permet de varier la vitesse de chaque robot à proximité de chaque situation de collision afin de supprimer ces situations). Une approche nommée champ de potentiel artificiel ou *Artificial Potential Field* (APF) en anglais est largement utilisée pour le contrôle de formation des systèmes multi-robots, lié au problème de planification de mouvements. Tanner et Kumar [119] ont présenté une stratégie à convergence asymptotique quasi globale pour des ensembles de robots dans des formations choisies.

Dans le cadre des méthodes de décomposition de l'espace des configurations, deux grandes familles d'approches sont connues sous le nom de décomposition en cellules et de squelettisation. Elles contribuent toutes les deux à réduire un problème de planification de mouvements continus dans un espace en un problème de recherche de graphe, par discrétisation et identification d'états canoniques et de chemins dans l'espace libre [102]. Ces méthodes placent les problèmes de planification dans des espaces géométriques discrets. Par association d'une valeur de surface à chaque cellule, la méthode de décomposition en cellules permet un calcul de couverture et permet ainsi l'obtention d'une fonction d'évaluation d'une action dans sa contribution à une mission globale dans son ensemble. Pour une mission composée de plusieurs actions, cette notion permet ainsi de mesurer la contribution de chaque action à l'ensemble de la mission. Hazon *et al.* [56] ont présenté un algorithme ORMSTC (*Online Robust Multi-Robot Spanning Tree Coverage* en anglais), basé sur une décomposition approximative en cellules permettant de modéliser un environnement inconnu. L'algorithme s'exécute en ligne, est

présenté comme robuste et complet. Les robots n'ont pas de connaissance a priori de leur zone de travail. Les robots continuent à se déplacer tant que la couverture complète n'est pas atteinte.

Une méthode classique de squelettisation, le *diagramme de Voronoï*, est couramment employée en robotique dans la résolution des problèmes d'exploration. Wurm *et al.* [130] ont présenté une stratégie pour la coordination d'une équipe de robots explorateurs, utilisant le diagramme de Voronoï pour segmenter la carte de l'environnement en un squelette permettant de définir des chemins. Une stratégie est utilisée pour définir des points clés [120]. Ces points clés sont utilisés comme objectifs à assigner à l'ensemble des robots pour réaliser l'exploration de l'environnement. Les points clés sont ici vus comme des tâches à répartir. Ils sont les noeuds du diagramme de Voronoï à laquelle la distance à l'obstacle le plus proche dans le plan. Une alternative aux diagrammes de Voronoï est la feuille de route probabiliste ou *Probabilistic RoadMap* (PRM) en anglais [65], qui a été largement utilisée pour les bras-manipulateurs [106, 104]. En termes d'application dans les systèmes de robots mobiles, Švestka et Overmars [117] ont présenté une approche basée sur l'approche PRM pour la planification de mouvements de plusieurs robots mobiles non-holonomes de type voiture dans le même espace de travail. L'espace de travail est ici statique. Les feuilles de route de chaque robot sont le résultat de composition de feuilles de route sous-jacentes calculées pour un unique robot dans un espace. Le chemin de chaque robot est calculé par apprentissage probabiliste. Une autre méthode probabiliste est l'exploration rapide par arbres aléatoires ou *Rapidly-exploring Random Tree* (RRT) en anglais [72]. Bruce et Veloso [18] ont décrit la méthode ERRT basée sur la méthode RRT pour les problèmes de planification de mouvements multi-robots. La méthode ERRT propose un premier mécanisme de cache de points de passage permettant de stocker la connaissance de l'endroit où un plan pourrait à nouveau être utile dans un avenir proche et un second mécanisme de recherche adaptative pour trouver les plus courts chemins.

Les approches PRM et RRT sont les principales méthodes de la littérature pour la planification de mouvements dans des espaces de configurations de grande dimension. Ces méthodes trouvent leur intérêt dans leur complexité sub-linéaire en la dimension des problèmes. Les méthodes de type recherche heuristique incrémentale (comme A* et D*) et de type recherche par carte topologique (comme *diagramme de Voronoï*), ont un temps d'exécution qui s'accroît exponentiellement avec la dimension de l'espace. Les approches PRM et RRT ont également l'avantage d'être simple à utiliser, de part la simplicité des composants nécessaires et de part leur généralité. Les méthodes basées sur un échantillonnage de l'espace des configuration sont dites à complétude probabiliste²⁴ et ne peuvent pas dé-

24. Pour un problème donné, ayant au moins une solution dans un espace donné, la probabilité

terminer si de l'existence d'une solution. Schwager *et al.* [107] ont introduit un framework d'optimisation pour le déploiement multi-robots. Ce framework rassemble plusieurs stratégies existantes. Il comprend des approches géométrique, des approches probabilistes et des approches basées sur l'utilisation des champs de potentiel.

Dans les systèmes multi-robots à communication limitée, la vitesse de calcul et la complexité de l'environnement, impliquent des approches différentes des problèmes de planification de mouvements. Ces approches concernent des problématiques relatives aux réseaux de capteurs sans fil ou *Wireless Sensor Network* (WSN) en anglais. Clark *et al.* [29] ont présenté un framework pour la planification de mouvements de robots mobiles. Ce framework utilise des réseaux dynamiques, dotés des capacités suivantes :

- La capacité de détecter une autre entité, d'établir avec cette dernière une connexion. L'établissement d'une connexion avec une autre entité permet de définir la notion de réseau, dans laquelle il sera possible d'ajouter d'autres entités et ainsi de définir des réseaux locaux.
- La capacité de partager avec une autre entité, son modèle de l'environnement et ses objectifs dans cet environnement. Cette capacité implique la possibilité de définir un environnement considéré comme commun au moment du partage. L'information est ensuite considérée comme valide pendant un laps de temps fixé.
- La capacité de définir, à l'aide d'un planificateur centralisé pour un réseau local, l'ensemble de trajectoires des entités de ce réseau local. La coordination des trajectoires est réalisée par le planificateur et tout ajout de nouvelle entité dans le réseau local implique la suppression des trajectoires précédentes et la génération d'un nouvel ensemble de trajectoires.

1.8 Prise de décision : centralisée vs décentralisée

La prise de décision est un processus cognitif aboutissant à la sélection d'un plan d'actions parmi les scénarios possibles. Chaque processus de décision produit un choix définitif. Dans le cas des systèmes multi-robots, la prise de décision guidée par la planification est centralisée ou décentralisée.

Dans des architectures centralisées, un agent de contrôle est responsable des informations globales concernant l'environnement ainsi que les robots. Cet agent de contrôle communique avec tous les robots pour leur permettre de partager des informations. L'avantage de cette architecture centralisée est de fournir une vue globale du monde dans lequel les robots évoluent et donc de pouvoir produire des

que le planificateur trouve une solution tend vers 1 quand le temps d'exécution tend vers l'infini.

plans globalement optimaux. Néanmoins, une architecture centralisée possède les désavantages suivants :

- Elle est efficace pour un petit nombre de robots et devient inefficace pour un grand nombre de robots.
- Elle est inadaptée aux environnements dynamiques et aux problèmes dans lesquels apparaissent des défaillances dans les communications.
- Elle produit un système très vulnérable. En cas de dysfonctionnement de l'agent de contrôle central, il est nécessaire de mettre en place un nouvel agent de contrôle central. Alternativement, l'échec de la restauration du contrôle central implique une équipe in-opérationnelle. Dans ce cas, aucune tâche ne peut plus être accomplie.

De nombreux travaux de la littérature cités dans les sections précédentes appartiennent à cette catégorie de système centralisé [29, 118, 119, 130, 73, 133, 135, 136].

Les architectures décentralisées se divisent en deux catégories : les architectures distribuées et les architectures hiérarchiques. Dans les architectures distribuées, il n'existe aucun agent de contrôle central. Tous les robots sont égaux en matière de contrôle et tous sont individuellement autonomes dans leur processus de prise de décision. Dans les architectures hiérarchiques, il existe un ou plusieurs agents de contrôle centraux. Ces agents de contrôle sont locaux à des sous-parties de l'espace de travail (appelées clusters). Les agents de contrôle sont donc responsables de l'organisation de chaque cluster et il n'y a qu'un agent de contrôle par cluster. L'architecture hiérarchique est une architecture hybride, intermédiaire entre l'architecture centralisée et l'architecture distribuée. Les architectures décentralisées répondent mieux à la problématique des environnements inconnus ou variables et ont de meilleurs résultats en terme de fiabilité, de flexibilité, d'adaptabilité et de robustesse. Néanmoins, les solutions proposées par ce type d'architecture sont souvent sous-optimales. Feddema *et al.* [37] se sont concentrés sur l'accessibilité d'entrée/sortie, l'observabilité structurelle, la contrôlabilité du système, ainsi que la stabilité de connexion pour les système de contrôle décentralisés. Ils ont montré comment ces principes sont applicables au contrôle de formation multi-robots et à la surveillance de périmètre. M+ [17] est un exemple type de système multi-robots utilisant une architecture distribuée dans laquelle chaque robot possède sa propre connaissance du monde et peut décider de ses actions futures en tenant compte de son contexte actuel et de la tâche à accomplir. Chaque robot connaît ses propres capacités, possède une vue locale du monde au travers de ses capteurs et une vue des capacités des autres robots au travers d'un processus de négociation. CEBOT [39] est un exemple type de système multi-robots utilisant une architecture hiérarchique dans laquelle les cellules (*i.e.* les robots) sont physiquement couplés avec d'autres cellules pour former des clusters. Au sein des clusters, des cellules maîtresses sont sélectionnées afin de co-

ordonner l'exécution des tâches. De nombreux travaux de la littérature cités dans les sections précédentes appartiennent à cette catégorie de système décentralisé [90, 14, 60, 131, 142, 42, 101, 123, 16, 76, 134].

1.9 Positionnement de nos travaux

L'objectif de cette thèse est l'étude de la planification de tâches et de la planification de mouvements pour un ensemble de robots mobiles évoluant dans un espace commun. L'étude réalisée dans nos travaux est centré sur la réduction des possibles situations d'attente dans les scénarios d'exécution des missions multi-robots.

Comme le présente ce chapitre, les problèmes de conflit de ressources apparaissent comme pierre angulaire de la coordination multi-robots. Dans le même temps, ce type de problème apparaît lorsque plusieurs robots évoluent dans un même espace. Dans une coordination réfléchie comme dans une coordination réactive, force est de constater l'apparition de situations d'attente. Ces situations d'attente, voulues ou non voulues, sont le résultat de manoeuvres d'évitement de collision, de congestion de l'espace et d'interblocage entre robots.

L'étude des travaux présents dans le domaine montre que ni la décomposition en sous-tâches, ni les approches de coordination délibérative, ni les approches de coordination dynamique ne permettent d'éviter ces situations. Les travaux de coordination implicite ou explicite permettent la gestion des situations d'attente dans le cadre des architectures centralisées ou décentralisées. Les travaux de la littérature comportent de nombreuses améliorations de localisation, de commande et d'achèvement des missions dans le cadre de l'utilisation de plusieurs robots. Il reste que l'insertion de situations d'attente demeure une solution classique des cas extrêmes de coordination. Nous proposons de minimiser les occurrences de situation d'attente dans les plans multi-robots dans la proposition de quatre nouvelles approches de planification dédiées aux systèmes multi-robots : une approche contrarienne centrée sur les échanges [134], une approche heuristique utilisant un modèle empirique des résultats escomptés [136], une approche par échantillonnage dédié à la décision [133] et une approche par échantillonnage adaptatif [135].

Le chapitre 2 présente nos deux premières contributions concernant la planification de tâches [134, 137, 136]. Elles se composent d'une approche contrarienne et d'une approche heuristique. Les travaux présentés dans l'approche contrarienne propose une solution robuste aux problèmes d'exploration. Cette solution est composée d'un algorithme de répartition des rôles impliquant des comportements asymétriques complémentaires. Les robots identifiés comme vendeurs ou acheteurs dialoguent pour établir une répartition des tâches de moindre coût. Les travaux

présentés dans l'approche heuristique propose une solution à temps d'exécution réduit et basse consommation d'énergie pour des problèmes de transport de marchandises multi-robots. L'approche proposée est guidée par une estimation des résultats escomptés et minimise le temps total d'exécution des missions.

Le chapitre 3 présente nos deux contributions suivantes concernant la planification de mouvements [133, 135]. Elles se composent d'une approche par échantillonnage dédié à la décision et d'une approche par échantillonnage adaptatif probabiliste. Les travaux présentés dans notre approche destinée à la décision regroupent un ensemble de composants existants dans l'état de l'art pour les agréger dans une approche symbiotique intelligente. Ils proposent l'utilisation d'un échantillonnage réalisé dans un espace des configurations proches de positions clés identifiées à l'aide d'un *diagramme de Voronoï*. Cette solution tend vers la génération de chemins différents pour un ensemble de robots. Ces travaux proposent une décomposition des cas de situation d'attente et réalise une planification minimisant ces situations dans des cas d'exploration multi-robots. Les travaux présentés dans notre approche par échantillonnage adaptatif probabiliste identifient les situations d'attente dans des scénarios de référence de transport de marchandises. La solution proposée est un planificateur probabiliste multi-requêtes. Après un pré-calcul jugé représentatif de l'espace des configurations partagée par les robots, une phase de rétraction permet d'identifier des positions caractérisantes des possibilités de coordination dans les différentes sous-parties de l'espace des configurations. La solution proposée permet de minimiser l'apparition des situations d'attente lors de l'exécution des plans de chaque robot.

Le chapitre 4 présente les expérimentations menées pour valider nos contributions. Ces expérimentations sont réalisées avec le simulateur *Player/Stage*. Les expériences impliquent des robots localisées par un module de localisation *SLAM* ou *GPS* guidant chaque robot avec ses propres perceptions. Les résultats montrent la robustesse et la simplicité des approches proposées dans des espaces intérieurs structurés et non structurés. Les expériences sont réalisées avec des groupes de robots de type *Pioneer 2-DX* avec télémètre laser et pince de saisie des marchandises pour les scénarios de transport. Les résultats montrent le temps d'exécution des missions et la consommation d'énergie résultante.

Le chapitre 5 présente une conclusion de l'ensemble de nos contributions et les différentes perspectives faisant suite à nos travaux.

Chapitre 2

Contributions à la planification de tâches

Dans ce chapitre, nous présentons nos travaux sur la planification de tâches multi-robots dans le contexte des problèmes d'exploration et de transport. Nous commençons par présenter une nouvelle approche de répartition des tâches décentralisée basée sur l'utilisation de règles commerciales. Dans les systèmes décentralisés, les robots dotés d'une communication à portée limitée, peuvent prendre des décisions en fonction d'informations locales. Contrairement aux approches précédentes, notre approche basée sur l'échange est conçue pour simuler les relations entre acheteurs et vendeurs dans un système commercial, afin d'obtenir une répartition dynamique de tâches en utilisant un mécanisme d'*offre non sollicitée*. Cette approche est mise en œuvre et évaluée dans une simulation d'exploration. Les résultats expérimentaux démontrent les résultats de notre approche dans une démarche scientifique comparative avec les approches précédentes.

Dans le contexte des problèmes de transport des systèmes multi-robots, l'objectif est de minimiser le temps total de transport tout en conservant une faible consommation énergétique des robots avec des garanties de sécurité et de qualité permettant d'assurer un transport acceptable¹ des marchandises. La question fondamentale à résoudre est alors de savoir quelle tâche assigner avec quel robot à chaque instant. Nous présentons une nouvelle solution en utilisant une stratégie de planification heuristique basée sur un modèle empirique. Contrairement aux approches précédentes, cette stratégie est conçue pour planifier la tâche de transport pour chaque robot en estimant le taux de production de marchandises sur la base de la coordination actuellement en cours dans le système multi-robots. Notre approche est mise en œuvre et évaluée en simulation. Les résultats expérimentaux

1. Pour transport acceptable, nous entendons sans détérioration des marchandises et du service fourni.

montrent que le délai d'achèvement de la mission transport dans son ensemble est inférieur tout en maintenant un faible niveau de consommation d'énergie.

Ce chapitre est organisé comme suit : la section 2.1 présente notre approche contrarienne centrée sur les échanges ; la section 2.2 présente notre approche heuristique basée sur un modèle empirique ; le chapitre se conclut dans la section 2.3.

2.1 Approche contrarienne centrée sur les échanges

Cette section² présente une nouvelle approche pour la répartition des tâches dans un système multi-robots décentralisé. Cette répartition dynamique est basée sur des règles commerciales appliquées à des groupes de robots autonomes. Cette approche est conçue pour simuler une relation entre acheteurs et vendeurs dans un système commercial. Le robot qui joue le rôle de l'acheteur demande la tâche et le robot qui joue le rôle du vendeur est responsable de la répartition des tâches. En raison de son contexte, l'approche proposée repose sur les trois caractéristiques suivantes :

- l'utilisabilité, les robots sont capables de mettre en œuvre l'approche proposée au travers de programmes simples, utilisables sur des architectures à faibles capacités de calculs ;
- l'efficacité, les robots sont capables de gérer la répartition des tâches à l'aide d'un système de communication leur permettant de dialoguer ;
- la robustesse³, les robots sont capables de prendre leurs propres décisions avec ou sans information venant d'autres robots.

Les approches classiques sont basées sur la notion de marché [113] ou sur la notion d'enchère [42] et utilisent des mécanismes d'enchère et de soumission⁴. Contrairement à ces approches, nous utilisons des notions de demande et de répartition. Un robot, appelé acheteur, fait une demande de répartition des tâches à un autre robot, appelé vendeur (ce que nous appelons un mécanisme d'*offre non sollicitée*). Après une période de temps fixée, le vendeur analyse les demandes reçues et affecte les tâches aux acheteurs selon un ensemble de règles. Ce mécanisme permet à notre approche d'assigner plusieurs tâches à plusieurs robots simultanément⁵.

2. Cette partie de nos travaux a été publiée dans *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2011)*.

3. L'utilisation des approches basées sur le contrôle adaptatif permet d'avoir une bonne robustesse [49].

4. Ces mécanismes sont inspirés des principes de microéconomie, dont [94] présente une synthèse des modèles empiriques et théoriques.

5. Par opposition aux approches basées sur le marché ou basées sur l'enchère, commençant par la diffusion des tâches à réaliser, notre approche basée sur l'échange présente le problème de

Dans cette section, nous considérons le problème d'exploration multi-robots comme scénario de test de référence. Plus précisément, le problème posé correspond ici à une exploration en environnement structuré inconnu⁶ pour une équipe homogène de robots mobiles. Ce problème trouve des applications dans le domaine du sauvetage de personnes dans les environnements dangereux tels que les bâtiments faisant face à des incendies ou des explosions, ou encore dans des situations post-cataclysmiques [61, 8, 12]. L'objectif devient dans ce cas d'explorer l'environnement dans sa totalité, en minimisant le temps nécessaire pour compléter cette exploration.

En comparaison avec les systèmes mono-robot, l'utilisation des systèmes multi-robots présente plusieurs avantages (cf. section 1.2). Dans le cas présent d'une mission d'exploration, l'utilisation d'un système multi-robots implique une réduction du temps d'accomplissement de la mission, dont l'issue est fondée sur la synthèse d'une bonne coordination. Sans coordination, l'utilisation d'un système multi-robots peut donner des résultats équivalents, plus mauvais, voir conduire à l'échec de la mission dans des cas extrêmes. La coordination est très liée à la répartition des tâches correspondant à l'ensemble de la mission. L'effet recherché dans la répartition des tâches est une affectation à des différents robots dans des sous-espaces différents. Il existe deux types de mécanismes de répartition de tâches, l'un étant centralisé et le second décentralisé. L'avantage d'un mécanisme centralisé est de pouvoir prétendre à la réalisation d'un plan optimal. Le désavantage d'un tel mécanisme est son inefficacité face aux équipes de grande taille. Comme son nom le suggère, l'utilisation d'un mécanisme décentralisé implique l'absence de planificateur central. Chaque robot d'un système multi-robots utilise des informations locales afin de répondre à leur propre plan d'exécution. Les mécanismes décentralisés possèdent une bonne adaptabilité et une grande robustesse, mais les solutions obtenus sont souvent sous-optimales.

2.1.1 Méthodes existantes

Gerkey et Mataric [42] ont proposé une approche de répartition des tâches basée sur la notion d'enchère pour une coordination multi-robots décentralisée. Le processus de coordination ressemble à une vente aux enchères, se déroulant en cinq étapes : l'annonce d'une ou plusieurs tâches, l'évaluation métrique des différentes tâches annoncées, l'éventuel dépôt de soumission pour une ou plusieurs de tâches annoncées, la fermeture des enchères et le suivi d'exécution (également appelé renouvellement de contrat). Cette stratégie est implémentée et testée dans

répartition des tâches à l'encontre de la croyance générale, en commençant par l'expression des coûts des tâches envisageables.

6. La structure est du type environnement intérieur, c'est à dire défini par un plan, composé de structures géométriques du type segments de droites et courbes.

un système de répartition des tâches appelé MURDOCH. Il est construit sur un modèle de communication publier/abonner, centré sur la gestion des ressources. Cependant, leur méthode nécessite une grande quantité de communications, augmentant ainsi la charge du système et réduisant la robustesse de cette solution.

Zlot *et al.* [142] ont introduit une approche de répartition des tâches basée sur la notion de marché pour l'exploration multi-robots. Cette approche emprunte aux architectures commerciales, la notion de recherche d'un bénéfice maximum, conjointement à la minimisation d'un coût et à la maximisation d'une utilité. L'avantage de cette approche réside dans l'information obtenue par la visite d'un point d'objectif, laquelle est associée à une fonction de coût permettant de bien maîtriser ses conséquences d'application. Le plus courant est d'utiliser la distance restante à parcourir pour atteindre l'objectif (en utilisant l'algorithme D* [112]). L'utilité est la différence entre le bénéfice et le coût. Semblable à la stratégie basée sur la notion d'enchère [42], cette approche utilise également un modèle de vente aux enchères, comprenant un mécanisme d'appels d'offres. Cette approche nécessite également des communications de grande taille.

Yamauchi [132] a présenté une stratégie d'exploration coopérative, décentralisée et tolérante aux pannes multi-robots basée sur le concept de frontières. Les frontières sont des régions situées entre l'espace exploré et l'espace inexploré. Dans cette approche, les robots partagent des informations perceptives, mais la représentation du monde dans lequel chaque robot évolue reste propre à chaque robot. Ainsi chaque robot prend ses propres décisions sur les lieux à explorer, par correspondance avec des tendances de navigation. Chaque fois qu'un robot arrive à proximité d'une nouvelle frontière, il scrute ses capteurs et construit une grille locale représentant les nouvelles informations. Cette grille est intégrée dans la grille globale du robot et est diffusée à destination des autres robots à portée de transmission. Dans des cas défavorables de synchronisation, plusieurs robots sont impliqués dans l'exploration d'une même frontière, réduisant ainsi la performance effective ⁷ d'une exploration multi-robots.

Burgard *et al.* [20] ont conçu un composant de coordination basé sur l'approche de Yamauchi [132]. Le composant applique une méthode probabiliste prenant en compte simultanément le coût nécessaire pour atteindre une frontière et son utilité. Le coût est donné par la distance de déplacement à la frontière et l'utilité est donnée par la taille de la zone inexplorée que le robot peut couvrir avec. Cette zone inexplorée est dépendante de la frontière cible et des capteurs utilisés sur le robot utilisé. Chaque fois qu'une frontière est attribuée à un robot, l'utilité de la zone visible inexplorée à proximité de cette frontière est réduit. Les nouvelles

7. Dans la plupart des cas, la réduction donne ici le même résultat en termes opérationnel et concerne uniquement la minimisation du temps d'exécution de la mission ou la minimisation de la consommation globale d'énergie.

zones inexplorées possibles sont transmises à tous les autres robots, de sorte que le problème de concurrence d'accès sur les zones n'apparaisse que plus difficilement. Toutefois, un agent central est requis pour cette approche. Si l'agent central tombe en panne, l'ensemble du système échouera.

Wurm *et al.* [130] ont proposé une technique pour la coordination d'une équipe de robots explorateurs par segmentation de l'environnement. Au lieu de considérer les frontières entre les zones inexplorées et les zones explorées comme endroits cibles, cette méthode prend en compte la structure de l'environnement et segmente l'environnement déjà exploré sur la base de *diagramme de Voronoï*. Chaque robot est ensuite assigné à un segment distinct avec une tâche d'explorer la zone correspondante. Les conflits entre les robots sont résolus par une *méthode hongroise* (également appelé algorithme de Kuhn-Munkres) [69]. Les résultats obtenus montrent l'amélioration significative produite par cette approche utilisant une segmentation par rapport à une approche standard basée sur la notion de frontière. Ces résultats sont obtenus dans des environnements intérieurs structurés. Cette approche est favorable à des explorations dans des environnements dangereux. Elle nécessite cependant un élément de planification central pour assurer une communication entre les robots coéquipiers, permettant ainsi de gérer la répartition des tâches.

Marjovi *et al.* [78] ont présenté une approche pour l'exploration multi-robots coopérative, centrée sur la problématique de recherche d'incendies dans un environnement inconnu. Cette approche permet également de réaliser la cartographie de l'environnement exploré. Elle vise à minimiser le temps global d'exploration, tout en permettant de localiser les sources d'incendie. Cette approche réalise une exploration décentralisée basée sur la notion de frontière, utilisée ici pour évaluer le rapport entre le coût (en utilisant l'algorithme A^* [54]) et l'utilité de la navigation en des points de passage cible. Le contrôle des robots pour l'évitement d'obstacles est réalisé avec une méthode de champ de potentiels. Les sources d'incendie sont identifiées par triangulation des perceptions visuelles et par identification de triangles ou ensembles de triangles assimilés à des flammes. Les besoins importants en communication limitent les résultats possibles de cette approche en augmentant le nombre de robots.

2.1.2 Contributions

Dans sa forme la plus courante, un système d'affaires est composé d'acheteurs et de vendeurs. La relation d'échanges entre ces deux types d'entités permet :

- aux acheteurs d'utiliser un moyen de paiement (souvent de l'argent) pour acheter des produits ou des services auprès des vendeurs ;
- aux vendeurs de collecter des moyens de paiement en vendant des produits ou des services à des acheteurs.

Dans cette section, nous présentons notre approche, construite sur cette relation d'échanges, dont le modèle est décrit par le tuple suivant :

$$Trade = \{R, M, C, B, P\} \quad (2.1)$$

dans lequel R est l'ensemble des robots mobiles autonomes impliqués et M représente la mission à accomplir. La mission M est composée de plusieurs tâches, $M = \{t_0, t_1, \dots, t_n\}$. C représente l'ensemble des coûts de chaque tâche $C = \{c_0, c_1, \dots, c_n\}$ (un exemple classique de coût est la distance à parcourir pour accomplir une tâche). B représente le bénéfice obtenu pour l'exécution de la mission (un exemple classique de bénéfice est le temps d'exécution nécessaire à l'achèvement de la mission). P représente le plan choisi de répartition des tâches. L'approche contrarienne centrée sur les échanges se décompose en deux étapes principales, qui sont la répartition des rôles et la répartition des tâches.

Répartition des rôles

La répartition des rôles caractérise le problème de l'identification du rôle de chaque robot dans le contexte de la mission : il s'agit ainsi de déterminer quel robot devient acheteur et quel robot devient vendeur. Dans le cas d'une équipe de robots mobiles homogène, les robots n'expriment a priori aucune préférence de rôle. Chaque robot est associé à un numéro unique, permettant de l'identifier. Avec ce numéro unique, chaque robot sans tâche se déclare à tous les autres robots. Par évaluation des numéros reçus dans une période de temps fixée, chaque robot définit le vendeur en choisissant le plus petit identifiant parmi les identifiants reçus (le sien compris). Les autres identifiants sont de fait considérés comme robots acheteurs. Ce mécanisme est présenté en détails dans l'algorithme 2.1.

La différence entre acheteur et vendeur se limite à la position dans l'achèvement d'une mission locale. Un rôle d'acheteur ou de vendeur est temporaire. L'acheteur d'une tâche peut devenir vendeur d'une autre tâche dans la poursuite d'une mission inachevée.

Répartition des tâches

Dans une mission de planification multi-robots regroupant un ensemble de tâches, chacun des robots est initialement considéré comme libre. Chaque robot revient également dans cet état après achèvement de chaque tâche associée à chaque définition de rôle de vendeur ou d'acheteur. Étant libre et face à une requête d'accomplissement d'une ou plusieurs tâches, un robot entre dans une phase de recherche de répartition des tâches.

Les robots acheteurs choisissent des tâches à soumissionner et estiment le coût nécessaire pour achever cette tâche. Cette phase d'acquisition des tâches et d'esti-

Algorithme 2.1 Répartition des rôles pour un robot r

```

1: if est_libre() == true then
2:    $r \leftarrow \text{identifiant}()$ 
3:   diffuser_identifiant()
4:    $temp \leftarrow 0$ 
5:    $R \leftarrow \emptyset$ 
6:   while  $temps < max$  do
7:      $R \leftarrow R \cup \text{reception}()$ 
8:      $temps \leftarrow temps + 1$ 
9:   end while
10:   $robot \leftarrow vendeur$ 
11:  for all  $r' \in R$  do
12:    if  $r' < r$  then
13:       $robot \leftarrow acheteur$ 
14:    break
15:  end if
16: end for
17: end if

```

mation des coûts est étroitement liée au problème de la décomposition des tâches pour un système multi-robots [137]. Dans une mission d'exploration, il s'agit de définir un ensemble de tâches d'exploration de régions inconnues pouvant être identifiées par une carte topologique de l'environnement [133]. Les relations entre les régions inconnues de l'espace à explorer imposent des contraintes dans la répartition et dans l'exécution des tâches d'exploration. Chaque robot doit extraire les *points clés* de l'espace [120]. Dans ce cas, il s'agit de distinguer des couloirs, des portes et des chambres. Le coût métrique est variable, il peut être défini par rapport à la tâche à achever comme la distance parcourue, comme le temps nécessaire ou encore comme l'énergie nécessaire. Dans cette section, nous avons choisi d'utiliser la distance parcourue.

En conséquence, chaque robot définit la distance entre sa position actuelle et la position cible (par exemple, un *point clé*) à l'aide de la distance euclidienne ou à l'aide de l'algorithme *wavefront*⁸ [71] :

$$c_r^t = \sqrt{\sum_{i=1}^2 (r_i - t_i)^2} \quad (2.2)$$

8. Une implémentation de référence de cet algorithme en C++ est disponible sur internet à l'adresse : <http://www.ai.univ-paris8.fr/~yz>

$$c_r^t = \text{wavefront}(\text{position}(r), \text{position}(t)) \quad (2.3)$$

où c_r^t représente le coût c estimé dans l'achèvement de la tâche t pour le robot r . Après estimation de ce coût, les robots acheteur doivent envoyer une demande d'achat au robot vendeur pour chaque nouvelle tâche. Nous avons appelé ce modèle le mécanisme d'*offre non sollicitée*. Le message d'achat contient principalement trois types d'informations :

$$pums_r = \{r, t, c\} \quad (2.4)$$

où $pums_r$ représente le message d'achat $pums$ envoyé par le robot acheteur r . Par ailleurs, le message d'achat contient l'identifiant r du robot acheteur. t désigne l'identifiant de la tâche et c représente le coût estimé de tâche c_r^t . Après le processus de soumission, un robot acheteur (s'il est choisi par le robot vendeur) recevra un message présentant l'opportunité d'obtenir la tâche. Du point de vue d'un robot acheteur, l'acquisition d'une tâche entraîne l'initialisation de son processus de planification de mouvements. Alternativement, s'il n'est pas choisi, il revient au processus de répartition de rôles pour une nouvelle tâche. Un organigramme de prise de décision détaillé pour le robot acheteur est donné dans la figure 2.1.

Pour un robot vendeur, la première étape consiste à sélectionner une tâche et à en estimer son coût. Ensuite, il devrait recueillir les possibles messages d'achat des robots acheteurs dans un délai de soumission fixé. Ce délai passé, le robot vendeur cesse de recevoir les possibles messages d'achat et d'en évaluer les demandes d'achat. Il assigne les tâches en cours d'affectation à chaque robot acheteur conformément à certaines règles. Un organigramme de prise de décision pour le robot vendeur est donné dans la figure 2.2.

La répartition des tâches est effectuée en utilisant un algorithme glouton. Il s'agit de choisir le robot à moindre coût estimé comme cible de la répartition des tâches. Les détails de l'implémentation sont donnés dans l'algorithme 2.2.

Le message de répartition des tâches envoyé au robot acheteur comprend principalement deux types d'informations :

$$tams_r = \{r, t\} \quad (2.5)$$

où $tams_r$ représente le message de répartition des tâches envoyé par le robot vendeur r . Dans le message, r identifie le nombre du robot vendeur et t identifie la tâche concernée. Il est intéressant de noter que le robot vendeur lui-même est également la cible de la répartition des tâches. En outre, le robot ayant terminé une tâche, mettra à jour sa liste des tâches et la diffusera à tous les coéquipiers.

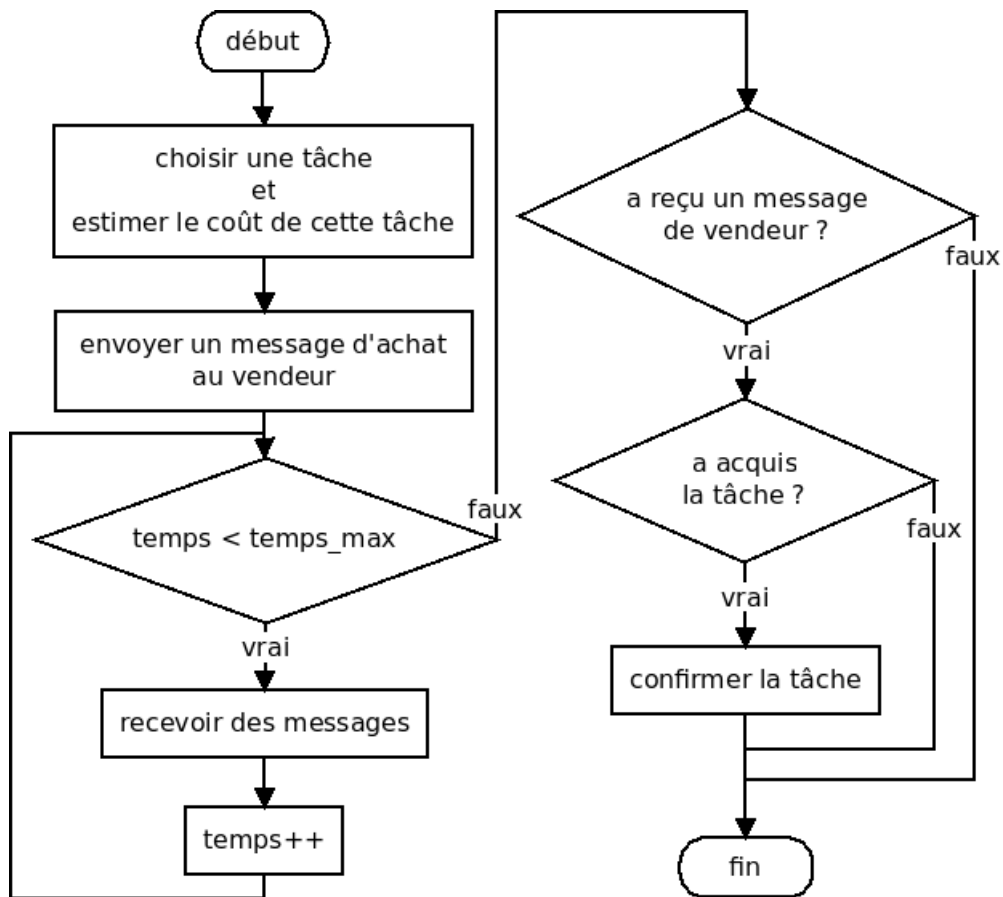


FIGURE 2.1 – Organigramme de prise de décision d'un robot acheteur

Algorithme 2.2 Répartition des tâches pour un robot vendeur

```

1:  $P \leftarrow \emptyset$ 
2: for all  $t \in pums$  do
3:   if  $t' \in P$  and  $t' == t$  then
4:     if  $\text{cout}(t) < \text{cout}(t')$  then
5:        $P \leftarrow \text{mettre\_a\_jour}(P, t')$ 
6:     end if
7:   else
8:      $P \leftarrow \text{ajouter}(P, t)$ 
9:   end if
10: end for

```

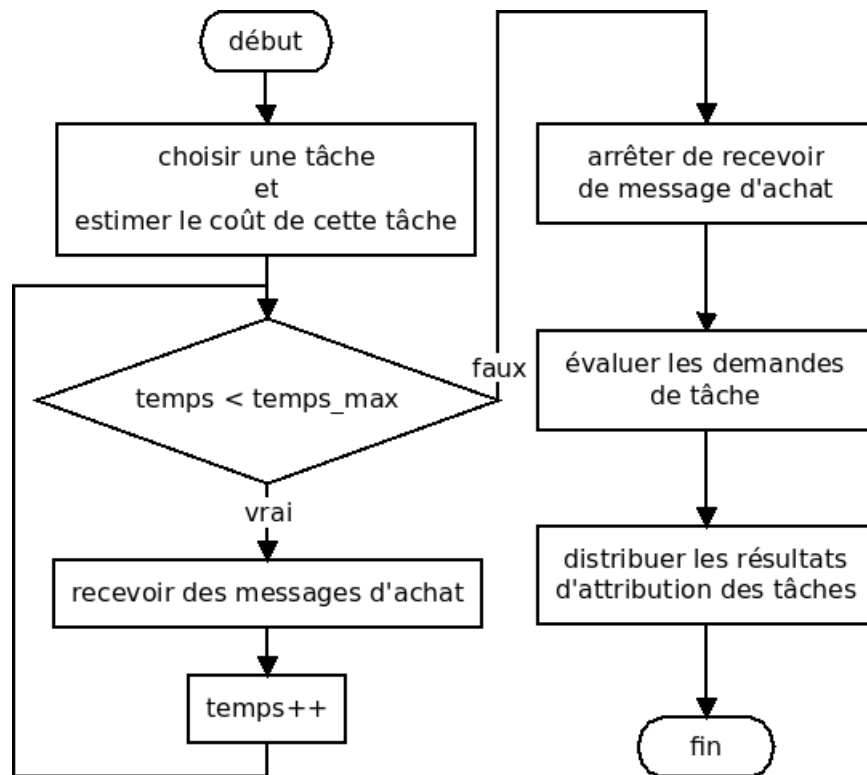


FIGURE 2.2 – Organigramme de prise de décision d'un robot vendeur

2.2 Approche heuristique basée sur un modèle empirique

Cette section⁹ présente une approche heuristique spécialisée dans la résolution des problèmes de transport de marchandises multi-robots avec des marchandises destinées à être transportées de leur lieu de production à un emplacement de consommation. Source et destination sont uniques. L'emplacement de la source et l'emplacement de la destination sont connus et fixés. Le taux de production des marchandises est inconnu. Étant donné que l'emplacement associé à la source ne peut pas stocker plus d'une unité à la fois, tous les robots sont placés dans un dépôt au début de la mission et retournent à ce dépôt dès qu'ils sont libres. La consommation d'un robot est supposée nulle dans la zone de dépôt. En dehors de cette zone, la consommation d'énergie est proportionnelle aux perceptions (dans l'énergie nécessaire à l'alimentation des capteurs et dans la consommation associée aux calculs de traitement des informations de perception) et aux déplacements des ro-

9. Cette partie de nos travaux a été publiée dans *Proceedings of the 6th IEEE International Conference on Intelligent Systems (IS 2012)*.

2.2. APPROCHE HEURISTIQUE BASÉE SUR UN MODÈLE EMPIRIQUE 37

bots (dans l'énergie nécessaire à la commande des actionneurs et dans la consommation associée aux calculs de trajectoire). Par soucis de simplification, le taux de cette consommation reste cependant proportionnel au temps de déplacement des robots. Par exemple, pour la réalisation d'une tâche de transport d'une durée de 10 secondes pour un robot, 10 unités d'énergie sont consommées. L'objectif de la mission est d'achever la mission de transport par exploitation de la collaboration possible entre les différents robots impliqués en réduisant le temps d'exécution et en limitant la consommation d'énergie à un niveau préalablement fixé pour la mission.

Nous proposons dans cette section d'utiliser une combinaison de deux manipulations suivantes : la réduction du temps total d'exécution de la mission et le maintien de la consommation d'énergie à un niveau minimum pour l'ensemble des robots impliqués dans la mission. L'exécution d'une mission et l'exécution d'une action sont ici définies par la modélisation mécanique de ces actions, comprenant les mouvements des actions de préhension et de libération du transport des marchandises tout en considérant la sécurité des marchandises dans leur transport. Les mouvements de préhension et de libération sont modélisés dans un espace de travail réel. La consommation d'énergie nécessaire aux actions des robots est utilisée pour l'évaluation des résultats. Cette consommation concerne des effecteurs de deux types : les premiers destinés aux déplacements dans l'espace de travail et les seconds destinés à la manipulation des marchandises.

L'évaluation des résultats est réalisée par une combinaison du temps d'achèvement de la mission et de la consommation d'énergie. Nous proposons une solution estimant le taux de production des marchandises par une fonction heuristique basée sur un modèle empirique. L'attribution des tâches de transport de marchandises aux robots est réalisée selon des règles établies dans un système de prise de décision centralisé. Notre approche utilise deux métriques évaluant simultanément le temps d'exécution et la consommation d'énergie. Par expérimentation en simulation, nous comparons les résultats de notre approche avec l'approche centralisée *re-planner* pour un système multi-robots. Les résultats de cette comparaison montrent que notre approche réalise une réduction du temps total d'exécution de la mission tout en maintenant un faible niveau de consommation d'énergie.

2.2.1 Méthodes existantes

Alami *et al.* [4] ont présenté une solution pour le contrôle d'une flotte de robots mobiles autonomes qui a été développé, implémenté et validé dans le cadre du projet MARTHA abréviation de *Multiple Autonomous Robots for Transport and Handling Applications*. Ils ont proposé une approche appelée paradigme de planification pour la coopération multi-robots, testée à la fois en simulation et dans le monde réel.

Vaughan *et al.* [123] ont décrit une méthode appelée LOST qui permet à une équipe de robots de naviguer entre des points d'intérêt dans un environnement initialement inconnu en utilisant une traînée de points de repère. Ils ont appliqué cette méthode à un exemple de tâche de transport de ressources, dans lequel plusieurs robots autonomes trouvent et traversent à plusieurs reprises un chemin. Ce chemin est réalisé dans un environnement inconnu entre un domicile connu et une zone de production de ressources dans une position initialement inconnue.

Tang et Parker [118] ont décrit une approche pour synthétiser automatiquement des solutions de planification de tâches pour les équipes hétérogènes. Cette approche, nommée ASyMTRe, repose sur des théories de schéma et des invariants de l'information. Elle permet à l'équipe de robots de connecter dynamiquement des schémas d'exécution au sein et entre les robots, permettant ainsi d'accomplir des tâches. Cette approche est validée dans deux scénarios différents : l'un étant le transport multi-robots et l'autre étant un problème du type boîte-poussant multi-robots.

Dahl *et al.* [31] ont présenté un algorithme pour la répartition des tâches dans des groupes de robots homogènes basé sur les chaînes de vacance. Grâce à des expériences réalisées en simulation, l'algorithme de la chaîne de vacance est démontré plus performant que les algorithmes de répartition de tâches lorsque les robots individuels sont sujets à des distractions, des pannes ou des changements de priorités dans les tâches pendant leur exécution. Ils ont également défini la notion de problème de transport prioritaire ou *Prioritized Transportation Problem* (PTP) en anglais comme une extension du problème de transport classique dans lequel les sources, les destinations et les objectifs du transport, sont divisés en ensembles de tâches de priorités différentes (appelés circuits).

2.2.2 Contributions

Une des principales raisons de la faible efficacité des transports dans un espace connu est le manque de connaissances concernant le taux de production des ressources dans cet espace. Une estimation régulière de ce taux permet de résoudre ce problème, pour lequel une fonction heuristique donne une solution efficace [58, 86, 89]. Le taux de production, noté T_n , reflète le temps écoulé entre la disparition de la n -ième marchandise¹⁰ et l'apparition de la $(n + 1)$ -ème marchandise. T_n est considéré comme une suite de variables aléatoires indépendantes et identiquement distribuées. La loi de probabilité des variables est inconnue, mais peut être déterminée par le test d'hypothèse. La valeur heuristique $h(n)$ est donc obtenue

10. Une marchandise disparaît quand elle est emportée. Cette considération est uniquement virtuelle dans la mesure où les marchandises ne disparaissent pas effectivement et sont en cours de transport.

2.2. APPROCHE HEURISTIQUE BASÉE SUR UN MODÈLE EMPIRIQUE 39

nue par la fonction de distribution suivante :

$$h(n) = E(T'_{n+1} | T_1, \dots, T_n) \quad (2.6)$$

L'équation (2.6) signifie que le temps écoulé depuis la disparition de la n -ième marchandise jusqu'à l'apparition de la $(n + 1)$ -ième marchandise par notre prévision. Étant donné que T_n une distribution uniforme, puis :

$$h(n) = \frac{1}{n} \sum_{i=1}^n T_i \quad (2.7)$$

En effet, l'ensemble du problème est une prise de décision limitée. Pour chaque robot $R_{i(i=0, \dots, m)}$, l'espace de décision pour chaque étape (également notée R_i) est fini. Par exemple :

$$R_i = \{gotoSource, gotoSink, gotoDepot\} \quad (2.8)$$

Ainsi, l'espace de décision pour chaque étape du planificateur centralisé est défini par :

$$R = \prod_{i=0}^m R_i \quad (2.9)$$

Pour N marchandises, l'espace de décision du planificateur centralisé est R^N . Chaque décision $r \in R^N$ est définie par $r(n) \in R(n = 1, \dots, N)$.

L'équation suivante définit le temps d'exécution noté $X(n)$, pour identifier le temps d'exécution de la mission après achèvement du transport de la n -ième marchandise, par :

$$X(n+1) = X(n) + \min_{t \in R} f(t, h(n), r_n) \quad (2.10)$$

Dans l'équation (2.10), f est une fonction permettant de calculer l'intervalle de temps entre le début des tâches de transport de la n -ième et de la $(n + 1)$ -ième marchandise. t représente la prise de décision actuelle. Si f est égale à sa valeur minimum (valeur dépendant de la mission), alors t est égale à $r(n)$. r_n représente l'information de localisation actuelle de chaque robot. Par exemple, t étant également une décision, si la décision prise par un robot est de sortir du dépôt pour se rendre à l'emplacement de l'apparition d'une ressource, alors :

$$f(t, h(n), r_n) = \max(h(n), T_{DepotToSource}) \quad (2.11)$$

L'équation (2.10) utilise un algorithme glouton. Il permet l'obtention d'une durée totale minimum pour accomplir la mission, sans pour autant garantir un faible niveau de la consommation d'énergie. Dans le cas de la figure 2.3, Robot 0

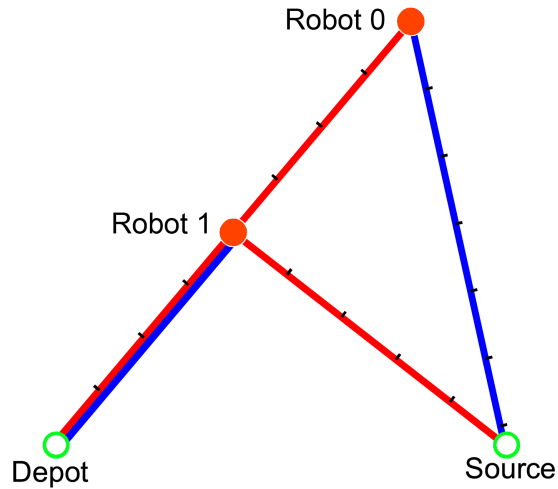


FIGURE 2.3 – Comparaison de deux plans de répartition des tâches

et Robot1 sont deux robots libres (*i.e.* sans tâche). La mission actuelle est le transport d’une marchandise apparue à la zone Source. Les segments rouges et les segments bleus représentent deux plans (le premier nommé plan rouge et le second plan bleu). Chaque plan (rouge et bleu) est composé d’un mouvement pour chaque robot (soit un total de deux mouvements par plan). Les deux plans sont concurrents, dans le sens où il s’agit ici de choisir entre le plan rouge et le plan bleu pour accomplir la mission. Le plan rouge envoie Robot0 au Dépôt et Robot1 à la Source. Le plan bleu envoie Robot0 à la Source et Robot1 au Dépôt. En utilisant l’équation (2.10), le planificateur choisit le plan rouge, en raison de la proximité entre Robot1 et la Source. Pourtant l’évaluation des consommations d’énergie des plans rouge et bleu sont respectivement de 13.0 et de 10.4 (ce qui suggère le choix du plan bleu).

Ainsi, l’équation (2.10) ne garantit pas l’obtention d’une solution optimale. Pour palier à ce problème, nous proposons l’utilisation d’une approche heuristique basée sur un modèle empirique de la consommation d’énergie permettant de remplacer la fonction $\min_{t \in R} f(t, h(n), r_n)$. Ce modèle empirique prend en compte le temps d’exécution des tâches composant la mission. La répartition des tâches découlant de ce modèle est présentée dans l’algorithme 2.3.

Dans l’algorithme 2.3, la ligne 5 réalise un test d’appartenance à un intervalle, qui s’ajuste itérativement. Cet ajustement est fonction des caractéristiques des robots mobiles et également des caractéristiques de l’environnement de travail choisi. $T_{toSource}(i)$ indique le temps nécessaire pour le robot i pour se rendre de son emplacement actuel à l’emplacement de la source. Dans l’algorithme 2.3, $T_{toSource}(i)$ et $T_{DepotToSource}$ sont deux valeurs estimées pendant l’exécution de la

Algorithme 2.3 Répartition heuristique des tâches

```

1: if transportee ( $n$ -ième marchandise) == true then
2:    $tache\_acceptee \leftarrow$  false
3:    $h(n) \leftarrow$  équation (2.7)
4:   for all au_depot ( $R_i$ ) == false and est_livreur ( $R_i$ ) == false do
5:     if  $h(n) < T_{toSource}(i)$  and  $h(n) > \frac{1}{2}T_{toSource}(i)$  then
6:        $tache\_acceptee \leftarrow$  true
7:       gotoSource ( $R_i$ )
8:       break
9:     end if
10:  end for
11:  if  $tache\_acceptee ==$  false then
12:    for all au_depot ( $R_i$ ) == true do
13:       $T_{wait} \leftarrow \max(0, h(n) - T_{DepotToSource})$ 
14:       $tache\_programmee(R_i, T_{wait})$ 
15:      break
16:    end for
17:  end if
18: end if

```

mission. $T'_{DepotToSource}$ et $T'_{SourceToSink}$ sont des valeurs estimées une seule fois. Ces deux valeurs sont des valeurs de référence, inconnue avant l'exécution de la mission, fixée à la suite de la première livraison et utilisée par la suite dans le calcul de $T_{toSource}(i)$ et $T_{DepotToSource}$. Avant la première livraison, aucune décision n'est nécessaire, ces valeurs ne sont pas utilisées. A la suite de la première livraison, elles sont fixées aux temps constatés lors de cette première itération. L'achèvement de la première tâche fixe les quatre valeurs $T'_{DepotToSource}$, $T'_{SourceToSink}$, $T_{toSource}(i)$ et $T_{DepotToSource}$. L'achèvement de chacune des tâches suivantes met à jour la valeur $T_{toSource}(i)$. L'achèvement d'une tâche pour un robot est atteinte à la livraison de chaque marchandise dans la zone destination. Une tâche peut commencer par un position au dépôt ou une position en cours de retour au dépôt. Après exécution d'une tâche pour le robot i , $T_{toSource}(i)$ est égal au temps nécessaire pour le robot i pour rejoindre la source des marchandises.

Le tableau 2.1 présente une comparaison entre notre approche heuristique et l'approche classique utilisant l'algorithme glouton nommé *re-planner* [126]. Les deux approches sont centralisées. Des différences apparaissent dans les méthodes d'évaluation pour le choix du robot i pour l'exécution d'une tâche, dans l'expression de résultats meilleurs en temps ou en énergie. La complexité est exprimée ici en fonction du nombre de robots. Le point fort de notre approche utilisant une fonction heuristique permet d'aboutir à des résultats prenant en consi-

TABLE 2.1 – Comparaison entre deux méthodes de répartition de tâches

	Approche re-planifier	Approche heuristique
Décision	centralisée	centralisée
Evaluation	algorithme glouton	fonction heuristique
Résultats	en temps ou en énergie	en temps et en énergie
Complexité	$O(n)$	$O(n)$

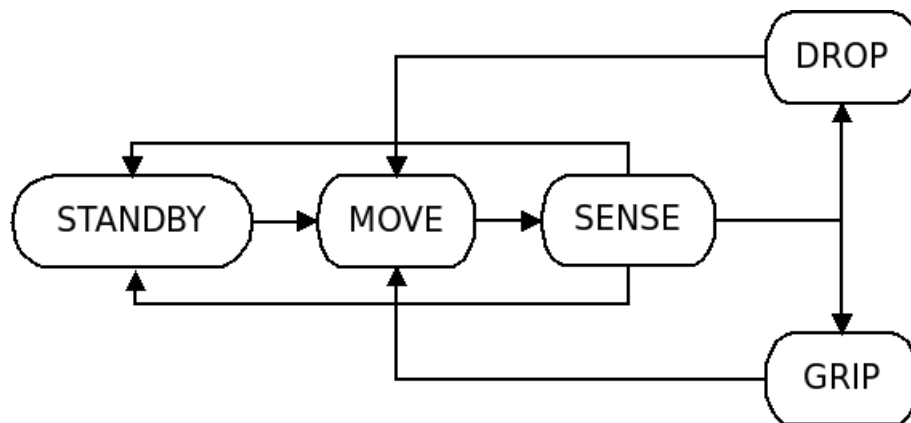


FIGURE 2.4 – Relations entre comportements pour la commande d'un robot

dération temps et énergie.

Le contrôle du comportement de chaque robot est défini par l'utilisation d'une technique de contrôle réactif pour l'architecture logicielle robotique. Cinq comportements sont définis pour contrôler chaque robot : STANDBY, MOVE, SENSE, GRIP et DROP. Les relations entre ces comportements sont illustrées dans la figure 2.4. Le comportement STANDBY définit les instants pendant lesquels le robot est en attente d'une nouvelle tâche et est au dépôt. Le comportement MOVE signifie que le robot est en cours de déplacement. Le comportement SENSE signifie que le robot est en train de percevoir l'emplacement du dépôt pour se positionner ou l'emplacement de la source pour prendre la marchandise ou l'emplacement de la destination pour déposer la marchandise. Les comportements GRIP et DROP signifient que le robot est asservi selon un processus de préhension ou de libération, pour prendre ou déposer une marchandise.

2.3 Conclusion

Dans ce chapitre, nous avons présenté nos travaux sur la planification de tâches multi-robots dans des scénarios d'exploration et de transport. Dans la première

section, nous avons d'abord présenté une nouvelle approche de répartition des tâches pour un système multi-robots. Cette approche appelée approche contractuelle centrée sur les échanges, est décentralisée et est basée sur l'échange d'informations dédiée à la coordination. L'idée fondamentale de cette nouvelle approche consiste à simuler les relations entre acheteurs et vendeurs dans un système commercial et à allouer dynamiquement la tâche en utilisant un mécanisme d'*offre non sollicitée*. Cette approche repose sur l'expression des principes d'utilisabilité, d'efficacité et de robustesse des solutions envisageables dans le déploiement d'un service d'exploration multi-robots en présence d'un environnement contenant des obstacles statiques. Les deux principaux composants de cette approche sont un mécanisme de répartition des rôles et un mécanisme de répartition de tâches. Dans le contexte de la prise de décision pour la coordination dans les systèmes multi-robots décentralisés, les robots doivent établir leurs plans respectifs en fonction de l'information locale observable avec une communication limitée.

Dans la deuxième section, nous avons présenté une nouvelle approche pour le problème de l'accomplissement d'une mission de transport multi-robots. Notre solution propose une assignation des tâches de transport basée sur l'estimation des temps d'apparition des marchandises dans une approche heuristique.

Chapitre 3

Contributions à la planification de mouvements

Dans ce chapitre, nous présentons nos travaux sur le sujet de la planification de mouvements multi-robots dans le contexte des problèmes d'exploration et de transport. Après une présentation des méthodes existantes, nous proposons une nouvelle approche par échantillonnage dédié à la décision pour l'exploration et une nouvelle approche utilisant des feuilles de route probabilistes pour le transport.

La première approche de ce chapitre utilise un échantillonnage réalisé dans un espace des configurations proches de *points clés*. Ces *points clés* sont identifiés à l'aide d'un *diagramme de Voronoï* et considérés comme des points de passage pour les robots. L'échantillonnage permet de construire un ensemble de chemins différents pour chaque robot. Il est cependant possible de retrouver deux configurations proches, voir identiques, dans deux chemins. Le nombre d'interférence entre les chemins générés est dépendant de la taille de l'espace des configurations, du nombre de points de passage identifiées, de la commandabilité des robots et du nombre de robots nécessaires à chaque instant. Les interférences entre robots sont donc évitées en probabilité, sans l'être systématiquement. Contrairement aux approches classiques de la littérature [120, 130], notre approche permet de planifier des chemins cinématiques supposés différents pour chaque robot afin de minimiser les problèmes de collision, de congestion et d'interblocage.

La deuxième approche de ce chapitre utilise des feuilles de route probabilistes construites à l'aide d'un échantillonnage adaptatif de l'espace des configurations. Cette approche est appelée ACS-PRM et se compose de trois étapes : un échantillonnage de l'espace des configurations, la construction de feuilles de route et la planification de mouvements. La construction des feuilles de route présuppose un échantillonnage adéquat de l'espace des configurations. Cet échantillonnage est défini sur une grille d'occupation en utilisant une méthode probabiliste adap-

tative (la densité de l'échantillonnage évoluant en fonction de l'environnement). Ceci permet de définir des points d'intérêts de deux types : les cibles potentielles qui correspondent à des points de possibles interférences et les jalons qui sont des points de liaison entre les cibles potentielles. Les feuilles de route sont des graphes composé de deux types de noeuds, pouvant inclure des cycles. La planification est ensuite défini par questionnement de ces feuilles de route pré-établies afin de trouver un chemin reliant la configuration courante à la configuration finale recherchée.

Ce chapitre est organisé comme suit : la section 3.1 présente un ensemble de travaux connexes et ayant inspirés nos approches ; la section 3.2 analyse le problème d'apparition des situations d'attente ; la section 3.3 présente notre approche par échantillonnage dédié à la décision ; la section 3.4 présente notre approche par échantillonnage adaptatif ; le chapitre se conclut dans la section 3.5.

3.1 Méthodes existantes

Švestka et Overmars [117] ont présenté une approche pour la planification de mouvements de robots mobiles non-holonomes dans un même espace de travail. Leurs travaux proposent l'utilisation de feuilles de route probabilistes, dans lesquelles les feuilles de route du système multi-robots sont établies par composition de feuilles de route locale pour un unique robot. Les auteurs ont également introduit la notion de *super-graphes* pour la planification de chemin multi-robots. Cependant, une grande partie de l'exécution de cette méthode est utilisée dans la recherche de chemin dans le *super-graphe*. Cette recherche est effectuée pour chaque requête individuelle formulée par chaque robot dans la planification de ses mouvements. Cette approche est probabiliste complète : quand le temps d'exécution tend vers l'infini, la probabilité qu'a cette méthode de trouver une solution tend vers 1.

Moors *et al.* [84] ont présenté un algorithme basé sur le graphe pour la planification de mouvements multi-robots coordonnée dans des environnements intérieurs à deux dimensions. Le cas pratique utilisé dans leurs travaux est la surveillance en environnement intérieur à l'aide d'un ensemble de robots. L'approche proposée génère un plan de mouvement coordonné par l'utilisation de l'algorithme de recherche A* [54], tout en prenant en compte les limitations et les incertitudes des capteurs. Les auteurs ont également introduit un framework basé sur de modèles réalistes de capteurs probabilistes et de hypothèses du pire des cas sur les motions de l'intrus. Ce framework permet de comparer les différentes approches et d'évaluer la performance de coordination de leur approche. Cependant l'utilisation de l'algorithme A* nécessite une simplification de l'environnement par discrétisation (sous forme de pavage). Cette simplification, dont le pas de dis-

crétisation influe sur les résultats¹, conduit à un compromis entre capacité de résolution et optimal recherché.

Clark [28] a présenté une stratégie de planification de mouvements pour un système multi-robots basée sur l'utilisation de feuilles de route probabilistes. Le réseau établi dans les communications entre robots est considéré comme une plate-forme de coordination. Ce réseau appelé réseau dynamique de robots ou *Dynamic Robot Network* (DRN) en anglais, est un réseau *ad hoc*², dans lequel l'approche de feuilles de route probabilistes est utilisée comme planificateur central. Chaque réponse fournie par le planificateur central, tient compte des requêtes et des réponses précédentes. L'empilement des réponses fournies permet d'augmenter la couverture globale de l'espace des configurations. Une seconde étape de génération de jalons est utilisée pour diminuer le temps de planification de chaque robot. L'utilisation d'une région nommée "endgame" permet d'augmenter la convergence des solutions proposées vers la configuration finale de chaque robot. Les résultats de cette approche montre une réelle efficacité dans le cas d'environnements structurés composés d'obstacles statiques. Point faible et point fort de cette approche demeurent dans la nature probabiliste des feuilles de route, dont le point faible est la convergence probabiliste des calculs vers une solution et la relative indépendance de la complexité de la résolution en fonction de la dimension du problème (et donc par extension du nombre de robots).

Saha et Isto [104] ont présenté une stratégie pour la planification de mouvements multi-robots décentralisée. Cette stratégie propose de fusionner les deux étapes d'une méthode de planification existante [71] pour améliorer sa fiabilité. La première étape calcule un chemin sans collision par robot. Chaque chemin est établi indépendamment des autres chemins, en ne considérant qu'un robot dans l'environnement. La deuxième étape définit le suivi de chaque chemin pour chaque robot en plaçant des situations d'attente à l'entrée de chaque interférence. Les interférences créent des priorités entre les robots et des espaces de coordination. La solution étudiée utilise des priorités fixes et l'utilisation de politiques de priorités présentée en perspective. Les auteurs ont démontré qu'une approche de planification centralisée donne de meilleurs résultats qu'une approche décentralisée dans de nombreux cas. Ce résultat est lié à l'utilisation des composants de la méthode probabiliste, dont l'utilisation a pour conséquence la convergence probabiliste de cette approche.

1. Un pas de discrétisation trop grand pourra supprimer des solutions et un pas de discrétisation trop petit pourra conduire à une explosion combinatoire, rendant le calcul non achevable.

2. Les réseaux ad hoc sont des réseaux sans fil capables de s'organiser sans infrastructure définie préalablement.

3.2 Situations d'attente

Les situations d'attente, rencontrées au fil de l'exécution d'un plan, sont l'une des raisons les plus importantes limitant les résultats des méthodes de planification de mouvements multi-robots. Ces situations d'attente sont divisées en trois catégories :

- les collisions, quand deux robots possèdent deux chemins avec un même point de passage à un même instant donné ;
- les congestions, quand un espace local est trop petit pour permettre les mouvements des robots à un instant donné ;
- les interblocages entre robots, quand une seule route est partagée par deux robots circulant en sens inverse.

Ce problème d'apparition des situations d'attente est équivalent à l'ajout d'une contrainte d'exclusion mutuelle aux points de passage lors de la phase de planification de mouvements multi-robots. L'ajout de cette contrainte implique l'absence de solutions dans certains cas, créant ainsi des situations d'attente obligatoires. Ce problème est une conséquence de la génération des chemins et pose donc la question de savoir s'il faut chercher à composer dans ces chemins (*i.e.* en ajoutant des situations d'attente) ou s'il faut revoir les chemins identifiés.

L'exclusion mutuelle peut être divisée en quatre catégories, comme présentée dans la figure 3.1. Lorsque deux robots se déplacent vers le même point de passage simultanément :

- dans la figure 3.1(a), le robot rouge et le robot bleu se déplacent de la droite vers la gauche séquentiellement (*i.e.* évoluant dans la même direction, selon le même chemin) ;
- dans la figure 3.1(b), le robot rouge et le robot bleu se déplacent de la droite vers la gauche simultanément (*i.e.* évoluant dans la même direction, selon des chemins différents) ;
- dans la figure 3.1(c), le robot rouge se déplace de la gauche vers la droite et le robot bleu se déplace de la droite vers la gauche (*i.e.* évoluant dans des directions différentes, selon le même chemin) ;
- dans la figure 3.1(d), le robot rouge se déplace du haut vers le bas et le robot bleu se déplace de la droite vers la gauche (*i.e.* évoluant dans des directions différentes, selon des chemins différents).

Deux tendances sont admises pour la gestion des exclusions mutuelles aux points de passage :

- La première tendance consiste à laisser les robots passer le point de passage un par un ou *One By One* (OBO) en anglais [117] (cf. figure A.1). La faiblesse de cette stratégie est qu'un robot doit attendre un autre robot pour pouvoir passer. Cette solution prolonge les temps d'achèvement des missions.

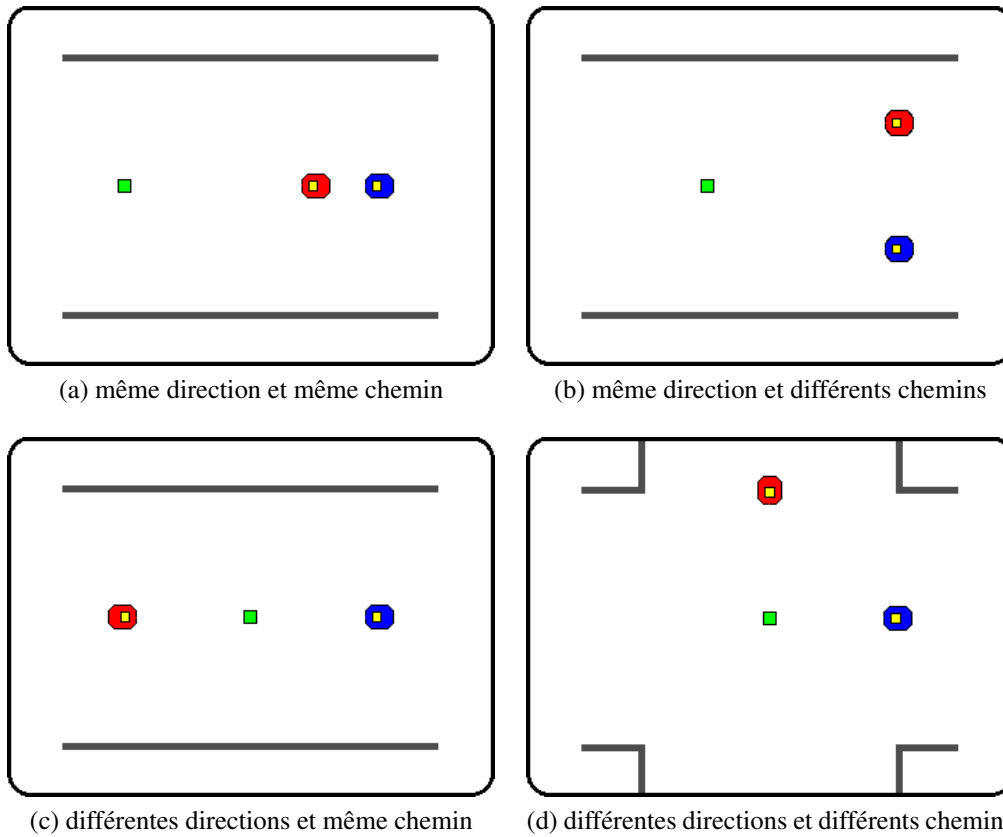


FIGURE 3.1 – Exemples type d'exclusion mutuelle de point de passage. Le cube vert de petite taille représente le point de passage.

- La seconde tendance consiste à re-planifier le chemin local en temps réel pour chaque robot par planification locale et évitement d'obstacles, à l'aide d'algorithmes comme l'histogramme de champ de vecteur plus ou *Vector Field Histogram Plus* (VFH+) en anglais [122] (cf. figure A.2) et le diagramme de proximité ou *Nearness Diagram* (ND) en anglais [82] (cf. figure A.3). La faiblesse de cette stratégie réside dans le temps nécessaire à la re-planification de nouveaux chemins, pouvant impliquer des temps d'attente supplémentaires non négligeables.

Ces deux tendances prolongent le temps d'exécution des missions. Nous proposons dans ce mémoire de réduire les cas d'apparition des situations d'attente par la mise en place d'une planification minimisant les possibles interférences entre les différents chemins des robots. L'augmentation du nombre de points de passage dans l'espace permet la génération de chemins possibles réduisant les possibles apparitions de situation d'attente. Nous proposons une solution permettant

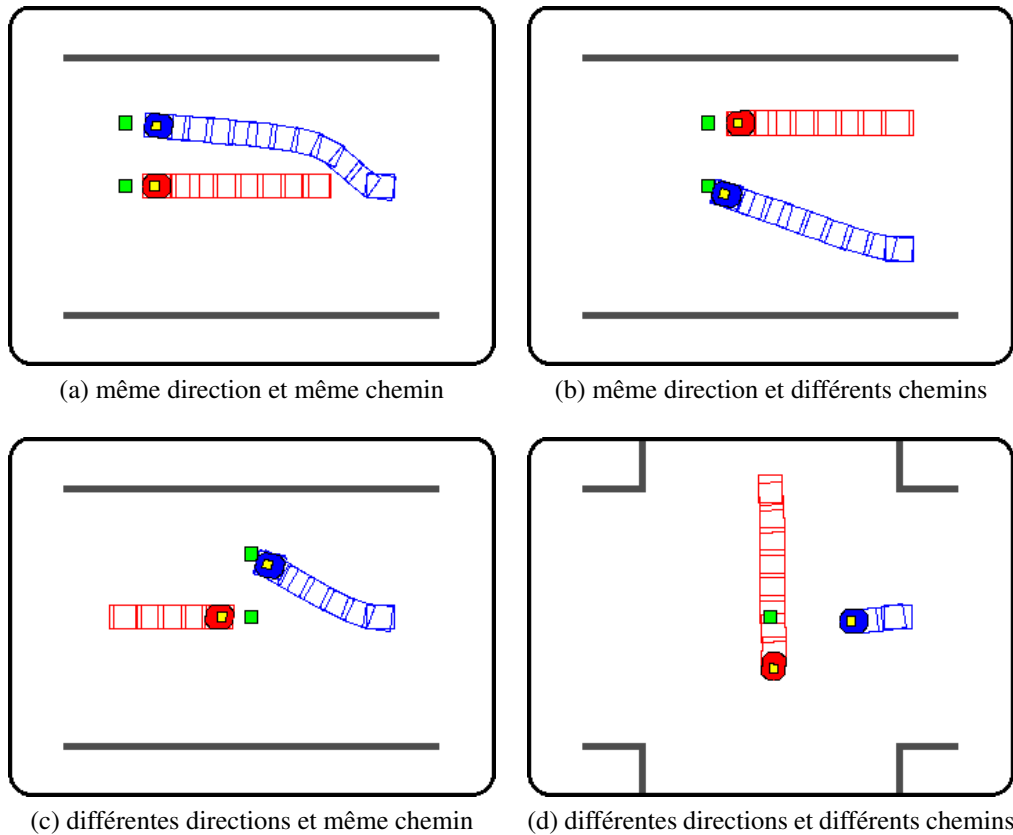


FIGURE 3.2 – Trajectoires de robots évitant les situations d’attente

de réduire les cas de collisions, de congestions et d’interblocages, minimisant les appels aux planificateurs locaux des robots et réduisant ainsi le temps d’exécution des missions. La figure 3.2 illustre une solution de ce type pour quatre situations choisies. Cette figure illustre les cas recherchés de points de passage, dans lesquels les situations d’attente sont évitées.

3.3 Approche par échantillonnage dédié à la décision

Cette section³ présente une approche par échantillonnage de l’espace des configurations conçue pour atténuer les problèmes d’apparition des situations d’attente observées dans l’exécution des missions multi-robots. L’objectif est d’ex-

3. Cette partie de nos travaux a été publiée dans *Proceedings of the Joint 41st International Symposium on Robotics and 6th German Conference on Robotics (ISR/ROBOTIK 2010)*.

plorer l'environnement dans son ensemble dans un temps d'exécution minimum à l'aide d'un group de robots.

L'approche proposée commence par échantillonner l'espace des configurations sur la base du *diagramme de Voronoï* en utilisant l'approche de feuille de route probabiliste (PRM) [65]. Elle se poursuit en assignant individuellement les robots à des points cibles d'exploration en utilisant la *méthode hongroise* [69]. L'idée développée est de construire une carte topologique différente pour chaque robot, permettant d'obtenir des chemins d'exploration différents. Ceci réduit le nombre des apparitions de situation d'attente et diminue le temps total d'exploration.

3.3.1 Diagramme de Voronoï

Nous utilisons d'abord le *diagramme de Voronoï* pour construire une carte topologique originale. Le *diagramme de Voronoï* est une méthode de partitionnement de graphe utilisé dans de nombreuses solutions pour la segmentation de carte [70, 120, 27, 140]. Pour une carte m fixée, pour chaque point p dans l'espace libre S de m , un ensemble $O_p(m)$ est défini. Cet ensemble est constitué des points d'obstacle plus proches à p . Le *diagramme de Voronoï* est ainsi représenté par un graphe $G(m) = (V, E)$, où V est l'ensemble des points dans $O_p(m)$ qui sont équidistants d'au moins deux obstacles les plus proches. E est l'ensemble des arêtes reliant deux points adjacents dans V de m :

$$V = \{p \in S \mid |O_p(m)| \geq 2\} \quad (3.1)$$

$$E = \{(p, q) \mid p, q \in V, p \text{ adjacent to } q\} \quad (3.2)$$

La figure 3.3 montre les *diagrammes de Voronoï* de deux cartes.

Une carte topologique est construite à partir de la carte grille d'occupation⁴ en appliquant le *diagramme de Voronoï*⁵. La carte topologique ainsi établie permet de proposer une solution pour la planification d'un unique robot dans un environnement statique [120]. L'idée présentée dans ces travaux est de décomposer la carte grille en un petit ensemble de régions séparées par des passages étroits tels que les portes. Ces passages étroits sont appelés les *lignes clés* et sont calculés par analyse d'un squelette de l'environnement. La carte partitionnée permet de construire un graphe isomorphe de l'espace, où les nœuds correspondent à des régions et les arcs permettent de relier les régions voisines. L'existence d'un arc implique l'existence d'un chemin pour un unique robot. Ce graphe est appelé carte

4. Une carte grille d'occupation permet de traduire les obstacles statiques de l'environnement.

5. Un *diagramme de Voronoï* permet de traduire les espaces libres de manoeuvre autour des ces obstacles

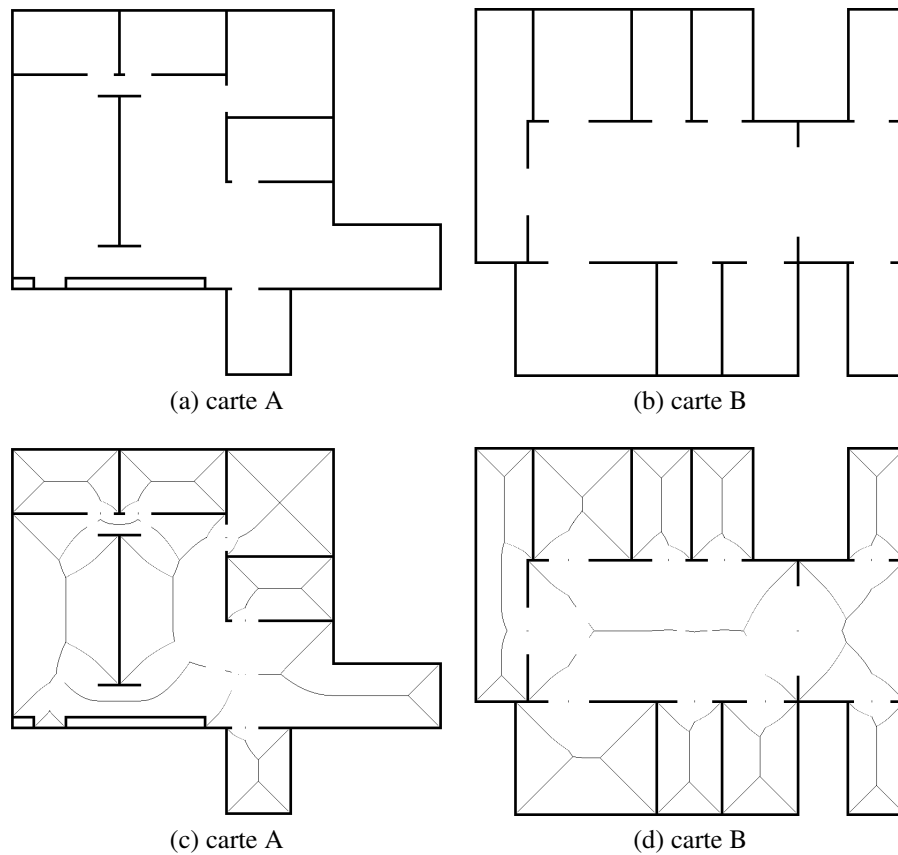


FIGURE 3.3 – Exemple de *diagrammes de Voronoï* de deux cartes. (a) et (b) représentent les cartes originales. (c) et (d) représentent les *diagrammes de Voronoï* générés.

topologique. Cette stratégie génère aussi une série de positions appelés *points clés*, où la distance est un minimum local à l'obstacle le plus proche dans la carte. Les *points clés* sont également les milieux des *lignes clés*. Ces points peuvent être considérés comme des points de passage pour la planification de mouvements de robots. Ils identifient également les portes et les passages difficiles. La stratégie présentée ici comporte des points négatifs :

- une identification approximative des portes ;
- une inadéquation aux environnements encombrés.

Pour remédier à ces problèmes, nous proposons un nouveau processus d'identification des points de passage. Ce processus est divisé en deux étapes :

- La première étape consiste à déterminer les portes ou passages étroits entre deux espaces plus larges. Les portes sont les milieux des segments dont la longueur à proximité de l'épaisseur des obstacles.

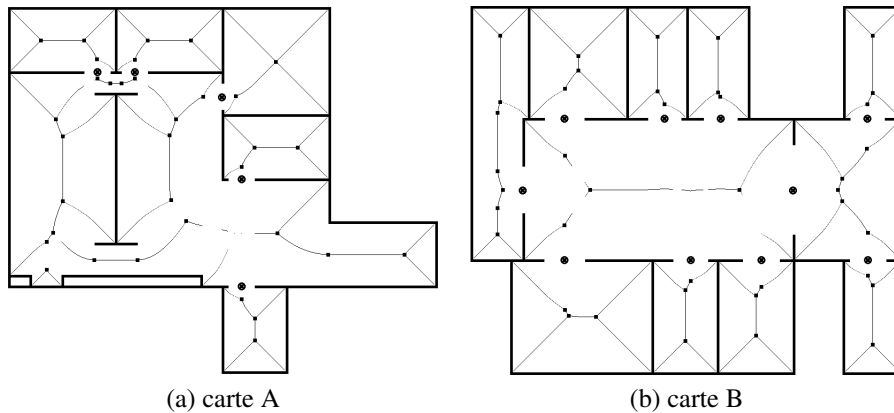


FIGURE 3.4 – Points de passage résultants du processus d'identification. Le point noir représente le point de passage. Le point noir entouré d'un cercle représente la porte.

- La deuxième étape consiste à déterminer les points de passage autres que les portes, qui sont les sommets du *diagramme de Voronoï*.

La figure 3.4 présente une application de ce processus aux deux cartes précédentes. Il apparaît un nombre plus importants de points de passage dans l'ensemble des cartes. Ce processus nécessite cependant une estimation de l'épaisseur minimale des obstacles.

3.3.2 Feuille de route probabiliste

L'échantillonnage de l'espace des configurations sur la base du *diagramme de Voronoï* est suivi par un deuxième échantillonnage basé sur l'utilisation des feuilles de route probabilistes (définies par la méthode PRM). La méthode PRM originale⁶ est présentée par Kavraki *et al.* [65]. Cette méthode est définie selon trois étapes :

- Premièrement, un échantillonnage aléatoire est construit dans l'espace des configurations libres (notée C_{libre} ⁷ du robot. Une configuration est dite libre si elle ne place pas le robot en collision avec un obstacle (l'espace des configuration associé est noté C_{obs}). La taille de cet échantillon de configurations est fixé arbitrairement en fonction d'une estimation de la complexité de l'environnement.

6. Une implémentation de référence de cet algorithme en C++ est disponible en ligne à l'adresse : <http://www.ai.univ-paris8.fr/~yz>

7. Dont l'écriture est C_{free} en anglais.

- Deuxièmement, un planificateur local⁸ relie les configurations de l'échantillon précédemment généré. Seules les configurations proches sont testées. Le résultat est un graphe appelé une feuille de route, définissant un ensemble de chemins.
- Troisièmement, chaque requête de planification globale est résolue par composition des chemins de la feuille de route. La résolution d'une requête commence par une tentative de connexion de la configuration initiale à une configuration de la feuille de route et une tentative de connexion de la configuration finale à une configuration de la feuille de route. Si ces deux appels au planificateur local sont réalisés avec succès, le problème de planification devient un problème de recherche de chemins dans un graphe (l'algorithme A* permet de résoudre ce problème dans ce cas).

La méthode PRM calcule une feuille de route sans collision dans un espace de configurations dans laquelle le robot est réduit à un point. Ce point représente le modèle du robot dans l'environnement et la dimension de l'espace des configurations est égal au nombre de degrés de liberté du robot. L'algorithme cherche ainsi un chemin libre pour un point dans la feuille de route. La figure 3.5 illustre ce processus de génération d'une feuille de route probabiliste pour un exemple de carte avec 80 échantillons aléatoires. La grille d'occupation ici utilisée possède une résolution de 500×500 .

Une extension de la méthode PRM, appelée feuille de route probabiliste basée sur l'axe médian ou *Medial Axis based PRM* (MAPRM) en anglais [128], propose de déplacer toutes les configurations constatées (*i.e.* libres et non libres) sur l'axe médian de l'espace libre et deviennent donc toutes des configurations libres. Nous proposons une variation de cette méthode qui ne transforme pas les configurations de C_{obs} (*i.e.* l'égalité entre itération de l'échantillonnage et nombre de configuration est perdue). Pour appliquer ce déplacement sans un calcul explicite de l'axe médian, nous proposons d'utiliser l'algorithme 3.1.

Seules les configurations aléatoire p échantillonnée jugées dans C_{libre} sont évaluées (algorithme 3.1 ligne 3). Par proximité aux obstacles, une configuration q est recherchée. Les configurations de ce type sont recherchées par échantillonnage d'un espace proche de p . Si aucune configuration n'est trouvée dans une proximité fixée, l'algorithme se poursuit en augmentant la valeur de proximité. Les premières configurations trouvées sont considérées comme représentatives des obstacles les plus proches. Quand q et p sont définies, elles permettent de définir une troisième configuration r par déplacement sur l'axe défini par le couple (p, q) . Ce déplacement de r répété tant que r est dans la carte et tant qu'il appartient à C_{libre} , nous permet de trouver une configuration de C_{obs} , proche de p . Ce rapprochement est réalisé par dichotomie dans C_{libre} (algorithme 3.1 lignes 6

8. Souvent, ce planificateur est par soucis de simplicité un planificateur en ligne droite.

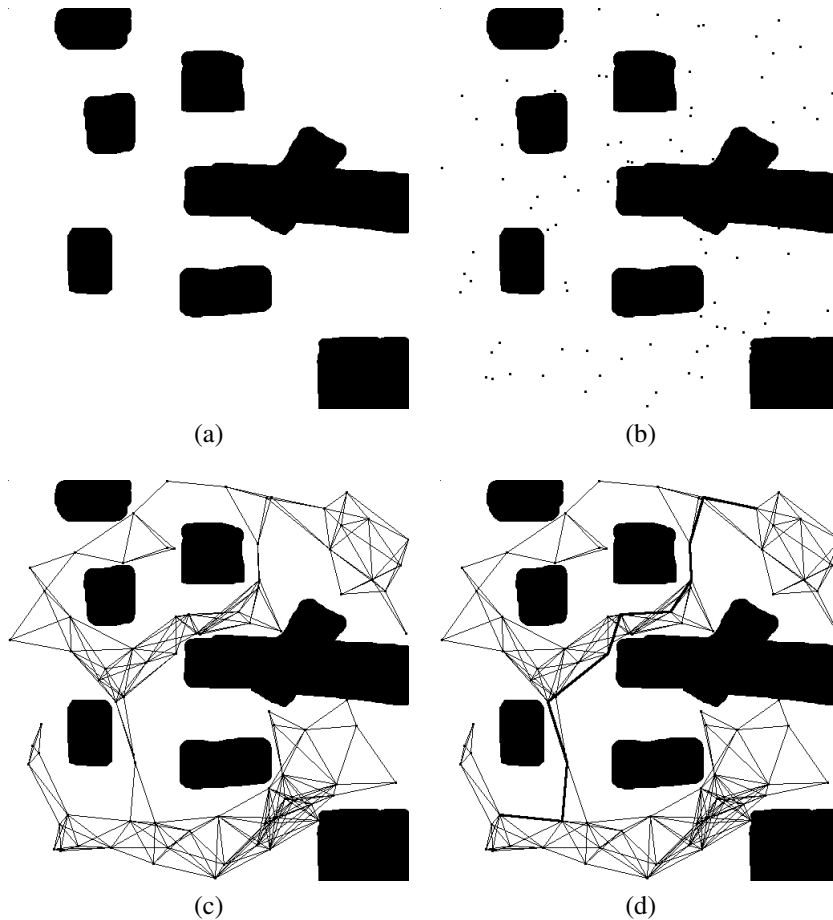


FIGURE 3.5 – Processus de génération d’une feuille de route probabiliste. (a) présente l’espace de travail. (b) présente des positions aléatoires de C_{libre}^9 . (c) présente la feuille de route construite en reliant les échantillons les plus proches avec un planificateur en ligne droite. (d) présente le chemin résultant de la recherche d’un chemin entre deux configurations.

à 9). L’algorithme produit une recherche auto-adaptative de l’axe médian, basée sur l’évaluation des translations possibles dans l’espace libre. Notre solution est basée sur la distinction de trois types de sous-espace dans C :

- les portes, qui sont identifiées dans l’étape précédente, par utilisation du *diagramme de Voronoï* ;
- les chambres, qui sont identifiées par les groupes de noeuds isolés du reste du graphe par une ou plusieurs portes ;
- les couloirs, qui sont les noeuds restants, connectés au positions clés de C .

Algorithme 3.1 Échantillonnage sur l'axe médian

```

1:  $nb \leftarrow 0$ 
2: while  $nb < N$  do
3:    $p \leftarrow \text{rand}()$ 
4:   if  $p \in C_{\text{libre}}$  then
5:      $q \leftarrow \text{plus\_proche}(p, C_{\text{obs}})$ 
6:      $r \leftarrow 2p - q$ 
7:     while  $r \in C_{\text{libre}}$  do
8:        $p \leftarrow \frac{p+r}{2}$ 
9:        $r \leftarrow 2p - q$ 
10:    end while
11:    if  $r \in C_{\text{obs}}$  then
12:       $AM \leftarrow AM \cup p$ 
13:    end if
14:  end if
15:   $nb++$ 
16: end while

```

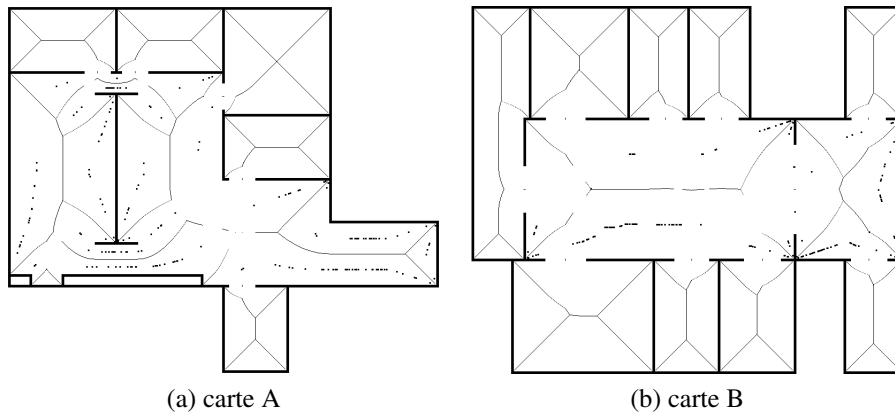


FIGURE 3.6 – Positions générées par échantillonnage sur la base du *diagramme de Voronoï*

Les résultats de l'application de l'algorithme 3.1 sont présentés dans la figure 3.6, dont l'échantillonnage est restreint aux espaces identifiés comme couloirs.

3.3.3 Méthode hongroise

Notre solution divise les environnements intérieurs en zones avec des fonctionnalités différentes appelées portes, chambres et couloirs. Pour maximiser la

contribution de chaque robot à la solution globale, notre solution favorise l'exploration de chambres différentes pour chaque robot. Par conséquent, les portes sont affectées à des robots comme des tâches. Cette affectation est réalisée à l'aide de la *méthode hongroise* [69]. Cette méthode est un algorithme d'optimisation combinatoire, permettant de résoudre un problème d'affectation en temps polynomial (*i.e.* $O(n^3)$). Une mise en œuvre utilisant la *méthode hongroise* pour calculer les affectations de cibles d'exploration à des robots a été présenté par Ko *et al.* [67]. Dans le contexte d'un problème multi-robots, il s'agit de coordonner n robots et n tâches. La solution recherchée dans l'attribution des tâches vise à minimiser le coût total (en temps ou en énergie).

Notre solution utilisant la *méthode hongroise* est composée de quatre étapes ¹⁰ :

1. Construire une matrice de taille $n \times n$ contenant les coûts d'affectation de chaque tâche pour chaque robot ¹¹. Ces coûts correspondent à des valeurs de distance ou des valeurs de consommation d'énergie. Le calcul se poursuit dans l'étape suivante.
2. Réduire la matrice en ligne et en colonne. Cette étape passe par l'identification du plus petit élément de chaque ligne et par une réduction d'autant chaque valeur de la ligne. Elle se poursuit par l'identification du plus petit élément de chaque colonne et par une réduction d'autant chaque valeur de la colonne. Il est important de noter que cette étape fait apparaître des valeurs nulles (*i.e.* zéros). Le calcul se poursuit dans l'étape suivante.
3. Calculer le nombre minimal de lignes nécessaires pour couvrir tous les valeurs nulles dans la matrice en ligne et en colonne. Si le nombre des lignes est égale à n , alors la recherche d'une répartition prends fin. Le plan optimal de répartition des tâches est donné par sélection d'une combinaison de la matrice des coûts modifiée, de telle façon que la somme est égale à zéro. Sinon le calcul se poursuit dans l'étape suivante.
4. Trouver le plus petit élément qui n'est pas couvert par l'une des lignes. Puis le soustraire de chaque élément non couvert et l'ajouter à chaque élément couvert. Le calcul se poursuit dans l'étape 3.

L'utilisation des étapes précédentes nécessite une matrice carrée. Ceci présuppose un nombre de tâches égale au nombre de robots (sous la condition que chaque robot peut effectuer une seule tâche à chaque instant). Si cette matrice n'est pas carré, elle sera simplement complétée par des robots fictifs ou des tâches fictives à l'aide de valeurs nulles. La matrice ci-dessous est un exemple de résolution à trois robots (notés r_0 , r_1 et r_2) pour deux tâches (notées t_0 et t_1) :

10. Une implémentation de référence de cet algorithme en C++ est disponible en ligne à l'adresse suivante : <http://www.ai.univ-paris8.fr/~yz>

11. Il existe une nouvelle matrice de coût à chaque nouvelle requête de planification.

Algorithme 3.2 Exploration par échantillonnage dédié à la décision

-
- 1: Construction du *diagramme de Voronoï*.
 - 2: Extraction des points de passage et identification des portes.
 - 3: Construction d'une feuille de route probabiliste.
 - 4: **while** \exists sous-espace non exploré **do**
 - 5: Attribution de tâches avec la *méthode hongroise*
 - 6: **end while**
-

$$C = \begin{pmatrix} \text{cout}(r_0, t_1) & \text{cout}(r_0, t_2) & 0 \\ \text{cout}(r_1, t_1) & \text{cout}(r_1, t_2) & 0 \\ \text{cout}(r_2, t_1) & \text{cout}(r_2, t_2) & 0 \end{pmatrix} \quad (3.3)$$

Ces trois composants (*i.e.* *diagramme de Voronoï*, feuille de route probabiliste et *méthode hongroise*) utilisés dans notre solution sont combinés à l'aide de l'algorithme 3.2. Leur utilisation combinée permet de prendre en compte les *points clés* et les points d'échantillonnage pour coordonner les robots dans l'achèvement de leurs tâches respectives, en fonction d'une métrique choisie¹².

3.3.4 Limite de l'approche

La figure 3.7 présente deux cas limite d'environnement. La situation A est un exemple d'intersection et la situation B est un exemple de passage étroit. Sur ces deux situations, le *diagramme de Voronoï* est superposé avec un échantillonnage composé de 20 configurations. Il apparaît que les configurations résultantes (issues du *diagramme de Voronoï* et issues de notre méthode d'échantillonnage dédié à la décision) sont très similaires. Dans de tels cas, l'échantillonnage résultant sera contraint dans des régions topologiques identiques et les trajectoires des robots seront similaires. Les résultats de l'utilisation d'un *diagramme de Voronoï* ou d'une politique échantillonnage (par exemple dédiée à la décision) seront de fait comparables. Les situations d'attente ne pourront pas être évitées par cloisement topologique. Dans le cadre des résolutions exemptes de dynamique temporelle (telles que notre approche par échantillonnage dédié à la décision), les intersections et les passages étroits sont un facteur limitant important des temps d'exécution d'exploration et de transport des architectures multi-robots.

Dans le cas des intersections, la coordination délibérative est connue pour permettre l'action simultanée de plusieurs robots dans un espace réduit selon des règles prédéfinies [64]. Notre approche par échantillonnage devient de fait, par manque d'espace libre, non adaptée à la résolution directe de telles situations.

12. Dans notre cas, ce coût est fonction de la distance parcourue.

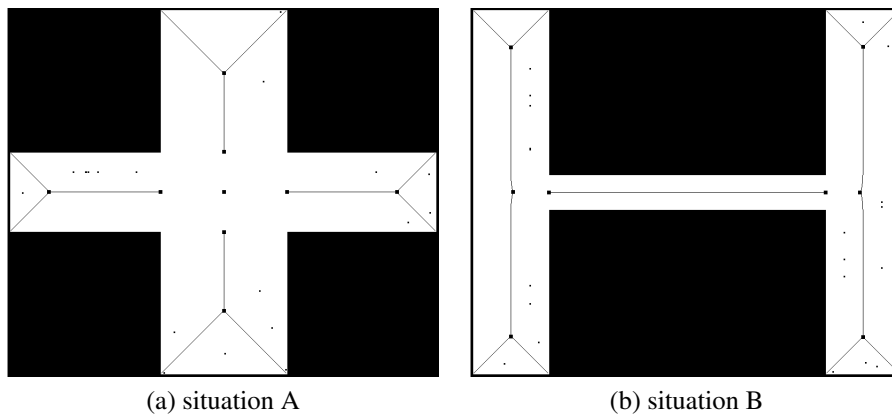


FIGURE 3.7 – Cas limite par échantillonnage dédié à la décision

Dans le cas de zones d'intersection topologiquement accessibles à plusieurs robots simultanément, il apparaît des possibilités d'attente en mouvement (par exemple, dans des chemins en boucles).

Les passages étroits font apparaître de nouveaux problèmes, notamment celui d'un passage engagé simultanément par deux robots en sens inverse. Arrivés au point de rencontre au milieu d'un passage étroit (limitant le passage à un unique robot), la seule évolution possible est de forcer l'un des deux robots à rebrousser chemin. Pour éviter de telles situations, une condition de bonne exécution, sans retour solution alternative simple à l'utilisation de plusieurs robots dans un passage étroits propose une condition d'accès aux passages étroits [117] (limitée au nombre de robots capables de passer de front dans un tel passage).

3.4 Approche par échantillonnage adaptatif

Cette section¹³ présente une nouvelle approche basée sur l'échantillonnage. A l'instar de la section précédente, notre solution est établie par identification des situations d'attente liés aux problèmes de collisions, de congestions et d'interblocages. Le problème de transport de marchandises multi-robots est choisi comme scénario de référence.

Pour faire face au problème d'apparition des situations d'attente dans les solutions de transport de marchandises multi-robots, nous proposons une nouvelle approche utilisant un planificateur de feuilles de route probabilistes (PRM) lui-même basé sur un échantillonnage croisé adaptatif ou *Adaptive Cross Sampling*

13. Cette partie de nos travaux a été publiée dans *Proceedings of the 12th International Conference on Intelligent Autonomous Systems (IAS 2012)*.

(ACS) en anglais. Cette approche appelée ACS-PRM est décomposée en trois étapes principales :

- Une première étape de génération de configurations en correspondance avec une grille d’occupation. Cette phase de génération est définie par une méthode d’échantillonnage croisé adaptatif.
- Une deuxième étape de construction de feuille de route contenant des cibles potentielles et des jalons, résultants de l’étape précédente.
- Une troisième étape de planification et d’exécution de mouvements basée sur la feuille de route de l’étape précédente.

Il existe deux principales approches qui permettent de résoudre le problème de la planification de mouvements pour un unique robot basé sur la représentation de grille de l’environnement (cf. section 1.7.2). La première consiste à utiliser les algorithmes de recherche incrémentale tels que A^* ¹⁴ [54], D^* [112] et *wavefront* [71]. L’utilisation de ces planificateurs nécessite une discrétisation ou un pavage de l’environnement. Ceci implique un compromis entre pas de discrétisation et temps de réponse attendu. Un grand pas de discrétisation permet une résolution très fine et demande un temps important de calcul. Un petit pas de discrétisation permet une résolution approximative (au pas de résolution près choisi) et demande peu de temps de calcul. Un grand pas de discrétisation demande également beaucoup de mémoire et un petit pas de discrétisation demande peu de mémoire.

La deuxième consiste à utiliser les méthodes basées sur l’échantillonnage telles que PRM [65] et RRT [72]. Ces méthodes ont été proposées pour permettre la résolution de problème de plus grande dimension en planification de mouvements. L’idée principale de ces approches est d’éviter la construction explicite de la région obstacle dans l’espace des configurations. Contrairement aux approches du paragraphe précédent (*i.e.* algorithmes de recherche incrémentale), les méthodes basées sur l’échantillonnage ont de bons résultats dans les environnements complexes et dans les espaces de configuration de grande dimension. Différentes extensions des méthodes PRM et RRT ont été proposées [5, 128, 111, 62, 51]. Notre approche présentée dans cette section est une nouvelle extension pour la planification de mouvements multi-robots.

3.4.1 Échantillonnage de l’espace des configurations

La première étape de notre approche ACS-PRM est un échantillonnage de C , dans lequel un nombre suffisant de points doit être généré pour être représentatif de l’espace libre de l’environnement. L’idée principale de cette étape est de laisser un point aléatoire p se rétracter en une position $P(q)$ à une distance d d’un obstacle

14. Une implémentation de référence de cet algorithme en C++ est disponible en ligne à l’adresse : <http://www.ai.univ-paris8.fr/~yz>

de C_{obs} selon des directions axiales orthogonales (cas dans lequel une solution simple est d'utiliser horizontale et verticale).

Pour des robots mobiles non holonomes placés dans un espace à deux dimensions (*i.e.* plan vu de dessus), trois degrés de liberté ou *Degree Of Freedom* (DOF) en anglais sont représentatifs de la configuration de chaque robot. Parmi ces trois DOFs, un DOF correspond aux commandes de rotation et deux DOFs correspondent à la translation le long des axes x et y . Les robots sont commandables selon une accélération vers l'avant et selon une accélération de l'angle de braquage. L'espace des configurations C est ainsi un groupe spécial euclidien :

$$SE(2) = \mathbb{R}^2 \times SO(2) \quad (3.4)$$

où $SO(2)$ est le groupe spécial orthogonal correspondant aux rotations.

Pour éviter les collisions, nous définissons une distance d égale à la somme d'un nombre positif w ¹⁵ et un rayon r avec son centre sur l'axe de rotation du robot¹⁶ :

$$d = r + w, (w > 0) \quad (3.5)$$

L'utilisation de w permet de faire face à l'influence négative d'erreurs provenant des capteurs.

C_{obs} représentant l'ensemble des configurations en collision avec les obstacles, $\forall q \in C_{obs}$ il s'agit de définir une direction r_q , puis de déterminer un point de symétrie $S(q)$ tel que :

$$S(q) = \{q + t\vec{r}_q \mid t > 0\} \cap C_{obs} \quad (3.6)$$

où, si :

$$\{q + t\vec{r}_q \mid t > 0\} \cap C_{obs} = \{q\} \quad (3.7)$$

alors :

$$S(q) = \infty \quad (3.8)$$

Sachant $dist(x, y)$ la distance entre le point x et le point y , la fonction de rétraction est définie par :

$$P(q) = \begin{cases} q + d\vec{r}_q & \text{si } dist(q, S(q)) \geq 2d \\ \frac{q + S(q)}{2} & \text{sinon} \end{cases} \quad (3.9)$$

15. La valeur de w est définie par l'utilisateur en fonction de l'environnement.

16. La valeur de r est définie par le cercle englobant permettant de couvrir l'ensemble du robot mobile.

Algorithme 3.3 Échantillonnage croisé adaptatif

```

1:  $nb \leftarrow 0$ 
2: while  $nb < N$  do
3:    $p \leftarrow \text{rand}()$ 
4:   if  $p \in C_{\text{libre}}$  then
5:     for chaque axe défini comme horizontal et vertical de  $C$  do
6:        $q \leftarrow \text{plus\_proche}(p, C_{\text{obs}})$ 
7:       if  $\text{dist}(p, q) \geq d$  then
8:          $p \leftarrow q + d\vec{r}_q$ 
9:       else
10:         $\{S(q)\} = \vec{qp} \cap C_{\text{obs}}$ 
11:         $p \leftarrow \frac{q+S(q)}{2}$ 
12:       end if
13:     end for
14:      $ACS \leftarrow ACS \cup p$ 
15:   end if
16:    $nb++$ 
17: end while

```

où $P(q)$ est la position rétractée du point p . Ainsi, l'échantillonnage aléatoire est adapté (cf. figure 3.8(b)) selon l'équation suivante :

$$ACS\text{-PRM} = \{P(q) | q \in C_{\text{obs}}\} \quad (3.10)$$

L'algorithme 3.3 présente le principe d'échantillonnage utilisé dans la solution ACS-PRM. Cet algorithme présente une complexité en temps de $O(n)$ et une complexité en espace de $O(1)$.

3.4.2 Construction de feuille de route

La deuxième étape est la construction d'une feuille de route, dans laquelle les cibles potentielles et les jalons sont extraites et connectées à la feuille de route. A partir de la rétraction des contours obtenue dans la section précédente, la construction de feuille de route est définie par :

- une première phase d'extraction de cibles potentielles se déroule comme suit : si $\text{dist}(p, q) < d$ alors p se rétracte selon $\frac{q+S(q)}{2}$ et est étiqueté comme appartenant à l'axe médian de C ; un ensemble de segments ayant une longueur l ¹⁷ est ainsi défini ; le milieu d'un segment est marqué comme une cible potentielle (cf. figure 3.8(c)) ;

17. Dans notre cas, la valeur de l est définie par l'épaisseur d'obstacle.

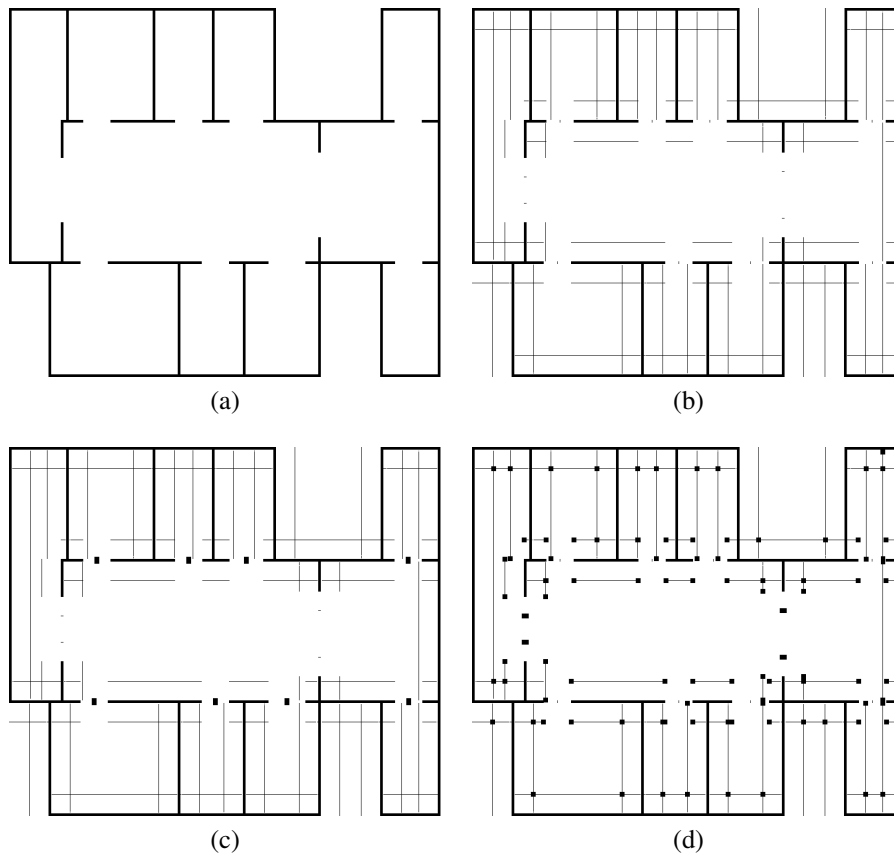


FIGURE 3.8 – Processus de génération d’une feuille de route à l’aide d’ACS-PRM. (a) présente la carte grille originale. (b) présente le résultat d’échantillonnage croisé adaptatif de C . (c) présente les cibles potentielles extraites (*i.e.* les portes). (d) présente les jalons extraits.

- une deuxième phase d’extraction de jalons à partir des terminaisons des segments ne contenant pas de cibles potentielles (cf. figure 3.8(d)) ;
- une troisième phase d’extraction de jalons à partir des intersections des contours (cf. figure 3.8(d)).

La figure 3.8 illustre le processus de génération d’une feuille de route à l’aide d’ACS-PRM pour un exemple de carte grille d’occupation contenant 200 000 échantillons aléatoires.

3.4.3 Planification de mouvements

La troisième étape est la planification de mouvements permettant la génération de chemins pour chaque robot. Ces chemins sont établis par interrogation de la

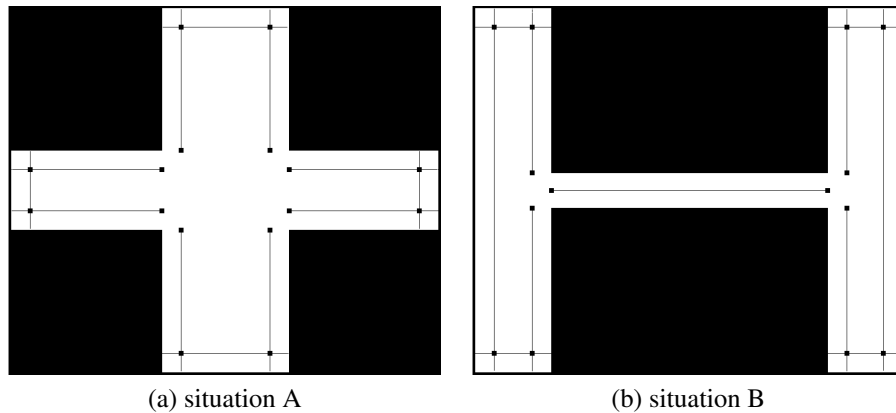


FIGURE 3.9 – Cas limite par échantillonnage adaptatif

feuille de route construite dans la section précédente. L'idée principale de cette étape se divise selon les trois principes suivants :

- Les cibles potentielles sont considérées comme nœuds objectifs de la planification et comme objets de la répartition de tâches. Ainsi les robots sont affectés aux cibles potentielles.
- L'attribution d'une cible potentielle est définie par le plus proche du robot présent mais plus loin de la cible attribué au robot précédent. Cette stratégie permet de maximiser la différences parmi les chemins.
- La planification en ligne droite du chemin de chaque robot dans l'espace des configurations selon la feuille de route. Cette étape correspond à l'utilisation classique de la méthode PRM.

3.4.4 Limite de l'approche

De façon similaire aux résultats précédemment présenté en section 3.3.4, les résultats obtenus montrent l'absence de gain pour les espaces réduits limités à un unique robot (intersection ou passage étroit).

La figure 3.10 présente les résultats obtenus dans un environnement irrégulier (non structuré) avec 20 000 échantillons aléatoires. Dans les environnements irréguliers, les zones d'intersections et les passages étroits sont d'autant plus difficiles à identifier de par la complexité des obstacles, des modèles des robots et donc des espaces de configurations résultants.

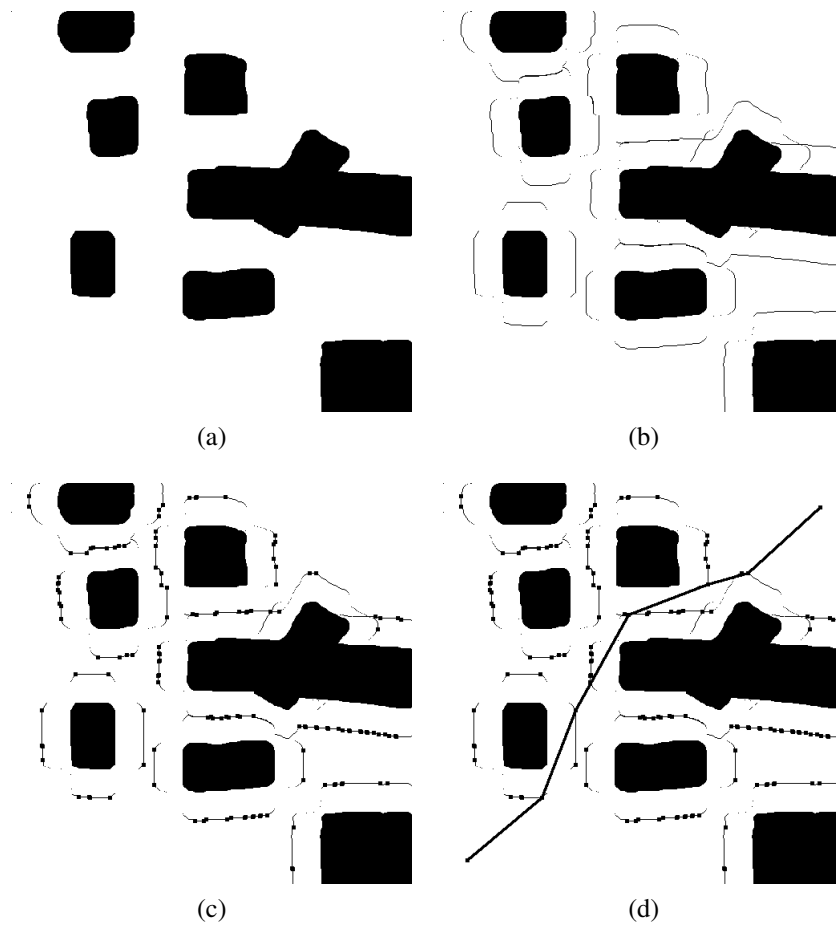


FIGURE 3.10 – Résultats en environnement irrégulier avec l’approche ACS-PRM. (a) présente un environnement irrégulier. (b) présente les configurations générées par l’approche ACS-PRM. (c) présente les jalons extraits. (d) présente un exemple de chemin par interrogation de la feuille de route construite.

3.5 Conclusion

Dans ce chapitre, nous avons présenté nos travaux concernant la planification de mouvements multi-robots. Les réflexions de nos travaux sont formalisés pour des cas d’exploration et de transport multi-robots.

Dans la première partie, nous avons présenté une nouvelle approche basée sur l’échantillonnage itératif visant à atténuer les problèmes de collision, de congestion et d’interblocage entre les robots. L’approche proposée combine *diagramme de Voronoï*, feuille de route probabiliste et *méthode hongroise*.

Dans la deuxième partie, nous avons présenté une nouvelle approche basée

sur l'utilisation de feuilles de route probabilistes associées à un échantillonnage croisé adaptatif (*i.e.* ACS-PRM). L'idée de l'approche réside dans la construction de chemins différents minimisant l'apparition des situations d'attente lors de l'exécution.

Chapitre 4

Expérimentations

Ce chapitre présente les expérimentations menées pour valider nos contributions présentées dans le chapitre 2 et le chapitre 3. Toutes les expériences rapportées dans ce document ont été réalisées avec un processeur Intel Core 2 Duo E8400 3.00GHz avec 2Go de RAM DDR2 cadencé à 800MHz. Les expérimentations sont réalisées avec le simulateur *Player/Stage* [46].

Ce chapitre est organisé comme suit : la section 4.1 présente les résultats expérimentaux obtenus avec l'approche contrarienne centrée sur les échanges ; la section 4.2 présente les résultats avec l'approche heuristique basée sur un modèle empirique ; la section 4.3 présente les résultats avec l'approche par échantillonnage dédié à la décision ; la section 4.4 présente les résultats avec l'approche par échantillonnage adaptatif ; le chapitre se conclut dans la section 4.5.

4.1 Approche contrarienne et exploration multi-robots

L'expérience choisie pour évaluer l'approche contrarienne centrée sur les échanges est l'exploration d'un environnement inconnu intérieur en utilisant un groupe de robots *Pioneer 2-DX*. Chaque robot est équipé d'un télémètre laser fournissant 361 échantillons dans un champ de vue de 180 degrés avec une portée maximale de 8 mètres. Chaque robot peut se localiser à l'aide d'un module de localisation du type SLAM (*Simultaneous Localization And Mapping* en anglais). La figure 4.1 montre un exemple type de robot *Pioneer 2-DX* avec télémètre laser.

Les robots mobiles partagent une grille d'occupation, établie par interprétation de la structure de l'environnement au cours de l'exécution de la mission d'exploration. La grille ainsi établie est utilisée comme une carte et guide la planification de chemin et l'évitement d'obstacles en temps réel. La planification de mouvements des robots est divisée en deux sous-parties : la planification globale et la planification locale. Nous utilisons l'algorithme *wavefront* [71] pour la planifica-

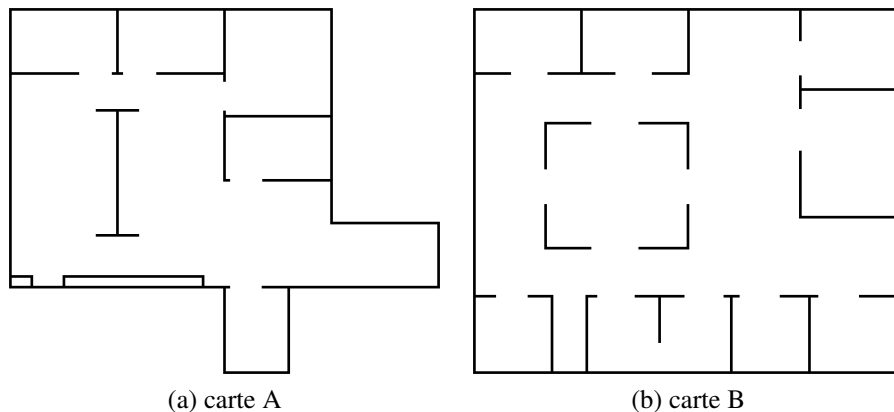
FIGURE 4.1 – Robot *Pioneer 2-DX* avec télémètre laser

FIGURE 4.2 – Deux cartes utilisées dans un problème d’exploration

tion globale et l’algorithme VFH+ [122] pour la planification locale, regroupant la recherche d’objectif et l’évitement d’obstacles proches. Le rapport entre le temps dans un monde réel et le temps de simulation est approximativement de 1 sur 3.

Pour évaluer notre approche, nous utilisons un nombre variable de robots dans différents environnements. Les résultats sont établis sur la base d’une comparaison avec une approche centralisée basée sur la *méthode hongroise* [69] dans laquelle la répartition des tâches est effectuée par un planificateur central. La figure 4.2 montre deux cartes utilisées pour définir deux espaces distincts. Ces deux espaces sont des espaces intérieurs clos structurés de 14 mètres de long sur 16 mètres de large. Un exemple d’exécution est présenté dans la figure 4.3.

Le tableau 4.1 (a) montre une comparaison des résultats entre notre approche

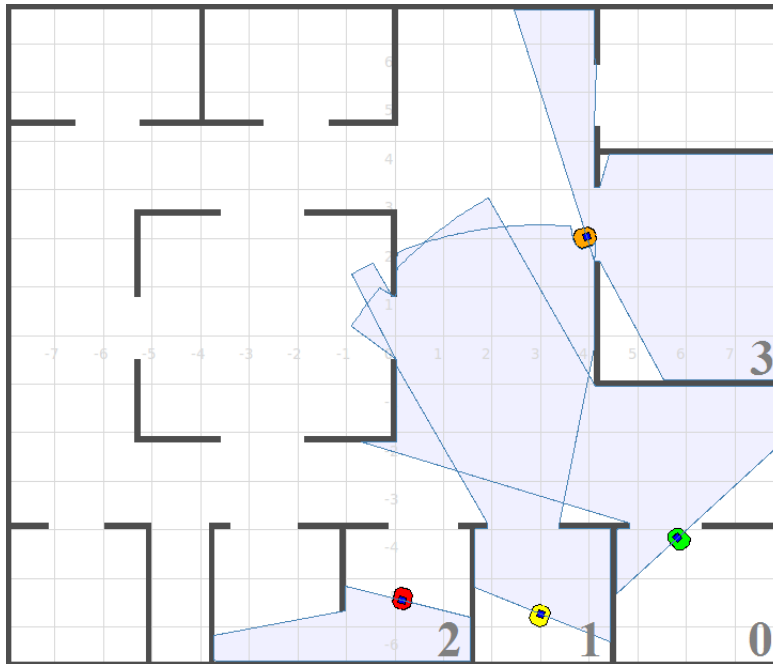


FIGURE 4.3 – Exemple d’exécution de l’approche contrarienne. Quatre robots collaborent dans l’exploration d’un environnement inconnu. Les nombres en noirs permettent de suivre les tâches actuellement en cours d’exécution. Le robot vert termine l’exploration de la salle 0 tout en se déplaçant vers une autre salle. Le robot jaune explore la salle 1. Le robot rouge explore la salle 2. Le robot orange se déplace vers la salle 3.

basée sur l’échange (décentralisée) avec l’approche basée sur la *méthode hongroise* (centralisée) sur la carte A. Les résultats sont évalués par la mesure du temps d’exécution des missions d’exploration. Chaque ensemble de données contient un temps moyen (en secondes) établi à partir de l’exécution de 10 missions (l’écart-type associé à chaque moyenne est indiqué entre parenthèses, à droite de chaque moyenne). La *méthode hongroise* est un moyen efficace de trouver une solution optimale pour un problème de répartition des tâches. Par conséquent, associée à une planification centralisée, elle définit un ensemble de résultats de référence permettant de mesurer les résultats de notre approche. Le tableau 4.1 (a) indique que la différence du temps d’achèvement des missions. Ces différences sont de 6,6% (soit 4,3 secondes) pour 2 robots, de 19,4% (soit 8,9 secondes) pour 3 robots et de 3,9% (soit 1,8 secondes) pour 4 robots.

Le tableau 4.1 (b) montre les résultats pour la carte B. La différence des temps d’exécution est de 6,3% (soit 4,5 secondes) pour 3 robots et de 7,4% (soit 4,6 secondes) pour 4 robots. Notre approche réalise un gain de temps de 14,2% (soit

TABLE 4.1 – Temps d'exécution des missions d'exploration

(a) carte A

Nombre de robots	Approche basée sur la méthode hongroise	Approche basée sur l'échange
2	65.20s (1.166)	69.50s (0.806)
3	45.90s (0.538)	54.80s (1.887)
4	45.90s (7.120)	47.70s (0.458)

(b) carte B

Nombre de robots	Approche basée sur la méthode hongroise	Approche basée sur l'échange
2	112.0s (3.317)	98.10s (1.300)
3	71.70s (1.487)	76.20s (3.027)
4	61.80s (1.249)	66.40s (1.020)

13,9 secondes) par rapport à l'approche de référence (*i.e.* par la *méthode hongroise*) dans le cadre d'une mission réalisée avec 2 robots. Ce gain de temps s'explique par la nature dynamique de l'environnement résultant des déplacements des robots, environnement dans lequel la planification (de tâches et de mouvements) est intrinsèquement difficile.

Une autre expérience a été menée pour évaluer la robustesse de notre approche. Le but de cette seconde expérience est de tester l'effet de la prise de décision d'un robot dans l'incertitude résultante d'une perte d'informations. Les justifications de cette perte sont les hypothétiques situations suivantes : endommagement de composants d'un robot, présence de barrières de communication (par expiration d'un délai de communication, par exemple conséquence des infrastructures d'exécution de la mission ou de la distance entre les robots), ou évènement imprévu aux circonstances difficiles. Pour une équipe composée de plusieurs robots reliés par un système de prise de décision décentralisé, chaque robot a besoin d'échanger des informations avec ses coéquipiers pour prendre une décision. Le problème de la perte d'informations influe sur les performances de décision. Les résultats présentés dans le tableau 4.2 montrent la robustesse de notre approche sur les carte A et B. La probabilité de perte d'informations (PPI) varie de 10%, 30% et 50%. Ces résultats présentent des temps moyen d'exécution en secondes, calculés sur l'exécution de 10 missions.

Le tableau 4.2 (a) montre pour la carte A : en cas de perte d'informations de 10%, le système conserve une bonne performance et les temps d'exécution des missions d'exploration sont prolongés tout en restant proches des résultats initiaux

TABLE 4.2 – Résultats avec PPI variable

(a) carte A

Nombre de robots	PPI 10%	PPI 30%	PPI 50%
2	70.40s	84.50s	85.10s
3	56.20s	63.00s	76.80s
4	50.80s	59.70s	73.90s

(b) carte B

Nombre de robots	PPI 10%	PPI 30%	PPI 50%
2	100.3s	141.3s	155.2s
3	82.00s	102.5s	124.6s
4	67.00s	83.00s	92.80s

(les différences entre eux sont de 0.9 secondes pour 2 robots, de 1.4 secondes pour 3 robots et de 3.1 secondes (4 robots); en cas de perte d'informations de 30%, la performance du système est abaissée et les temps d'exécution des missions d'exploration sont prolongés par les effets indésirables de la perte d'informations. En comparaison avec les résultats initiaux, les différences sont de 15.0 secondes pour 2 robots, de 8.2 secondes pour 3 robots et de 12.0 secondes pour 4 robots; en cas de perte d'informations de 50%, la performance du système diminue encore et les différences entre temps d'exécution des missions d'exploration et temps initiaux sont de 15.6 secondes pour 2 robots, de 22.9 secondes pour 3 robots et de 26.2 secondes pour 4 robots.

Le tableau 4.2 (b) montre pour la carte B : en cas de perte d'informations de 10%, les différences sont de 2.2 secondes pour 2 robots, de 5.8 secondes pour 3 robots et de 0.6 secondes pour 4 robots par rapport aux résultats initiaux; en cas de perte d'informations de 30%, les différences sont de 43.2 secondes pour 2 robots, de 26.3 secondes pour 3 robots et de 16.6 secondes pour 4 robots; en cas de perte d'informations de 50%, les différences sont de 57.1 secondes pour 2 robots, de 48.4 secondes pour 3 robots et de 26.4 secondes pour 4 robots.

D'après le tableau 4.2, l'efficacité d'une résolution multi-robots décentralisée pour un problème d'allocation de tâches dépend de la qualité des échanges d'informations entre les robots. Les temps d'exécution des missions d'exploration se prolongent conjointement avec l'augmentation de la probabilité de perte d'informations. En outre, ces augmentations sont considérées comme acceptables dans nos expériences, car sub-linéaire en la perte d'informations. Ce comportement démontre la bonne robustesse de notre approche. Ce comportement reste à améliorer dans les cas extrêmes (*i.e.* si aucune communication).

FIGURE 4.4 – Robot *Pioneer 2-DX* avec pince

4.2 Approche heuristique et transport de marchandises

L'expérience choisie pour évaluer l'approche heuristique basée sur un modèle empirique est le transport d'un certain nombre de marchandises d'une source vers une destination en utilisant un groupe de robots *Pioneer 2-DX*. Chaque robot est équipé d'une pince permettant de détecter une marchandise, de la ramasser, de la transporter et de la déposer. La capacité de charge de chaque robot est limitée à une seule unité de marchandise. Chaque robot peut se localiser à l'aide d'un module de localisation du type GPS (*Global Positioning System* en anglais). La figure 4.4 montre un exemple type de robot *Pioneer 2-DX* avec pince.

Nos expériences sont effectuées dans un espace clos avec 16 mètres de long et de 16 mètres de large contenant plusieurs obstacles fixes. Le rapport entre le temps dans un monde réel et le temps de simulation est de 1 sur 5. Les robots mobiles partagent une grille d'occupation commune avec des informations sur la structure de l'environnement. Cette grille est utilisée comme carte pour la planification de chemin et l'évitement d'obstacles en temps réel. Nous avons utilisé l'algorithme *wavefront* [71] pour la planification globale et l'algorithme ND [82] pour la planification locale. Nos expérimentations incluent un composant de planification centrale, capable de communiquer avec tous les robots à chaque instant.

L'évaluation de notre solution est réalisée dans quatre environnements expérimentaux de simulation, présentés dans la figure 4.5. Ces environnements sont représentatifs des relations de distance entre dépôt (noté D), source (notée S) et destination (notée K), c'est à dire des quatre relations suivantes : $KD > KS > SD$, $KD > SD > KS$, $KS > KD > SD$ et $KS > SD > KD$. Il existe deux autres relations : $SD > KD > KS$ et $SD > KS > KD$. Toutefois, ces deux possèdent des distances

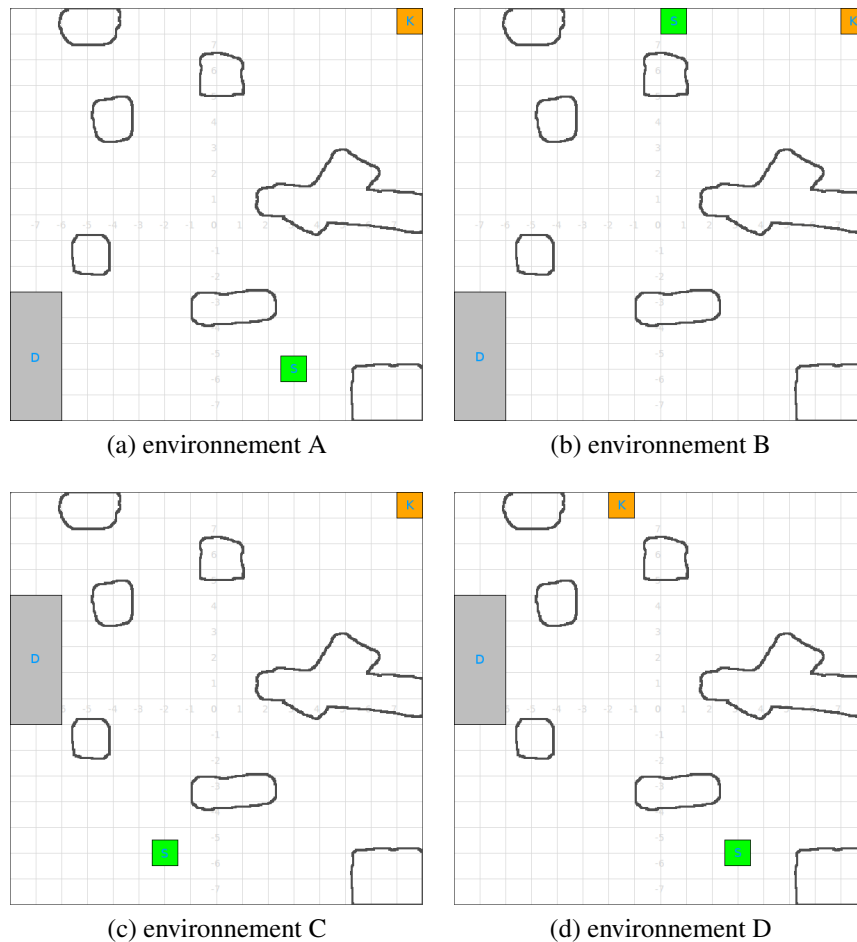


FIGURE 4.5 – Quatre environnements expérimentaux de simulation. La zone grise représente le dépôt (D), la zone verte représente la source (S) et la zone orange représente la destination (K). L'environnement A correspond au cas $KD > KS > SD$, l'environnement B au cas $KD > SD > KS$, l'environnement C au cas $KS > KD > SD$ et l'environnement D au cas $KS > SD > KD$.

importantes entre source et dépôt, qui sont illogiques en réalité.

La figure 4.6 est un exemple d'exécution d'une simulation, dans lequel le robot rouge est en attente au dépôt, le robot vert est en cours de déplacement pour le chargement d'une marchandise, le robot orange transporte une marchandise en direction de l'emplacement de livraison et le robot bleu rentre au dépôt (suite à l'achèvement récent d'un dépôt de marchandise).

Nous avons utilisé une équipe de robots pour effectuer plusieurs expériences et comparer notre approche heuristique à l'approche *re-planner* [126]. Le principe

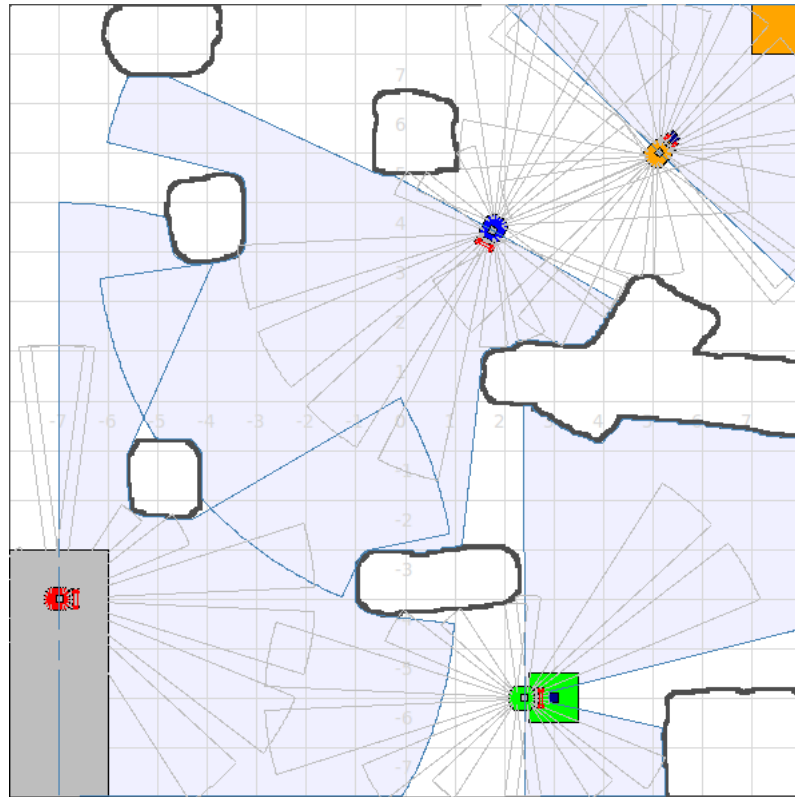


FIGURE 4.6 – Exemple d’exécution de l’approche heuristique. Quatre robots coopérants dans une mission de transport. Les marchandises à transporter sont représentées par des carrés bleu foncé. La zone grise représente le dépôt pour les robots. La zone verte représente un lieu de production de marchandises. La zone orange représente un site de livraison des marchandises. L’expérience présentée ici inclut une unique zone de production et une unique zone de livraison dans un souci de clarté des résultats.

de la solution présentée dans *re-planner* est d’affecter chaque marchandise au robot le plus proche de celle-ci. Le taux de production des marchandises varie aléatoirement de 4 à 12 secondes. Les résultats expérimentaux sont présentés dans la figure 4.7.

La figure 4.7 montre les résultats expérimentaux obtenus dans l’exécution de missions incluant quatre robots pour le transport de 50 marchandises. Les quatre histogrammes correspondent aux résultats des quatre environnements de simulation choisis. Chaque histogramme contient deux ensembles de données expérimentales correspondant à la durée totale du transport et la somme de l’énergie consommée par l’ensemble des robots en utilisant l’approche heuristique et en utilisant l’approche *re-planner*. Chaque mission est répétée 10 fois, pour évaluer

4.2. APPROCHE HEURISTIQUE ET TRANSPORT DE MARCHANDISES 75

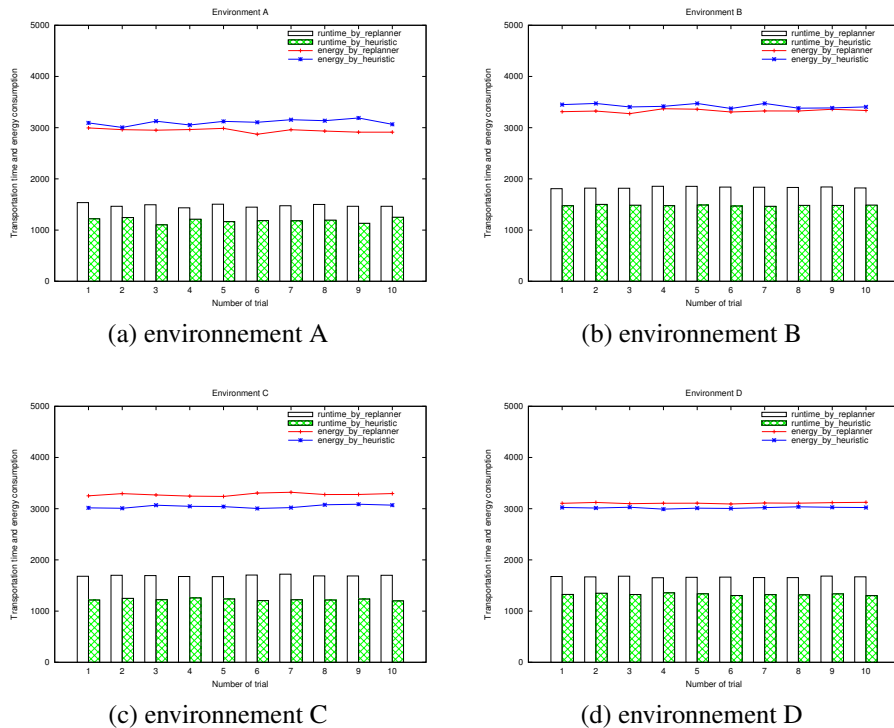


FIGURE 4.7 – Résultats comparés de l’approche heuristique et de l’approche *re-planner* dans les quatre environnements expérimentaux de simulation présentés dans la figure 4.5. Les résultats de l’approche *re-planner* correspondent aux bâtons blancs et aux lignes rouges. Les résultats de l’approche heuristique correspondent aux bâtons verts et aux lignes bleues. Chaque diagramme représente donc l’exécution de 10 missions identiques, différenciées selon l’axe des x . Selon l’axe des y , il est possible de lire le temps d’exécution de chaque mission sur les bâtons (blancs ou verts) et la consommation d’énergie sur les tracés (rouges ou bleus).

la stabilité et la robustesse de chaque solution. La figure 4.7 montre que le temps de transport est considérablement réduit avec notre approche et que la consommation d’énergie obtenue en utilisant notre approche est toujours maintenue à un faible niveau (avec une différence en moyenne inférieure à 4.8% de la consommation utilisée par l’approche *re-planner*, les consommations d’énergie sont jugées équivalentes.).

Un autre avantage important de notre approche heuristique réside dans une consommation d’énergie répartie uniformément entre les robots. Le tableau 4.3 présente le nombre moyen des tâches complétées et des énergies consommées par les robots dans l’environnement A. Le nombre de tâches complétées correspond au nombre des marchandises transportées. Les données pour les tâches complé-

TABLE 4.3 – Nombre moyen des tâches complétées et des énergies consommées dans l’environnement A

(a) approche re-planner

Numéro de robot	Tâches complétées	Énergies consommées
0 (robot rouge)	0	0
1 (robot orange)	19.7 (6.133)	1129.0 (343.6)
2 (robot bleu)	23.3 (1.952)	1360.0 (113.0)
3 (robot vert)	7.0 (6.618)	455.7 (368.7)

(b) approche heuristique

Numéro de robot	Tâches complétées	Énergies consommées
0 (robot rouge)	11.4 (1.497)	780.1 (112.1)
1 (robot orange)	13.0 (1.000)	789.4 (59.84)
2 (robot bleu)	13.4 (1.281)	792.8 (75.77)
3 (robot vert)	12.2 (0.980)	746.7 (66.86)

tées et les énergies consommées sont établies sur 10 exécutions (l’écart-type associé à chaque moyenne est indiqué entre parenthèses). La présence de valeurs nulles dans les résultats de l’approche *re-planner* laisse supposer l’adéquation d’un nombre de trois robots pour l’achèvement d’une mission dans cette expérience (conséquence de l’affectation d’aucune tâche au robot rouge). Les résultats de l’approche heuristique montre l’inverse, par le biais de meilleurs résultats avec quatre robots.

Ces expériences démontrent que deux facteurs principaux affectent les résultats d’une approche pour la résolution d’un problème de transport de marchandises multi-robots :

- Les relations entre dépôt, source et destination. Si la source et la destination sont fixées, alors le choix de la position de l’emplacement d’un dépôt influe sur les résultats. Dans la figure 4.7, l’histogramme (a) (*i.e.* $KD > KS > SD$) possède un temps de transport et une consommation d’énergie qui sont inférieurs aux valeurs de l’histogramme (b) (*i.e.* $KD > SD > KS$).
- Le nombre de robots constituant l’équipe multi-robots. Le tableau 4.3 montre les consommations moyennes d’énergie de chaque robot. Pour une quantité de robots trop faible, le temps d’exécution total de la mission sera étendu. Au contraire, une quantité trop importante de robots peut entraîner une surconsommation d’énergie. Par conséquent, la définition et le contrôle d’un nombre adéquate de robots au fil de l’exécution des missions est une considération importante.

4.3. APPROCHE PAR ÉCHANTILLONNAGE ET EXPLORATION MULTI-ROBOTS 77

Enfin, la figure 4.7 montre un temps total de transport proche de la moyenne dans chaque histogramme. La stabilité de cette valeur démontre une bonne qualité du service de transport ainsi achevé dans la maîtrise du temps de livraison. Il est important de noter que le transport des marchandises est réalisé avec une pince, ce qui implique des possibilités d'évènements indésirables dans le transport, dus des collisions dans le cas de vitesses de déplacement trop importantes. En cas de collision, selon les positions et selon la vitesse de chaque robot, les marchandises se trouvent endommagées ou perdues (par perte de préhension sur la marchandise). Ces collisions sont possibles avec des obstacles et avec d'autres robots. Par conséquent, afin d'assurer la sécurité de transport en évitant le dommage ou la perte des marchandises, la vitesse de chaque robot a été limitée jusqu'à 1,0 m/s.

4.3 Approche par échantillonnage et exploration multi-robots

L'expérience choisie pour évaluer l'approche par échantillonnage dédié à la décision est l'exploration d'un environnement inconnu intérieur en utilisant un groupe de robots *Pioneer 2-DX* que nous montrons dans la figure 4.1. Un serveur central de planification a la possibilité de communiquer avec tous les robots et d'attribuer des tâches d'exploration à chacun. Tous les robots partagent une grille d'occupation représentative de la structure de l'environnement au cours de leur mission d'exploration.

Les expériences impliquent deux à six robots placés dans deux environnements distincts. Une comparaison entre les résultats de notre approche et l'approche proposée par le *diagramme de Voronoï* est donnée. Les figures 3.3 (a) et (b) représentent deux cartes expérimentales utilisées dans nos simulations. Ces cartes sont des espaces clos avec 14 mètres de long et 16 mètres de large. Le rapport entre le temps de monde réel et le temps de simulation est de 1 sur 3. Un exemple d'exécution est présenté dans la figure 4.8.

Les résultats de nos expériences sont présentés dans la figure 4.9. Nous avons mesuré le gain en temps d'exploration pour notre approche par comparaison avec l'approche basée sur le *diagramme de Voronoï*. Dans chaque diagramme, l'abscisse représente la taille de l'équipe des robots, l'ordonnée représente le gain de temps en pourcentage de la durée totale d'exploration. La barre d'erreurs indique un intervalle de confiance de chaque gain avec un niveau de confiance de 95%. La figure 4.9 montre un gain de temps d'exploration de 5.8% à 16.1% sur la carte A et de 6.4% à 15.3% sur la carte B. Ces résultats démontrent l'efficacité de notre solution.

Nous avons également évalué le nombre moyen d'occurrences d'exclusion

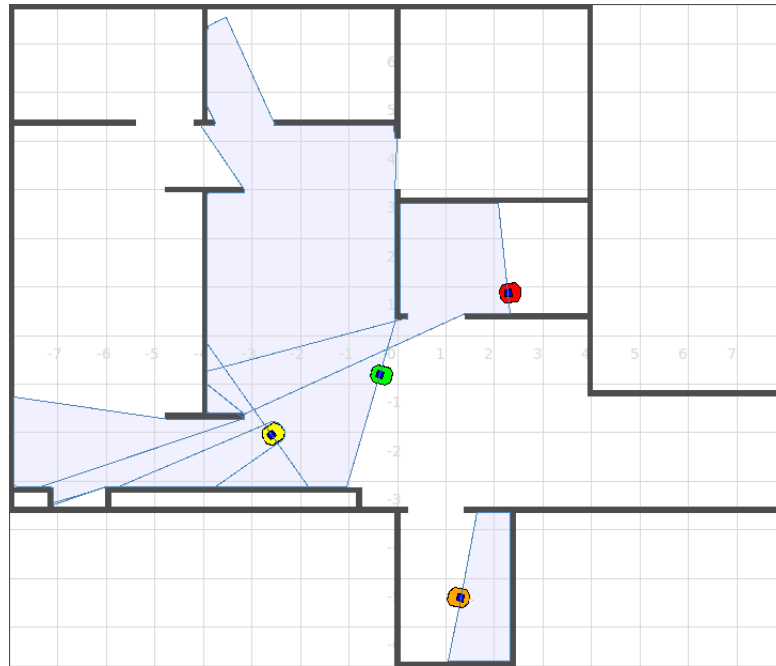


FIGURE 4.8 – Exemple d’exécution de l’approche par échantillonnage dédié à la décision. Quatre robots collaborent dans l’exploration d’un environnement inconnu. Le robot rouge et le robot orange explorent différentes chambres. Le robot vert et le robot jaune explorent les couloirs le long des chemins différents.

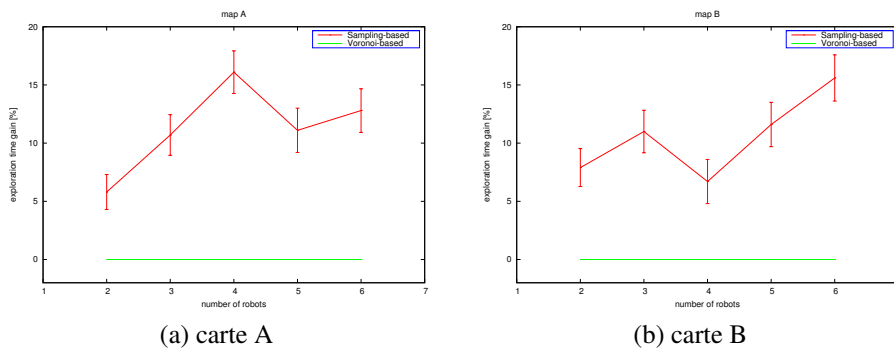


FIGURE 4.9 – Gain en temps dans un problème d’exploration de notre approche par échantillonnage par comparaison avec une approche basée sur le *diagramme de Voronoï*.

mutuelle de point de passage dans chaque carte comme indiqué dans le tableau 4.4. Ce tableau montre que le problème des situations d’attente est considérablement réduit en utilisant notre approche. Il est important de noter que contrairement

4.4. APPROCHE PAR ÉCHANTILLONNAGE ET TRANSPORT DE MARCHANDISES 79

TABLE 4.4 – Nombre des exclusions mutuelles dans un problème d’exploration

(a) carte A

Nombre de robots	2	3	4	5	6
Echantillonnage	2.5	2.9	4.3	3.1	3.4
Diagramme de Voronoï	6.2	6.7	11.6	8.1	10.2

(b) carte B

Nombre de robots	2	3	4	5	6
Echantillonnage	0.0	1.4	2.7	2.9	2.4
Diagramme de Voronoï	4.1	6.2	6.4	7.7	10.7

à l’approche basée sur l’utilisation du *diagramme de Voronoï*, notre approche permet de planifier des chemins séparés pour des robots dans un couloir.

4.4 Approche par échantillonnage et transport de marchandises

L’expérience choisie pour évaluer l’approche par échantillonnage adaptatif (ACS-PRM) est le transport d’un certain nombre de marchandises d’une source vers de multiples destinations en utilisant un groupe de robots *Pioneer 2-DX* que nous montrons dans la figure 4.4. La figure 4.10 montre deux cartes utilisées, qui sont des environnements structurés. Pour chaque carte, la zone verte indique la position d’origine des marchandises, et les zones jaunes représentent les destinations qui sont toujours placées dans les chambres. La carte A dispose de huit chambres donc huit destinations. La carte B dispose de sept chambres donc sept destinations. La taille de l’équipe de transport varie de deux à huit robots. Pour chaque taille d’équipe, 10 essais expérimentaux sont effectués pour une mission de transport de 50 marchandises. L’objectif de la mission est de transporter les marchandises en direction de chaque chambre. L’objectif est un transport équilibré vers chaque destination. Le rapport entre le temps dans un monde réel et le temps de simulation est approximativement de 1 sur 1.

Dans les expériences, on suppose qu’il existe un serveur central capable de communiquer vers tous les robots et d’affecter des tâches de transport à chaque robot. Une tâche de transport est définie par une position d’origine et une destination. Nous avons également supposé que tous les robots partagent une grille commune et chacun possède d’une information complète sur tous les autres robots. Cette hypothèse est importante pour la planification de chemin et pour l’évi-

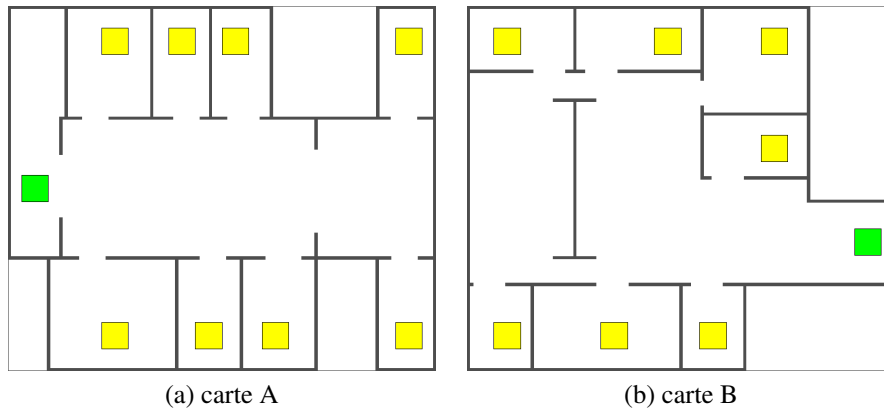


FIGURE 4.10 – Deux cartes utilisées dans un problème de transport

tement d'obstacles en temps réel.

La figure 4.11 est un exemple d'exécution d'une simulation, dans lequel le robot rouge réalise une livraison, le robot vert est sur son chemin de retour après terminaison de livraison, le robot jaune se déplace pour charger une marchandise et le robot orange transporte une marchandise vers une destination.

Les résultats de nos expériences sont présentés dans la figure 4.12. Nous avons mesuré le gain en temps de transport par comparaison avec l'approche générale qui utilise l'algorithme *wavefront* [71] pour la planification de mouvements globale et l'algorithme VFH+ [122] pour la planification de mouvements locale. Dans chaque diagramme, l'abscisse représente la taille de l'équipe des robots, l'ordonnée représente le gain de temps en pourcentage de la durée totale de transport. La barre d'erreur indique un intervalle de confiance de chaque gain avec un niveau de confiance de 95%. La figure 4.12 montre un gain de temps de transport de 6,7% à 12,2% sur la carte A et de 6,1% à 12,0% sur la carte B peut être obtenue en utilisant notre approche. Ces résultats démontrent l'efficacité de notre solution.

L'approche ACS-PRM permet de construire des feuilles de route dans un temps réduit. Les expériences montrent que les temps de cartographie sont respectivement de 0,321 secondes et de 0,329 secondes pour la carte A et la carte B avec 200 000 échantillons aléatoires sur une moyenne de 10 essais.

Le tableau 4.5 présente le nombre moyen d'occurrences d'exclusion mutuelle de point de passage dans chaque carte. Ce tableau montre que le problème des situations d'attente est considérablement réduit en utilisant l'approche ACS-PRM.

4.4. APPROCHE PAR ÉCHANTILLONNAGE ET TRANSPORT DE MARCHANDISES 81

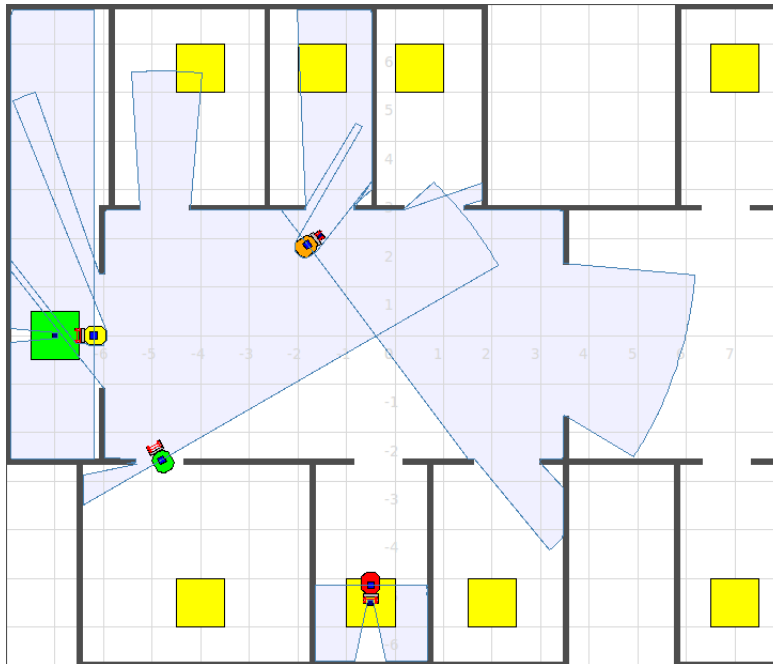


FIGURE 4.11 – Exemple d'exécution de l'approche par échantillonnage adaptatif. Quatre robots coopérants dans une mission de transport. La pièce bleu foncé identifie les marchandises à transporter. La zone verte représente la position d'origine des marchandises. La zone jaune représente la destination, qui correspond à chaque chambre, où les marchandises doivent être livrées par les robots.

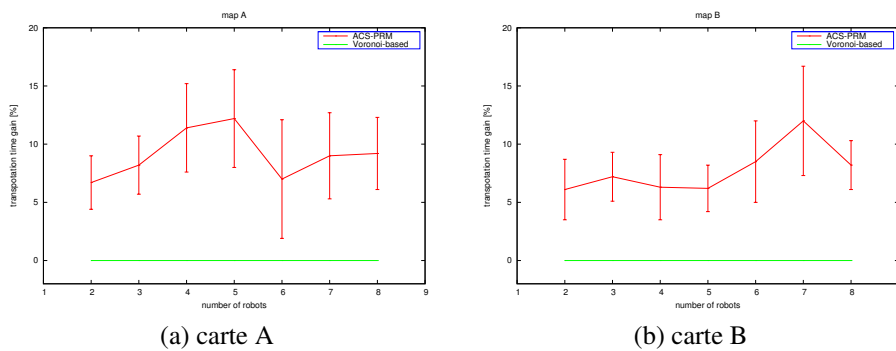


FIGURE 4.12 – Gain en temps dans un problème de transport de notre approche ACS-PRM par comparaison avec l'approche générale.

TABLE 4.5 – Nombre des exclusions mutuelles dans un problème de transport

(a) carte A

Nombre de robots	2	3	4	5	6	7	8
Approche ACS-PRM	1.6	3.1	6.4	7.5	10.0	11.1	16.2
Approche générale	15.3	18.7	26.0	26.8	19.9	23.7	27.2

(b) carte B

Nombre de robots	2	3	4	5	6	7	8
Approche ACS-PRM	3.8	4.3	7.1	14.9	12.8	10.3	16.7
Approche générale	17.1	19.2	19.0	26.5	27.4	27.0	29.9

4.5 Conclusion

L'approche contrarienne centrée sur les échanges a été testée et évaluée dans un problème d'exploration. Une première expérience en simulation expose les bons résultats de notre solution par comparaison avec la *méthode hongroise*. Une seconde expérience en simulation démontre la robustesse de notre solution dans le cas d'une perte d'informations. Nos résultats montrent que, l'approche contrarienne centrée sur les échanges propose des solutions optimales en temps pour les problèmes d'exploration multi-robots en utilisant un système décentralisé.

L'approche heuristique basée sur un modèle empirique a été testée et évaluée en simulation dans un problème de transport de marchandises. Les résultats démontrent que notre approche propose des résultats inférieurs en temps et identiques en consommation d'énergie, par rapport à l'approche *re-planner*.

L'approche par échantillonnage dédié à la décision a été testée et évaluée en simulation dans un problème d'exploration. Les résultats démontrent que le temps global d'achèvement d'une mission avec notre solution est considérablement inférieur aux résultats proposés par une approche fondée sur l'utilisation du *diagramme de Voronoï*.

L'approche par échantillonnage adaptatif a été testée et évaluée en simulation dans un problème de transport de marchandises. Les résultats montrent qu'en utilisant notre approche, le temps total nécessaire pour achever la mission de transport est considérablement réduit par rapport à l'approche générale.

Chapitre 5

Conclusion et perspectives

Les travaux résumés dans ce mémoire présentent quatre nouvelles méthodes pour la coordination multi-robots. Deux de ces contributions sont destinées à la planification de tâches et deux de ces contributions concernent la planification de mouvements. Ces contributions proposent des solutions pour les architectures multi-robots centralisées et décentralisées.

L'étude des problèmes de coordination multi-robots du point de vue de la planification de tâches nous a permis de proposer deux nouvelles solutions. La première solution est une approche contrarienne centrée sur les échanges conçue pour la répartition des tâches décentralisée dynamique basée sur des règles commerciales. Cette approche simule la relation entre acheteurs et vendeurs dans un système commercial et propose l'utilisation d'un mécanisme d'*offre non sollicitée*. Après une première phase de définition de rôles, les robots acheteurs agissent et les robots vendeurs répartissent les tâches. La deuxième solution est une approche heuristique conçue pour traiter le problème de la répartition des tâches de transport de marchandises. Le problème considéré est défini par un site producteur, une zone destination, un taux de production inconnu, un nombre de robots fixé doté de pince pour le transport des marchandises et un dépôt permettant de placer les robots en attente de tâche. Notre solution minimise le temps total de transport tout en gardant une faible consommation énergétique des robots. Cette solution est réalisée par estimation du taux de production de marchandises avec une fonction heuristique basée sur un modèle empirique. Cette solution est réalisée dans une architecture centralisée, permettant l'attribution de marchandises aux robots impliqués.

L'étude des problèmes de coordination multi-robots du point de vue de la planification de mouvements nous a permis également de proposer deux solutions supplémentaires. Cette partie de nos travaux formalise le problème des situations d'attente entre robots dans l'exécution d'actions dans un espace commun. Ayant présenté les solutions de la littérature, nous avons proposé deux nouvelles solu-

tions basées sur la méthode d'échantillonnage. L'une agrège plusieurs méthodes existantes placées dans une approche centralisée créant un échantillonnage itératif atténuant les problèmes de congestion, de collision et d'interblocage entre robots. La seconde propose une méthode d'échantillonnage permettant de créer des feuilles de route probabiliste adaptée à la complexité de l'espace à traverser. La solution proposée permet de construire des chemins réduisant les probabilités d'interférence entre les chemins des robots et par la même, les occurrences de situation d'attente lors de l'exécution.

Ces résultats ont été validés et testés dans des expériences comparatives avec des approches de référence avec le simulateur *Player/Stage* en utilisant différentes instrumentations de la base mobile *Pioneer 2-DX*. L'utilisation de ce simulateur nous permet de présenter de bons résultats dans un environnement proche du monde réel. Les résultats montrent des temps de transport et d'exploration inférieurs aux travaux antérieurs du domaine. Les résultats montrent également la consommation d'énergie des robots impliqués et exposent des propriétés de stabilité par équi-répartition entre les robots impliqués. L'originalité de nos travaux réside dans la proposition d'un ensemble de solutions pour un large spectre d'applications centralisées, décentralisées, orientées transport et orientées exploration. L'étude de la stabilité dans un système multi-robots reste une perspective intéressante, dans la formulation de nouvelles approches de coordination de mouvements et de tâches.

Une autre perspective de nos travaux est de placer nos approches dans le contexte des problèmes de planification de mouvements des robots humanoïdes et aux modélisations bio-informatique. Avec des modèles articulés complexes, ces problèmes possèdent des espaces de configurations de grandes dimensions et de fait, sont explicitement non-représentables. Cette non-possibilité de représentation explicite, due à la complexité de C , rends le problème de coordination de plusieurs agents (robots humanoïdes ou molécules) difficiles. Pour résoudre ce problème de planification dans C sans se restreindre à une planification approximative dans un sous-espace de W , l'utilisation de notre approche ACS-PRM permettra d'établir dynamiquement les espaces libres et de coordonner dynamiquement les acteurs dans des espaces étroits. Ces espaces sont un point clé de l'utilisation des robots dans des tâches journalières humaines et dans des interactions entre humains et robots. L'application de l'approche ACS aux algorithmes RRT peut également être une solution.

Issue des travaux sur l'auto-organisation dans les systèmes multi-agents, une autre perspective de nos travaux se situe dans l'étude d'algorithmes de structuration et d'organisation des robots en sous-groupes pour la réalisation de la mission de transport et d'exploration. L'ajout de caractères différents au sein des robots (tels que des consommations d'énergie différentes) créant des groupes de robots hétérogènes, impliquera des variations plus importante dans les stratégies de ré-

partition. L'ajout du caractère incertain de l'exploration est également une des sources possibles des variations à réaliser, par implication des risques d'échec de la mission pour l'ensemble du système multi-robots. Afin de généraliser ces variations, l'étude de la notion d'adaptation au sens des systèmes multi-agents pourrait être une source d'inspiration intéressante.

Annexe A

Exemples de solutions pour la gestion des exclusions mutuelles aux points de passage

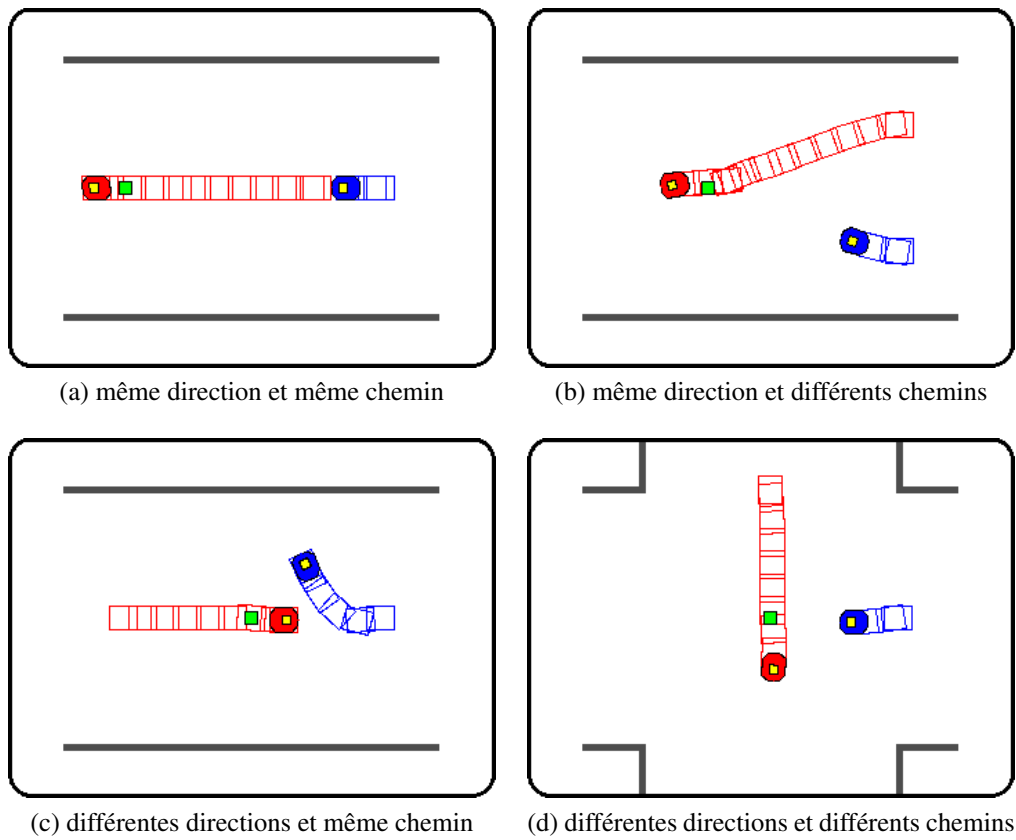


FIGURE A.1 – Trajectoires de robots obtenues avec l’algorithme OBO

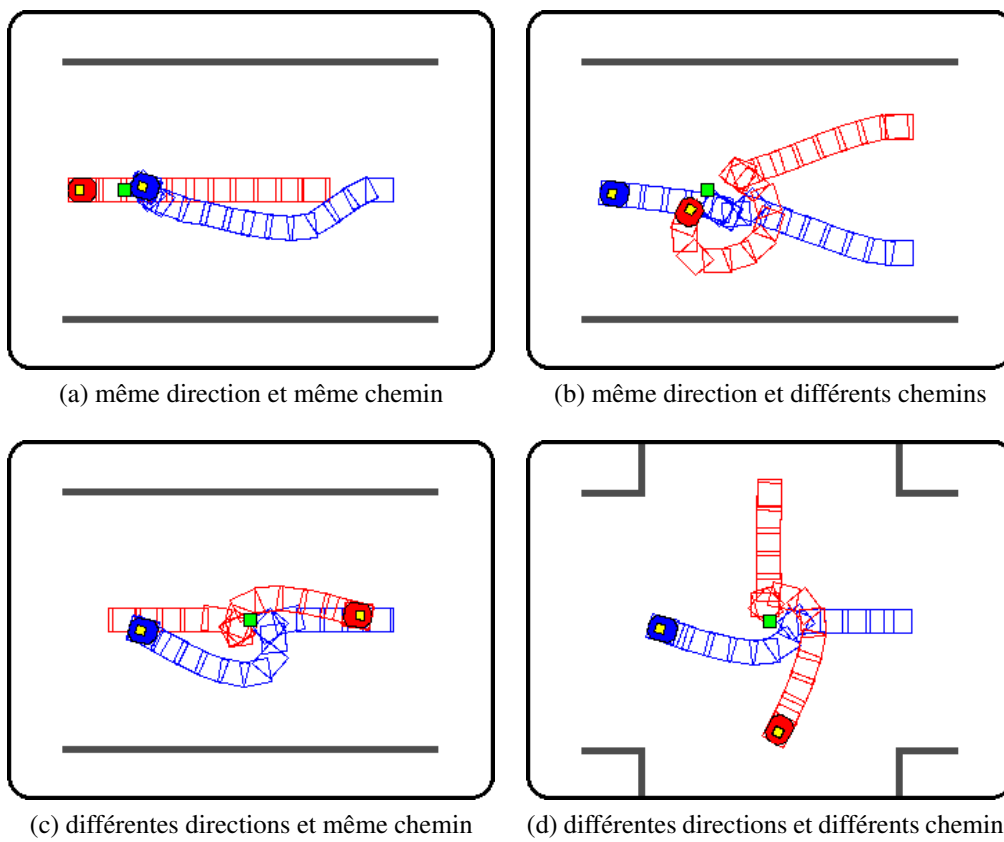


FIGURE A.2 – Trajectoires de robots obtenues avec l’algorithme VFH+

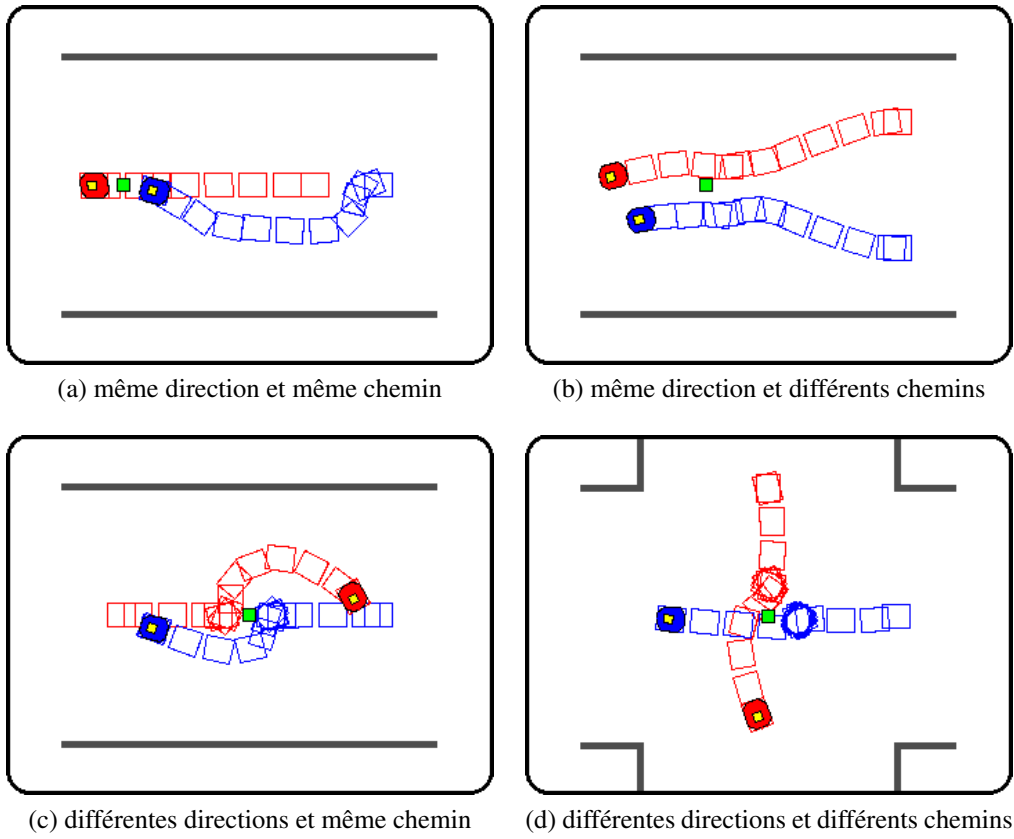


FIGURE A.3 – Trajectoires de robots obtenues avec l’algorithme ND

Annexe B

Articles publiés

– Articles de conférence avec comité de lecture

1. Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. Multi-robot heuristic goods transportation. In *Proceedings of the 6th IEEE International Conference on Intelligent Systems (IS 2012)*, pages 409-414, Sofia, Bulgaria, September 2012.
2. Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. On the problem of task planning in multi-robot systems. In *Proceedings of the 9th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2012)*, pages 423-426, Rome, Italy, July 2012.
3. Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. ACS-PRM : Adaptive cross sampling based probabilistic roadmap for multi-robot motion planning. In *Proceedings of the 12th International Conference on Intelligent Autonomous Systems (IAS 2012)*, pages 843-851, Jeju Island, Korea, June 2012.
4. Zhi Yan, Nicolas Jouandeau and Arab Ali Cherif. Multi-robot decentralized exploration using a trade-based approach. In *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2011)*, pages 99-105, Noordwijkerhout, Netherlands, July 2011.
5. Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. Sampling-based multi-robot exploration. In *Proceedings of the Joint 41st International Symposium on Robotics and 6th German Conference on Robotics (ISR/ROBOTIK 2010)*, pages 44-49, Munich, Germany, June 2010.

Bibliographie

- [1] <http://marsrovers.jpl.nasa.gov/mission/status.html#spirit>. 5
- [2] <http://www.grappa.univ-lille3.fr/icga>. 8
- [3] <http://www.robocup.org>. 8
- [4] Rachid Alami, Sara Fleury, Matthieu Herrb, Félix Ingrand, and Frédéric Robert. Multi-robot cooperation in the MARTHA project. *IEEE Robotics and Automation Magazine*, 5(1) :36–47, March 1998. 8, 16, 37
- [5] Nancy M. Amato, O. Burchan Bayazit, Lucia K. Dale, Christopher Jones, and Daniel Vallejo. OBPRM : An obstacle-based prm for 3d workspaces. In *Proceedings of the Workshop on Algorithmic Foundations of Robotics (WAFR'98)*, pages 155–168, Houston, TX, USA, March 1998. 60
- [6] Tamio Arai, Enrico Pagello, and Lynne E. Parker. Editorial : Advances in multi-robot systems. *IEEE Transactions on Robotics and Automation*, 18(5) :655–661, 2002. 3
- [7] Hajime Asama, Akihiro Matsumoto, and Yoshiki Ishida. Design of an autonomous and distributed robot system : ACTRESS. In *Proceedings of the 1989 IEEE/RSJ International Workshop on Intelligent Robots and Systems (IROS'89)*, pages 283–290, Tsukuba, Japan, September 1989. 2, 7
- [8] Stephen Balakirsky, Stefano Carpin, Alexander Kleiner, Michael Lewis, Arnoud Visser, Jijun Wang, and Vittorio Amos Ziparo. Towards heterogeneous robot teams for disaster mitigation : Results and performance metrics from robocup rescue. *Journal of Field Robotics*, 24(11–12) :943–967, November 2007. 8, 29
- [9] Tucker Balch and Ronald C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1(1) :27–52, 1994. 14
- [10] Tucker Balch and Ronald C. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation*, 14(6) :926–939, December 1998. 13

- [11] D. P. Barnes and J. O. Gray. Behaviour synthesis for co-operant mobile robot control. In *Proceedings of the 1991 International Conference on Control (Control'91)*, pages 1135–1140, Edinburgh, UK, March 1991. 7
- [12] Joseph L. Baxter, Edmund K. Burke, Jonathan M. Garibaldi, and Mark Norman. Shared potential fields and their place in a multi-robot co-ordination taxonomy. *Robotics and Autonomous Systems*, 57(10) :1048–1055, October 2009. 7, 8, 29
- [13] Halûk Bayram and H. Isil Bozma. Multi-robot navigation with limited communication - deterministic vs game-theoretic networks. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'10)*, pages 1825–1830, Taipei, Taiwan, October 2010. 14
- [14] Maren Bennewitz, Wolfram Burgard, and Sebastian Thrun. Optimizing schedules for prioritized path planning of multi-robot systems. In *Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA'01)*, pages 271–276, Seoul, Korea, May 2001. 21, 25
- [15] Mary Berna-Koes, Illah Nourbakhsh, and Katia Sycara. Communication efficiency in multi-agent systems. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*, pages 2129–2134, New Orleans, LA, USA, April 2004. 14
- [16] Antonio Bicchi, Antonio Danesi, Gianluca Dini, Silvio Porta, Lucia Pallottino, Ida Savino, and Riccardo Schiavi. Heterogeneous wireless multirobot system. *IEEE Robotics and Automation Magazine*, 15(1) :62–70, March 2008. 7, 14, 25
- [17] Silvia C. Botelho and Rachid Alami. M+ : A scheme for multi-robot cooperation through negotiated task allocation and achievement. In *Proceedings of the 1999 IEEE International Conference on Robotics and Automation (ICRA'99)*, pages 1234–1239, Detroit, MI, USA, May 1999. 2, 16, 17, 24
- [18] James Bruce and Manuela Veloso. Real-time randomized path planning for robot navigation. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, pages 2383–2388, Lausanne, Switzerland, September 2002. 22
- [19] Joachim Buhmann, Wolfram Burgard, Armin B. Cremers, Dieter Fox, Thomas Hofmann, Frank E. Schneider, Jiannis Strikos, and Sebastian Thrun. The mobile robot RHINO. *AI Magazine*, 16(2) :31–38, 1995. 4
- [20] Wolfram Burgard, Mark Moors, Dieter Fox, Reid Simmons, and Sebastian Thrun. Collaborative multi-robot exploration. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA'00)*, pages 476–481, San Francisco, CA, USA, April 2000. 5, 8, 30

- [21] Philippe Caloud, Wonyun Choi, Jean-Claude Latombe, Claude Le Pape, and Mark Yim. Indoor automation with many mobile robots. In *Proceedings of the 1990 IEEE/RSJ International Workshop on Intelligent Robots and Systems (IROS'90)*, pages 67–72, Ibaraki, Japan, July 1990. 2
- [22] Y. Uny Cao, Alex S. Fukunaga, and Andrew B. Kahng. Cooperative mobile robotics : Antecedents and directions. *Autonomous Robots*, 4(1) :7–27, 1997. 2, 13
- [23] Luiz Chaimowicz, Mario F. M. Campos, and Vijay Kumar. Dynamic role assignment for cooperative robots. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 293–298, Washington, DC, USA, May 2002. 17
- [24] Jianping Chen, Yimin Yang, and Liang Wei. Research on the approach of task decomposition in soccer robot system. In *Proceedings of the 1st International Conference on Digital Manufacturing and Automation (ICDMA'10)*, pages 284–289, Changsha, China, December 2010. 16
- [25] Arab Ali Cherif. Collective behavior for a micro-colony of robots. In *Proceedings of the Second European Conference on Artificial Life (ECAL'93)*, pages 1–10, Brussels, Belgium, May 1993. 1
- [26] Arab Ali Cherif. A constitution and an economic model for the organisation and emergence of collective behaviour in a colony of robots. In *Proceedings of From Perception to Action conference*, pages 334–337, Lausanne, Switzerland, September 1994. 1
- [27] Howie Choset and Joel Burdick. Sensor-based exploration : The hierarchical generalized voronoi graph. *The International Journal of Robotics Research*, 19(2) :96–125, February 2000. 51
- [28] Christopher M. Clark. Probabilistic road map sampling strategies for multi-robot motion planning. *Robotics and Autonomous Systems*, 53(3–4) :244–264, 2005. 47
- [29] Christopher M. Clark, Stephen M. Rock, and Jean-Claude Latombe. Motion planning for multiple mobile robots using dynamic networks. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA'03)*, pages 4222–4227, Taipei, Taiwan, September 2003. 23, 24
- [30] Anthony Cowley, Camillo J. Taylor, and Ben Southall. Rapid multi-robot exploration with topometric maps. In *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA'11)*, pages 1044–1049, Shanghai, China, May 2011. 8

- [31] Torbjorn S. Dahl, Maja J. Matarić, and Gaurav S. Sukhatme. Multi-robot task allocation through vacancy chain scheduling. *Robotics and Autonomous Systems*, 57(6–7) :674–687, June 2009. 7, 16, 18, 38
- [32] M. Bernardine Dias, Robert Zlot, Nidhi Kalra, and Anthony Stentz. Market-based multirobot coordination : A survey and analysis. Technical Report CMU-RI-TR-05-13, Carnegie Mellon University, Pittsburgh, PA, USA, April 2005. 15
- [33] Edsger W. Dijkstra. Cooperating sequential processes. In F. Genuys, editor, *Programming Languages : NATO Advanced Study Institute*, pages 43–112. Academic Press, 1968. 9
- [34] Marco Dorigo and Thomas Stützle. *Ant Colony Optimization*. MIT Press, 2004. 7
- [35] Gregory Dudek, Michael R. M. Jenkin, Evangelos Miliotis, and David Wilkes. A taxonomy for multi-agent robotics. *Autonomous Robots*, 3(4) :375–397, 1996. 2, 5
- [36] Alessandro Farinelli, Luca Iocchi, and Daniele Nardi. Multi-robot systems : A classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B : Cybernetics*, 34(5) :2015–2028, 2004. 3, 8, 13
- [37] John T. Feddema, Chris Lewis, and David A. Schoenwald. Decentralized control of cooperative robotic vehicles : Theory and application. *IEEE Transactions on Robotics and Automation*, 18(5) :852–864, 2002. 24
- [38] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 8(3) :325–344, 2000. 5, 7
- [39] Toshio Fukuda, Tsuyoshi Ueyama, Yoshio Kawauchi, and Fumihito Arai. Concept of cellular robotic system (CEBOT) and basic strategies for its realization. *Computers and Electrical Engineering*, 18(1) :11–39, 1992. 2, 24
- [40] Brian P. Gerkey and Maja J Matarić. Murdoch : Publish/subscribe task allocation for heterogeneous agents. In *Proceedings of the fourth international conference on Autonomous agents (Agents’00)*, pages 203–204, Barcelona, Spain, June 2000. 2, 7
- [41] Brian P. Gerkey and Maja J. Matarić. Principled communication for dynamic multi-robot task allocation. In *Proceedings of the 7th International Symposium On Experimental Robotics (ISER’00)*, pages 353–362, Honolulu, HI, USA, December 2000. 13

- [42] Brian P. Gerkey and Maja J. Matarić. Sold! : Auction methods for multi-robot coordination. *IEEE Transactions on Robotics and Automation*, 18(5) :758–768, October 2002. 10, 17, 18, 25, 28, 29, 30
- [43] Brian P. Gerkey and Maja J. Matarić. Multi-robot task allocation : Analyzing the complexity and optimality of key architectures. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA'03)*, pages 3862–3868, Taipei, Taiwan, September 2003. 16, 17
- [44] Brian P. Gerkey and Maja J. Matarić. Are (explicit) multi-robot coordination and multi-agent coordination really so different? In *Proceedings of the AAAI Spring Symposium 2004 on Bridging the Multi-Agent and Multi-Robotic Research Gap*, pages 1–3, Palo Alto, CA, USA, March 2004. 12
- [45] Brian P. Gerkey and Maja J. Matarić. A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 23(9) :939–954, September 2004. 16, 17
- [46] Brian P. Gerkey, Richard T. Vaughan, and Andrew Howard. The Player/S-tage project : Tools for multi-robot and distributed sensor systems. In *Proceedings of the 11th International Conference on Advanced Robotics (ICAR'03)*, pages 317–323, Coimbra, Portugal, June 2003. 1, 67
- [47] Dani Goldberg, Vincent Cicirello, M Bernardine Dias, Reid Simmons, Stephen Smith, and Anthony Stentz. Market-based multi-robot planning in a distributed layered architecture. In *Proceedings from the 2003 International Workshop on Multi-Robot Systems*, pages 27–38, Washington, D.C., USA, March 2003. 17
- [48] David Gouaillier, Vincent Hugel, Pierre Blazevic, Chris Kilner, Jérôme Monceaux, Pascal Lafourcade, Brice Marnier, Julien Serre, and Bruno Maisonnier. Mechatronic design of NAO humanoid. In *Proceedings of the 2009 IEEE International Conference on Robotics and Automation (ICRA'09)*, pages 769–774, Kobe, Japan, May 2009. 5
- [49] Zahia Guessoum, Jean-Pierre Briot, S. Charpentier, Olivier Marin, and Pierre Sens. A fault-tolerant multi-agent framework. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'02)*, pages 672–673, Bologna, Italy, July 2002. 28
- [50] Zahia Guessoum, Mikal Ziane, and Nora Faci. Monitoring and organizational-level adaptation of multi-agent systems. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'02)*, pages 514–521, New York, NY, USA, August 2004. 3
- [51] Julien Guitton, Jean-Loup Farges, and Raja Chatila. Cell-RRT : Decomposing the environment for better plan. In *Proceedings of the 2009 IEEE/RSJ*

- International Conference on Intelligent Robots and Systems (IROS'09)*, pages 5776–5781, St. Louis, MO, USA, October 2009. 60
- [52] Yi Guo and Lynne E. Parker. A distributed and optimal motion planning approach for multiple mobile robots. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 2612–2619, Washington, DC, USA, May 2002. 21
- [53] Hossam Hanna. Decentralized approach for multi-robot task allocation problem with uncertain task execution. In *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 535–540, Alberta, Canada, August 2005. 18
- [54] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2) :100–107, 1968. 21, 31, 46, 60
- [55] A. Dominik Haumann, Kim D. Listmann, and Volker Willert. Discoverage : A new paradigm for multi-robot exploration. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA'10)*, pages 929–934, Anchorage, AK, USA, May 2010. 8
- [56] Noam Hazon, Fabrizio Mieli, and Gal A. Kaminka. Towards robust on-line multi-robot coverage. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA'06)*, pages 1710–1715, Orlando, FL, USA, May 2006. 21
- [57] Geoffrey Hollinger, Sanjiv Singh, Joseph Djughash, and Athanasios Kehagias. Efficient multi-robot search for a moving target. *The International Journal of Robotics Research*, 28(2) :201–219, February 2009. 14
- [58] Gwo Jen Hwang and Shian Shyong Tseng. A heuristic task assignment algorithm to maximize reliability of a distributed system. *IEEE Transactions Reliability*, 42(3) :408–415, September 1993. 38
- [59] Luca Iocchi, Daniele Nardi, and Massimiliano Salerno. Reactivity and de-liberation : a survey on multi-robot systems. *Lecture Notes in Computer Science*, 2103 :9–32, 2001. 12
- [60] Markus Jäger and Bernhard Nebel. Decentralized collision avoidance, deadlock detection, and deadlock resolution for multiple mobile robots. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'01)*, pages 1213–1219, Maui, HI, USA, October 2001. 10, 25
- [61] James S. Jennings, Greg Whelan, and William F. Evans. Cooperative search and rescue with a team of mobile robots. In *Proceedings of the 8th International Conference on Advanced Robotics (ICAR'97)*, pages 193–200, Monterey, CA, USA, July 1997. 8, 29

- [62] Nicolas Jouandeau. *Algorithmique de la planification de mouvement probabiliste pour un robot mobile*. PhD thesis, Paris 8 University, December 2004. 60
- [63] Nicolas Jouandeau, Vincent Hugel, and Zhi Yan. L3M-SIM team description. In *RoboCup 2012, 3D Soccer Simulation League*, Mexico City, Mexico, 2012. 8
- [64] Shin Kato, Sakae Nishiyama, and Jun'ichi Takeno. Coordinating mobile robots by applying traffic rules. In *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'92)*, pages 1535–1541, Raleigh, NC, USA, July 1992. 12, 58
- [65] Lydia E. Kavraki, Mihail N. Kolountzakis, and Jean-Claude Latombe. Analysis of probabilistic roadmaps for path planning. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation (ICRA'96)*, pages 3020–3025, Minneapolis, MN, USA, April 1996. 22, 51, 53, 60
- [66] Eric Klavins. Communication complexity of multi-robot systems. *Algorithmic Foundations of Robotics V*, 7 :275–292, 2002. 13
- [67] Jonathan Ko, Benjamin Stewart, Dieter Fox, Kurt Konolige, and Benson Limketkai. A practical, decision-theoretic approach to multi-robot mapping and exploration. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'03)*, pages 3232–3238, Las Vegas, NV, USA, October 2003. 57
- [68] C. Ronald Kube and Eric Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30(1–2) :85–101, 2000. 10
- [69] Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1) :83–97, 1955. 31, 51, 57, 68
- [70] Benjamin Kuipers and Yung-Tai Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systems*, 8(1–2) :47–63, November 1991. 51
- [71] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991. 20, 33, 47, 60, 67, 72, 80
- [72] Steven M. LaValle. Rapidly-exploring random trees : A new tool for path planning. Technical Report TR 98-11, Iowa State University, Ames, IA, USA, October 1998. 22, 60
- [73] Ryan Luna and Kostas E. Bekris. Efficient and complete centralized multi-robot path planning. In *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'11)*, pages 3268–3275, San Francisco, CA, USA, September 2011. 11, 24

- [74] Chaomin Luo, Simon X. Yang, and Deborah A. Stacey. Real-time path planning with deadlock avoidance of multiple cleaning robots. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA'03)*, pages 4080–4085, Taipei, Taiwan, September 2003. 11
- [75] Raj Madhavan, Kingsley Fregene, and Lynne E. Parker. Distributed heterogeneous outdoor multi-robot localization. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 374–381, Washington, DC, USA, May 2002. 5, 7, 13
- [76] Leandro Soriano Marcolino and Luiz Chaimowicz. Traffic control for a swarm of robots : Avoiding group conflicts. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, pages 1949–1954, St. Louis, MO, USA, October 2009. 11, 25
- [77] Leandro Soriano Marcolino and Luiz Chaimowicz. Traffic control for a swarm of robots : Avoiding target congestion. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, pages 1955–1961, St. Louis, MO, USA, October 2009. 11
- [78] Ali Marjovi, João Goncalo Nunes, Lino Marques, and Aníbal de Almeida. Multi-robot exploration and fire searching. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, pages 1929–1934, St. Louis, MO, USA, October 2009. 31
- [79] Agostino Martinelli, Frederic Pont, and Roland Siegwart. Multi-robot localization using relative observations. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 2797–2802, Barcelona, Spain, April 2005. 7
- [80] Maja J. Matarić, Martin Nilsson, and Kristian T. Simsarian. Cooperative multi-robot box-pushing. In *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'95)*, pages 556–561, Pittsburgh, PA, USA, August 1995. 10
- [81] Nathan Michael, Michael M. Zavlanos, Vijay Kumar, and George J. Pappas. Distributed multi-robot task assignment and formation control. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA'08)*, pages 128–133, Pasadena, CA, USA, May 2008. 18
- [82] Javier Minguez and Luis Montano. Nearness diagram (ND) navigation : Collision avoidance in troublesome scenarios. *IEEE Transactions on Robotics and Automation*, 20(1) :45–49, 2004. 49, 72
- [83] Natsuki Miyata, Jun Ota, Tamio Arai, and Hajime Asama. Cooperative transport by multiple mobile robots in unknown static environments asso-

- ciated with real-time task assignment. *IEEE Transactions on Robotics and Automation*, 18(5) :769–780, 2002. 10
- [84] Mark Moors, Timo Röhling, and Dirk Schulz. A probabilistic approach to coordinated multi-robot indoor surveillance. In *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 3447–3452, Alberta, Canada, August 2005. 46
- [85] Edgar Nett and Stefan Schemmer. Reliable real-time communication in cooperative mobile applications. *IEEE Transactions on Computers*, 52(2) :166–180, February 2002. 13
- [86] Koichi Ogawara, Jun Takamatsu, Hiroshi Kimura, and Katsushi Ikeuchi. Estimation of essential interactions from multiple demonstrations. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA'03)*, pages 3893–3898, Taipei, Taiwan, September 2003. 38
- [87] Anibal Ollero, Joaquin Ferruz, Fernando Caballero, Sebastian Hurtado, and Luis Merino. Motion compensation and object detection for autonomous helicopter visual navigation in the COMETS system. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*, pages 19–24, New Orleans, LA, USA, April 2004. 2
- [88] Enrico Pagello, Antonio D'Angelo, Federico Montesello, Francesco Garella, and Carlo Ferrari. Cooperative behaviors in multi-robot systems through implicit communication. *Robotics and Autonomous Systems*, 29(1) :65–77, October 1999. 14
- [89] M. Pardowitz, R. Zollner, and R. Dillmann. Incremental acquisition of task knowledge applying heuristic relevance estimation. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA'06)*, pages 3011–3016, Orlando, FL, USA, May 2006. 38
- [90] Lynne E. Parker. ALLIANCE : An architecture for fault tolerant, cooperative control of heterogeneous mobile robots. In *Proceedings of the 1994 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94)*, pages 776–783, Munich, Germany, September 1994. 2, 7, 17, 25
- [91] Lynne E. Parker. *Heterogeneous Multi-Robot Cooperation*. PhD thesis, Massachusetts Institute of Technology, February 1994. 7, 19
- [92] Guilherme A. S. Pereira, Bruno S. Pimentel, Luiz Chaimowicz, and Mário F. M. Campos. Coordination of multiple mobile robots in an object carrying task using implicit communication. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 281–286, Washington, DC, USA, May 2002. 14

- [93] Bruno S. Pimentel and Mario F. M. Campos. Cooperative communication in ad hoc networked mobile robots. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'03)*, pages 2876–2881, Las Vegas, NV, USA, October 2003. 13
- [94] Robert Pindyck and Daniel Rubinfeld. *Microeconomics (8th Edition)*. Prentice Hall, 2012. 28
- [95] David Popenoe. *Sociology (11th Edition)*. Prentice Hall, 1999. 7
- [96] Marc Raibert, Kevin Blankespoor, Gabriel Nelson, Rob Playter, and the BigDog Team. BigDog, the rough-terrain quadruped robot. In *Proceedings of the 17th World Congress The International Federation of Automatic Control*, pages 10822–10825, Seoul, Korea, July 2008. 5
- [97] Jean-Louis Deneubourg Ralph Beckers, Owen E. Holland. From local actions to global tasks : Stigmergy and collective robotics. In *Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 181–189, Cambridge, MA, USA, July 1994. 1
- [98] Ioannis Rekleitis, Vincent Lee-Shue, Ai Peng New, and Howie Choset. Limited communication, multi-robot team based coverage. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*, pages 3462–3468, New Orleans, LA, USA, April 2004. 14
- [99] Ioannis M. Rekleitis, Gregory Dudek, and Evangelos E. Milios. Multi-robot collaboration for robust exploration. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA'00)*, pages 3164–3169, San Francisco, CA, USA, April 2000. 8
- [100] Martijn N. Rooker and Andreas Birk. Multi robot exploration under the constraints of wireless networking. *Control Engineering Practice*, 15(4) :435–445, 2007. 14
- [101] Stergios I. Roumeliotis and George A. Bekey. Distributed multirobot localization. *IEEE Transactions on Robotics and Automation*, 18(5) :781–795, 2002. 5, 7, 25
- [102] Stuart J. Russell and Peter Norvig. *Artificial Intelligence : A Modern Approach (2nd Edition)*. Prentice Hall, 2002. 7, 8, 15, 20, 21
- [103] Paul E. Rybski, Sascha A. Stoeter, Maria Gini, Dean F. Hougen, and Nikolaos P. Papanikolopoulos. Performance of a distributed robotic system using shared communications channels. *IEEE Transactions on Robotics and Automation*, 18(5) :713–727, 2002. 9
- [104] Mitul Saha and Pekka Isto. Multi-robot motion planning by incremental coordination. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'06)*, pages 5960–5963, Beijing, China, October 2006. 22, 47

- [105] Yoshiaki Sakagami, Ryujin Watanabe, Chiaki Aoyama, Shinichi Matsunaga, Nobuo Higaki, and Kikuo Fujimura. The intelligent ASIMO : system overview and integration. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, pages 2478–2483, Lausanne, Switzerland, September 2002. 5
- [106] Gildardo Sánchez and Jean-Claude Latombe. Using a PRM planner to compare centralized and decoupled planning for multi-robot systems. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 2112–2119, Washington, DC, USA, May 2002. 22
- [107] Mac Schwager, Daniela Rus, and Jean-Jacques Slotine. Unifying geometric, probabilistic, and potential field approaches to multi-robot deployment. *The International Journal of Robotics Research*, 30(3) :371–383, March 2011. 23
- [108] Weihua Sheng, Qingyan Yang, Song Ci, and Ning Xi. Multi-robot area exploration with limited-range communications. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04)*, pages 1414–1419, Sendai, Japan, September 2004. 8
- [109] Pedro M. Shiroma and Mario F. M. Campos. CoMutaR : A framework for multi-robot coordination and task allocation. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, pages 4817–4824, St. Louis, MO, USA, October 2009. 18
- [110] Reid G. Smith. The contract net protocol : High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, 29(12) :1104–1113, December 1980. 17
- [111] Guang Song and Nancy M. Amato. Randomized motion planning for car-like robots with C-PRM. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'01)*, pages 37–42, Maui, HI, USA, October 2001. 60
- [112] Anthony Stentz. Optimal and efficient path planning for partially-known environments. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation (ICRA'94)*, pages 3310–3317, San Diego, CA, USA, May 1994. 21, 30, 60
- [113] Anthony Stentz and M. Bernardine Dias. A free market architecture for coordinating multiple robots. Technical Report CMU-RI-TR-99-42, Carnegie Mellon University, Pittsburgh, PA, USA, December 1999. 17, 28
- [114] Peter Stone and Manuela Veloso. Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork. *Artificial Intelligence*, 110(2) :241–273, June 1999. 16

- [115] Peter Stone and Manuela Veloso. Multiagent systems : A survey from a machine learning perspective. *Autonomous Robots*, 8(3) :345–383, 2000. 3, 4
- [116] Freek Stulp, Michael Isik, and Michael Beetz. Implicit coordination in robotic teams using learned prediction models. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA'06)*, pages 1330–1335, Orlando, FL, USA, May 2006. 14
- [117] Petr Švestka and Mark H. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. In *Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA'95)*, pages 1631–1636, Nagoya, Japan, May 1995. 22, 46, 48, 59
- [118] Fang Tang and Lynne E. Parker. Asymtre : Automated synthesis of multi-robot task solutions through software reconfiguration. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 1501–1508, Barcelona, Spain, April 2005. 2, 16, 24, 38
- [119] Herbert G. Tanner and Amit Kumar. Towards decentralization of multi-robot navigation functions. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 4132–4137, Barcelona, Spain, April 2005. 21, 24
- [120] Sebastian Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1) :21–71, 1998. 22, 33, 45, 51
- [121] Eduardo Todt, Gustavo Raush, and Raúl Suárez. Analysis and classification of multiple robot coordination methods. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA'00)*, pages 3158–3163, San Francisco, CA, USA, April 2000. 12
- [122] Iwan Ulrich and Johann Borenstein. VFH+ : Reliable obstacle avoidance for fast mobile robots. In *Proceedings of the 1998 IEEE International Conference on Robotics and Automation (ICRA'98)*, pages 1572–1577, Leuven, Belgium, May 1998. 49, 68, 80
- [123] Richard T. Vaughan, Kasper Stoy, Gaurav S. Sukhatme, and Maja J. Mataric. LOST : Localization-space trails for robot teams. *IEEE Transactions on Robotics and Automation*, 18(5) :796–812, October 2002. 7, 8, 25, 38
- [124] Zhi Dong Wang, Yasuhisa Hirata, and Kazuhiro Kosuge. Control a rigid caging formation for cooperative object transportation by multiple mobile robots. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*, pages 1580–1585, New Orleans, LA, USA, April 2004. 10

- [125] Zhi Gang Wang, Meng Chu Zhou, and Nirwan Ansari. Ad-hoc robot wireless communication. In *Proceedings of the 2003 IEEE International Conference on Systems, Man and Cybernetics (SMC'03)*, pages 4045–4050, Washington, DC, USA, October 2003. 13
- [126] Jens Wawerla and Richard T. Vaughan. A fast and frugal method for team-task allocation in a multi-robot transportation system. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA'10)*, pages 1432–1437, Anchorage, AK, USA, May 2010. 5, 7, 8, 19, 41, 73
- [127] Barry Brian Werger and Maja J. Matarić. Broadcast of local eligibility for multi-target observation. In *Proceedings of the 5th International Symposium on Distributed Autonomous Robotic Systems (DARS'00)*, pages 347–356, Knoxville, TN, USA, October 2000. 17
- [128] Steven A. Wilmarth, Nancy M. Amatoy, and Peter F. Stiller. MAPRM : A probabilistic roadmap planner with sampling on the medial axis of the free space. In *Proceedings of the 1999 IEEE International Conference on Robotics and Automation (ICRA'99)*, pages 1024–1031, Detroit, MI, USA, May 1999. 54, 60
- [129] Ling Wu, Miguel Angel Garcia, Domenec Puig, and Albert Sole. Voronoi-based space partitioning for coordinated multi-robot exploration. *Journal of Physical Agents*, 1(1) :37–44, September 2007. 8
- [130] Kai M. Wurm, Cyrill Stachniss, and Wolfram Burgard. Coordinated multi-robot exploration using a segmentation of the environment. In *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08)*, pages 1160–1165, Nice, France, September 2008. 8, 22, 24, 31, 45
- [131] Seiji Yamada and Jun'ya Saito. Adaptive action selection without explicit communication for multi-robot box-pushing. *IEEE Transactions on Systems, Man, and Cybernetics, Part C : Applications and Reviews*, 31(3) :398–404, 2001. 14, 25
- [132] Brian Yamauchi. Frontier-based exploration using multiple robots. In *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*, pages 47–53, Minneapolis, MN, USA, May 1998. 30
- [133] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. Sampling-based multi-robot exploration. In *Proceedings of the Joint 41st International Symposium on Robotics and 6th German Conference on Robotics (ISR/ROBOTIK 2010)*, pages 44–49, Munich, Germany, June 2010. 8, 11, 20, 24, 25, 26, 33

- [134] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. Multi-robot decentralized exploration using a trade-based approach. In *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (ICINCO'11)*, pages 99–105, Noordwijkerhout, Netherlands, July 2011. 8, 19, 20, 25
- [135] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. ACS-PRM : Adaptive cross sampling based probabilistic roadmap for multi-robot motion planning. In *Proceedings of the 12th International Conference on Intelligent Autonomous Systems (IAS'12)*, pages 843–851, Jeju Island, Korea, June 2012. 8, 12, 24, 25, 26
- [136] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. Multi-robot heuristic goods transportation. In *Proceedings of the 6th IEEE International Conference on Intelligent Systems (IS'12)*, pages 409–414, Sofia, Bulgaria, September 2012. 8, 24, 25
- [137] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. On the problem of task planning in multi-robot systems. In *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (ICINCO'12)*, pages 423–426, Rome, Italy, July 2012. 25, 33
- [138] Zhi Yan, Nicolas Jouandeau, and Vincent Hugel. L3M-SIM team description. In *RoboCup 2011, 3D Soccer Simulation League*, Istanbul, Turkey, 2011. 8
- [139] Wei Ye, Richard T. Vaughan, Gaurav S. Sukhatme, John Heidemann, Deborah Estrin, and Maja J. Matarić. Evaluating control strategies for wireless-networked robots using an integrated robot and network simulation. In *Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA'01)*, pages 2941–2947, Seoul, Korea, May 2001. 9
- [140] Zoran Zivkovic, Bram Bakker, and Ben Kröse. Hierarchical map building and planning based on graph partitioning. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA'06)*, pages 803–809, Orlando, FL, USA, May 2006. 51
- [141] Robert Zlot and Anthony Stentz. Complex task allocation for multiple robots. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 1515–1522, Barcelona, Spain, April 2005. 8, 16
- [142] Robert Zlot, Anthony (Tony) Stentz, M. Bernardine Dias, and Scott Thayer. Multi-robot exploration controlled by a market economy. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 3016–2023, Washington, DC, USA, May 2002. 7, 8, 17, 25, 30

Titre : Contributions à la coordination de tâches et de mouvements pour un système multi-robots

Résumé : Nous étudions dans ce mémoire le problème de la coordination de tâches et de mouvements pour un système multi-robots du point de vue de la planification. Nos travaux se concrétisent par quatre nouvelles approches : une approche contrarienne centrée sur les échanges, une approche heuristique basée sur un modèle empirique, une approche par échantillonnage dédié à la décision et une approche par échantillonnage adaptatif. Toutes les approches proposées dans ce mémoire ont été testées et évaluées par des expériences de simulation.

Mots-clés : Système multi-robots, coordination, planification de tâches, planification de mouvements.

Title: Towards task and motion coordination of multi-robot systems

Abstract: This thesis studies the issue of multi-robot coordination from the standpoint of planning, which contains task planning and motion planning. Two new approaches are proposed for the task planning. The first one is a decentralized approach based on trading rules. This approach is designed to simulate the relationship between buyers and sellers in a business system, to achieve dynamic task allocation by using a mechanism of unsolicited bid. The second approach is an heuristic one which is based on an empirical model. It is designed to assign the transportation task to individual robots by estimating the production rate of goods in a centralized system. Two new approaches are proposed for the motion planning. Both are sampling-based approaches, designed to plan separate kinematic paths for multiple robots to minimize the issue of waiting situation including congestion, collision and deadlock, in an effective way in order to improve the system planning efficiency. All these proposed approaches have been tested and evaluated through simulation experiments.

Keywords: Multi-robot system, coordination, task planning, motion planning.