

RÉPUBLIQUE DU CAMEROUN

***** Paix

- Travail - Patrie

UNIVERSITÉ DE YAOUNDÉ I

FACULTÉ DES SCIENCES

DÉPARTEMENT D'INFORMATIQUE



REPUBLIC OF CAMEROON

Peace - Work - Fatherland

UNIVERSITY OF YAOUNDE I

FACULTY OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

**Protocole d'allocation des tâches dans une constellation de robots basée
sur la méthode hybride des enchères des enchères et K-Means**

En vue de l'obtention du diplôme de Master Recherche en Informatique

Option :

Science des Données

Présenté par :

PAMOE DISSANG Estébanc

Matricule :

16U2749

Superviseur :

Pr. ABESSOLO ALO'O Ghislain, Maître de Conférence

Année académique 2023 / 2024





Université de Yaoundé 1
Département d'Informatique

**Protocole d'allocation des tâches dans une constellation de robots basée
sur la méthode hybride des enchères des enchères et K-Means**

Présenté par :

PAMOE DISSANG Estébanc

Matricule :

16U2749

Superviseur :

Pr. ABESSOLO ALO'O Ghislain, Maître de Conférence

Année académique 2023 / 2024

DÉDICACES

Je dédie ce mémoire à ma chère famille.

REMERCIEMENTS

Tout d’abord, je remercie l’Éternel Dieu pour le soutien qu’il m’a toujours apporté et pour l’accomplissement de ce travail ; sans lui rien de ce ceci ne serait. Je témoigne ma gratitude et ma reconnaissance à toutes les personnes ayant contribué à la réalisation de ce travail :

Le Docteur ABESSOLO Ghislain, Chargé de cours à l’Université de Yaoundé 1, pour sa patience, ses conseils avisés et son soutien tout au long de ce projet de recherche. Ses retours constructifs ont grandement contribué à l’amélioration de mon travail.

Au directeur du laboratoire MSIO(Modélisation des Systèmes d’Information des Organisations), le Pr. ATSA Roger, pour le cadre de travail mis à disposition.

Au Professeur TSOPZE Norbert, Maître de conférence à l’Université de Yaoundé 1, pour ses précieux conseils qui m’ont permis d’appréhender les contours opérationnels de ce projet de recherche.

A mes parents et grands parents, Mr. DISSANG Jovani et Mme GUIADEM Agathe, Mr KOUAM Jean PhD. et Mme KOUAM Elisabeth, pour leur amour inconditionnel, leur soutien moral leurs sacrifices.

A mes frères, DISSANG Valdes, DISSANG Bolivard et DISSANG David, pour avoir meublé chaque étapes de ce parcours d’encouragement, de soutien et de sourire.

A ma fiancée et mon fils, EDI’I Jenny et DISSANG Paul-Henry, pour leur soutien inconditionnel, leurs encouragements et compréhension face à mes heures de travail acharné, vous êtes ma force et ma motivation.

A mes amis fidèles, YANHAMO Charnelle, KOUAMO Jules, MVUH Franck, EMATA Vichot, TCHASSEU Franck, FONGUE Audrey, EVINA Liliane et TCHOUALA Jiresse pour l’amour inconditionnel, le soutien et les moments de détente qui ont meublés cette aventure.

A mes chers complices, TSAGUE Arsène, DJILE Christophe KOUAM Charlain, KOUAM Rostand, KOUAM Arnaud, KENMOGNE Steve, PONE Junior, TCHOULI Frédéric, OROCK Ebot, TALLA Sonia et la famille GAPIN pour les conseils et la motivation.

RÉSUMÉ

Dans le contexte actuel, les interventions tactiques et opérationnelles face à des incidents tels que les catastrophes naturelles et les incendies sont souvent insuffisantes, en raison de facteurs anthropologiques (comme le danger pour la vie des équipes de sauvetage) et naturels (comme l'accessibilité des zones sinistrées). L'utilisation croissante de robots vise à assurer le sauvetage des personnes et des biens sans mettre en péril la vie des secouristes. Pour attribuer les tâches de sauvetage, les méthodes d'enchères sont employées afin de sélectionner le robot approprié pour une tâche. Nous avons amélioré le protocole d'enchères HaucTioN en y intégrant la notion de parallélisme pour éliminer le goulot d'étranglement dans la détermination des mises gagnantes. Le protocole amélioré, nommé HaucTioN-K en raison de l'utilisation de l'algorithme de clustering K-Means pour former K groupes de mises, s'est révélé supérieur à son prédécesseur tant au niveau de la complexité temporelle que des performances expérimentales.

Mots-clés : Système Multi-Agent, Allocation des Tâches, Algorithme des Enchères, Algorithme K-Means, Processus parallèles.

ABSTRACT

In today's context, tactical and operational responses to incidents such as natural disasters and fires are often inadequate due to anthropological factors (such as the risk to the lives of rescue teams) and natural factors (such as the accessibility of affected areas). The increasing use of robots aims to ensure the rescue of people and property in disaster zones without risking the lives of rescuers. Auction methods are used to assign rescue tasks, selecting the appropriate robot for each task. We have enhanced the HaucTioN protocol by integrating the concept of parallelism to eliminate bottlenecks in determining winning bids. The improved protocol, named HaucTioN-K due to the use of the K-Means clustering algorithm to form K groups of bids, proved superior to its predecessor in terms of temporal complexity and experimental performance.

Keywords : Multi-Agent System, Task Allocation, Auction Algorithm, K-Means Algorithm, Parallel processing.

TABLE DES MATIÈRES

TABLE DES FIGURES	vii
LISTE DES TABLEAUX	viii
LISTE DES ABREVIATION.....	ix
LISTE DES DEFINITIONS.....	x
CHAPITRE 1 : INTRODUCTION GENERALE.....	11
1.1 Contexte	11
1.2 Problématique.....	11
1.3 Méthodologie	12
1.4 Résultats.....	12
1.5 Organisation du mémoire.....	12
CHAPITRE 2 : REVUE DE LITTÉRATURE SUR LES APPROCHES D'ALLOCATION DES TÂCHES	13
2.1 Introduction.....	13
2.2 Généralités.....	13
2.2.1 Les éléments constitutifs d'un système MRTA	13
2.2.2 Taxonomie des systèmes MRTA.....	14
2.3 Etat de l'art	15
2.3.1 Les approches centralisées.....	15
2.3.2 Les approches décentralisées	16
2.3.3 Les approches distribuées	18
2.4 Comparaison des approches	23
2.4.1 Comparaison des performances	23
2.4.2 Tableau récapitulatif	23
2.5 Conclusion	24
CHAPITRE 3 : OPTIMISATION DE L'ALGORITHME HAUTION POUR L'ALLOCATION DES TÂCHES DANS UN SYSTÈME MULTI-ROBOT.....	25
3.1 Introduction	25
3.2 Schéma récapitulatif des étapes du protocole HaucTioN	25
3.2.1 Limites du protocole HaucTioN	26
3.3 Présentation du protocole HaucTioN-K	27
3.3.1 Principe du protocole HaucTioN-K.....	27
3.3.2 Fonctionnement du protocole HaucTioN-K.....	27
3.3.3 Schéma général du protocole HaucTioN-K	29
3.3.4 Validation du protocole HaucTioN-K.....	30
3.3.5 Avantages et limitations du protocole HaucTioN-K.....	32
3.4 Conclusion.....	33
CHAPITRE 4 : EXPERIMENTATION DU PROTOCOLE HAUTION-K.....	34

4.1	Introduction	34
4.2	Description de l'environnement d'expérimentation	34
4.2.1	Architecture technique	34
4.2.2	Architecture de simulation	34
4.2.3	Les données de simulation	35
4.2.4	Les métriques d'évaluation	35
4.2.5	Notion de fonction objectif	36
4.3	Utilisation du protocole HaucTioN	37
4.4	Utilisation du protocole HaucTioN-K	38
4.5	Validation expérimentale du protocole HaucTioN-K	39
4.5.1	Analyse des métriques: utilité, récompense et coût	39
4.5.2	Interprétation des variations de métriques	39
4.6	Conclusion	40
CHAPITRE 5 : CONCLUSION GENERALE ET PERSPECTIVES.....		41
5.1	Conclusion générale.....	41
5.2	Perspectives	42
BIBLIOGRAPHIE.....		43
ANNEXE A : EXPLICATIONS SUR K-MEANS		i
A.	Concepts généraux du Machine Learning	i
B.	L'algorithme K-Means	i
C.	Limites et complexité de K-Means	ii

TABLE DES FIGURES

Figure 1 : Automate à états finis décrivant le fonctionnement d'un robot[10].....	14
Figure 2 : Représentation visuelle des trois axes de la taxonomie proposée par Gerkey et Mataric[4]	14
Figure 3 : Illustration des différents types de réseaux[2].....	15
Figure 4 : Architecture du modèle proposé d'allocation des tâches avec un superviseur humain[8].....	17
Figure 5 : Vue d'ensemble du protocole HaucTioN.....	20
Figure 6 : Processus d'estimation de mise.....	21
Figure 7 : Processus de détermination des gagnants.....	22
Figure 8 : Comparaison des approches génétiques, de recuit simulé et d'enchères[9].....	23
Figure 9 : Schéma logique du protocole HaucTioN.	26
Figure 10 : Schéma récapitulatif du protocole HaucTioN-K.....	30
Figure 11 : Schéma de l'architecture de simulation.....	35
Figure 12 : Utilisation du protocole HaucTioN pour {N=10 et M=10}	37
Figure 13 : Utilisation du protocole HaucTioN pour {N=10 et M=50}	37
Figure 14 : Utilisation du protocole HaucTioN pour {N=50 et M=10}	37
Figure 15 : Utilisation du protocole HaucTioN-K pour {N=10 et M=10}	38
Figure 16 : Utilisation du protocole HaucTioN-K pour {N=10 et M=50}	38
Figure 17 : Utilisation du protocole HaucTioN-K pour {N=50 et M=10}.....	38
Figure 18 : Graphique de comparaison des protocoles HaucTioN et HaucTioN-K	39
Figure 19 : Illustration du fonctionnement de l'algorithme K-Means[12]	ii

LISTE DES TABLEAUX

Tableau 1 : Tableau récapitulatif des approches étudiées.....	24
Tableau 2 : Spécifications des étapes du protocole HaucTioN-K.	31
Tableau 3 : Tableau de comparaison des métriques utilisées	39

LISTE DES ABREVIATION

CEPMMT	Centre Européen de Prévision Météorologique à Moyen Terme
HaucTioN	Hierarchical Auction with HTN planning
HaucTioN-K	Hierarchical Auction with HTN planning on K branches
HTN	Hierarchical Task Network
MAS	Multi-Agent System
MILP	Mixed-Integer Linear Programming
MRS	Multi-Robot System
MRTA	Multi-Robot Task Allocation System
SA-reCBS	Simulated Annealing and recurrent Conflict Based Search
V-REP	Virtual Robot Experimentation Platform
WDP	Winner Determination Problem

LISTE DES DEFINITIONS

Automate fini	Modèle de représentation du comportement d'un robot en définissant des états discrets et des transitions entre ces états en réponse à des événements ou des conditions
Commissaire-priseur	Robot qui soumet un article à la vente lors d'un tour de vente aux enchères
Enchérisseur	Robot qui propose des mises dans l'optique d'acheter un article en vente
Goulot d'étranglement	Point d'un système limitant les performances globales, et pouvant avoir un effet sur les temps de traitement et de réponse
Items	Articles mis en vente dans une mise aux enchères
MatLab	Environnement de programmation utilisé pour la modélisation, la simulation et le contrôle de systèmes robotiques
Protocole	Ensemble de règles qui régissent les échanges de données ou le comportement collectif de processus en réseaux ou d'objets connectés
Robot	Machine programmable capable d'exécuter une série de tâches automatiquement ou semi-automatiquement
Solveur Gurobi	Solveur d'optimisation mathématique utilisé pour optimiser les trajectoires, la planification des tâches et l'allocation des ressources
V-REP	Virtual Robot Experimentation Platform est une plateforme de simulation robotique qui permet de modéliser, de programmer et de simuler des robots et des environnements complexes en 3D
Vente aux enchères	Vente ouverte au public, à la fin de laquelle le bien proposé est adjudgé au plus offrant.

CHAPITRE 1 : INTRODUCTION GENERALE

1.1 Contexte

Les chiffres du CEPMMT indiquent que le nombre de catastrophes naturelles a connu une forte augmentation cette dernière décennie[11]. Que ce soit les crues soudaines qui surviennent à chaque saison de pluie dans la ville de Douala, l'incendie qui a ravagé la Mairie de Yaoundé 4 le 8 septembre 2023 ou encore le puissant séisme de magnitude 7.4 qui a frappé Taïwan le 3 avril 2024, chacune de ces catastrophes représente une mission urgente et délicate pour les équipes de sauvetage et les victimes.

Des études ont montré que les 72 premières heures sont cruciales pour l'équipe d'intervention d'urgence afin de sauver des vies et mettre les populations touchées à l'abri du danger[2]. Cependant, dans la plupart des cas, l'envoi immédiat d'une équipe d'intervention humaine est soit trop dangereux, soit pas pratique[3]. Pour résoudre ce dilemme entre sauver la vie des populations et ne pas mettre en danger la vie des équipes de sauvetage, des robots sont utilisés comme premiers intervenants pour inspecter et identifier les points chauds clés après qu'une catastrophe ait frappé un endroit[11]. Ces robots peuvent être des drones, des robots sauveurs, des robots aquatiques ou encore des robots aérospatiaux[3].

La coopération optimale des robots disponibles lors d'un incident joue un rôle crucial dans les interventions d'urgence et la prise de décisions[11]. Les robots permettent aux équipes de sauvetage de collecter des données précises en temps réel sur les zones touchées grâce à leurs capteurs, ce qui leur permet de prendre des décisions éclairées et de sauver des vies sans nécessairement risquer celle des équipes de secours[2 ; 3]. Aussi, ces robots peuvent aller plus loin lors d'un incident ; Ils peuvent notamment identifier les foyers chauds, évaluer les moyens nécessaires afin d'optimiser l'utilisation des ressources, communiquer et coordonner les interventions, veiller à la sécurité des secouristes et même pulvériser de l'eau dans les zones inaccessibles par ceux-ci.

1.2 Problématique

Le sauvetage des victimes d'une catastrophe par une équipe de robots nécessite une parfaite collaboration entre eux afin de réaliser la mission d'une part, mais aussi de minimiser les ressources consommées tout en maximisant l'utilité de chaque robot[1 ; 5].

Nous pouvons dès lors nous poser la question de savoir: comment coordonner la réalisation des tâches d'une mission par une équipe de robots? Autrement dit, comment déterminer lequel des robots exécute laquelle des tâches afin d'atteindre l'objectif global de la mission?

Dans la littérature, le problème d'allocation de tâches à une équipe de robots est connu sous le nom de Multi-Robot Task Allocation System (MRTA)[6]. Pour résoudre ce problème, les chercheurs G. Korsah et al.[6] proposent une nouvelle taxonomie des systèmes MRTA appelée *iTax*, qui prend explicitement en considération les notions d'interdépendance entre les tâches, d'utilité des robots et les contraintes d'intervention. En se basant sur cette taxonomie (*iTax*), les chercheurs Yifan Bai et al.[13] proposent une approche qui permet à un robot maître d'allouer les tâches à des robots esclaves en combinant les algorithmes d'optimisation combinatoires de recuit simulé et de recherche basée sur le conflit. Malheureusement, ce modèle devient rapidement inefficace en cas de panne de communication entre le robot central qui coordonne l'allocation des tâches et les robots esclaves qui les exécutent.

Les chercheurs A. Agrawal et al.[1] proposent quant à eux une approche décentralisée d'allocation des tâches à une équipe de robots basée sur l'apprentissage par renforcement et la navigation décentralisée. La principale limite de cette approche réside dans les collisions observées entre les robots dans la pratique, collisions dues aux contraintes issues aux précédentes décisions d'allocation.

Non loin de cette architecture totalement décentralisée, Martina Lippi et Alessandro Marino[8] proposent une allocation des tâches à des robots coordonnés par des humains; l'idée étant de placer des clusters de robots autonomes sous la coordination d'un agent ; l'approche d'allocation des tâches au sein

d'un cluster étant basée sur la programmation linéaire par nombre entier. Malheureusement, comme le signalent les auteurs, cette approche est difficile à mettre en œuvre dans un cas réel d'incident, à cause de l'incapacité des agents à rester neutres et impartiaux dans leur mission de coordination des clusters de robots.

Pour venir à bout des limites suscitées, les chercheurs Antoine Millot et al.[2] proposent une approche d'allocation des tâches combinant la méthode des enchères et la planification hiérarchique. Toutefois, ils n'arrivent à prouver la performance de leur modèle qu'en faisant abstraction de nombreuses circonstances réelles d'intervention de robots comme l'hétérogénéité des objectifs de la mission ou encore l'hétérogénéité des robots.

1.3 Méthodologie

Pour répondre à la question principale de ce mémoire, nous allons utiliser un modèle hybride constitué de la méthode des enchères et de l'algorithme K-Means.

La méthode des enchères a la particularité de distribuer totalement le pouvoir de décision aux robots du système et de gagner en performance tout en restant robuste face aux éventuelles ruptures de signal de communication entre les robots. L'algorithme K-Means (voire annexe A) quant à lui rapproche notre modèle de la réalité, en simulant l'utilisation de robots hétérogènes dans une mission de sauvetage, avec notamment la création des clusters de robots basés sur leurs capacités.

1.4 Résultats

Après avoir déroulé notre méthodologie, nous obtenons comme résultat un Modèle d'allocation de tâches dans un système multi-robot; Ce modèle permet, une fois les robots déployés sur la zone d'incident, d'assurer en temps réel les missions de détection de points chauds, de planification des ressources nécessaires pour en venir à bout de la catastrophe, de diffusion des signaux sonores et lumineux pour guider les hommes et animaux éventuellement piégés par l'incident, de communication avec les équipes de secours telle que les urgences et la police, voire même de diffuser un message d'alerte spécifique aux rescapés.

1.5 Organisation du mémoire

Outre ce chapitre introductif, ce mémoire est organisé ainsi qu'il suit: Au chapitre 2, nous allons présenter une revue de la littérature sur les MRTA. Le chapitre 3 quant à lui va présenter le processus d'élaboration de notre protocole. Au chapitre 4, nous allons faire une expérimentation du protocole proposé avant de conclure et donner quelques perspectives de recherche au chapitre 5.

CHAPITRE 2 : REVUE DE LITTÉRATURE SUR LES APPROCHES D'ALLOCATION DES TÂCHES

2.1 Introduction

L'allocation des tâches à un système Multi-Agent (ou Multi-Agent System, MAS) fait l'objet des travaux de recherche dans les domaines de l'électronique, des mathématiques, de l'informatique fondamentale, de l'intelligence artificielle et même des sciences sociales. L'agent ici peut être soit un robot, un humain ou un programme informatique. Dans ce chapitre, nous allons présenter les travaux des différents auteurs qui se sont intéressés à l'allocation des tâches dans un système multi-agent.

2.2 Généralités

L'allocation des tâches à un système multi-agent, connu sous le nom du problème MRTA[4] consiste à déterminer quels agents doivent exécuter les tâches disponibles afin d'atteindre les objectifs globaux du système. Cet ensemble d'agents, qui peut être constitué d'humains, de robots, ou encore des deux, porte le nom de système multi-robot ou encore système multi-agent[4]

Dans cette section, nous présentons les éléments constitutifs d'un système MRTA, puis la taxonomie d'un MRTA

2.2.1 Les éléments constitutifs d'un système MRTA

Les travaux effectués dans le domaine de la robotique ont permis de relever les principaux éléments qui influencent la conception et la mise en œuvre d'un système MRTA au rang desquels:

- Les robots individuels et les tâches à allouer ;
- Les objectifs de la mission, ses contraintes et les critères d'évaluation des résultats.

a) Les robots individuels et l'allocation des tâches

Dans un MAS, chaque robot est une entité autonome capable de perception, de mouvement et d'interaction avec l'environnement. Ils peuvent être de différents types (aériens, terrestres, sous-marins) et avoir des capacités variées. Grâce à des algorithmes d'allocation, ils doivent décider quelles tâches accomplir en fonction de leurs capacités et des priorités. Les auteurs Shinkyu Park et al.[10] modélisent un robot comme un automate à états finis (voire figure 1) afin de représenter comment chacun d'eux évolue dans son processus interne de décision.

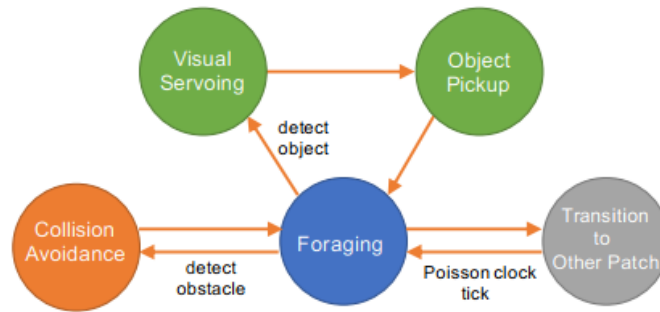


Figure 1 : Automate à états finis décrivant le fonctionnement d'un robot[10]

b) Notion d'objectifs de la mission

Indépendamment de sa nature, une mission peut, dans la plupart des cas, se résumer à trois éléments principaux[11]:

- Un but, composé d'objectifs à accomplir ;
- Des contraintes de mission: qui sont des impératifs sur la façon d'accomplir le but ;
- Des critères d'évaluation: qui fournissent quant à eux d'apprécier le niveau d'accomplissement des deux premiers éléments

c) Les principaux défis du problème MRTA

Les défis directs du problèmes MRTA sont à la fois ceux propres à la réalisation d'une tâche (comme la quantité d'énergie nécessaire à déplacer un massif) et ceux dus aux éventuelles dépendances entre les tâches; les défis indirects quant à eux sont attribués aux circonstances de communication et de coordination des composantes du système[11].

2.2.2 Taxonomie des systèmes MRTA

Biran P.Gerkey et al.[4] proposent une grille de classification en huit blocs du système MRTA (voire figure 1) qui permet de modéliser la façon de coordonner les robots(robot task), de sélectionner les tâches (task type) et de faire évoluer le système (allocation type). La figure 2 ci-dessous représente la grille de classification de Biran P.Gerkey et al.[4] dans un repère orthonormé de dimension 3

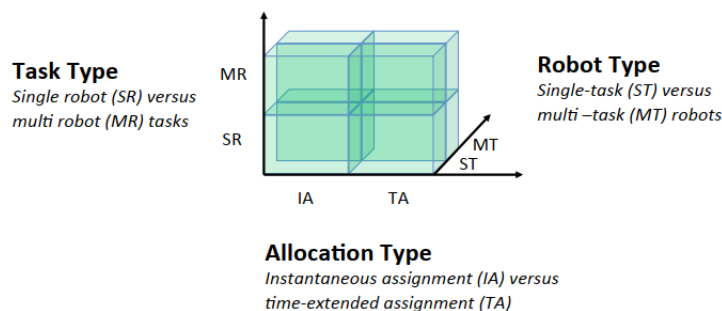


Figure 2 : Représentation visuelle des trois axes de la taxonomie proposée par Gerkey et Mataric[4]

a) Coordination d'un robot

Ici, on s'intéresse au nombre de tâches qu'un robot peut réaliser simultanément:

- On parlera de Multi-Task system (MT) lorsqu'un robot peut effectuer plus d'une tâche simultanément;
- On parlera de Single-Task system (ST) lorsqu'un robot ne peut effectuer qu'au plus une tâche à la fois.

b) Sélection des tâches

Suivant la nature des objectifs d'un problème MRTA, certaines tâches peuvent nécessiter plus d'un robot simultanément pour être bouclées :

- On parle ainsi d'un Multi-Robot (MR) system pour un système où une tâche peut nécessiter plus d'un robot pour sa réalisation;
- On parle en revanche de Single-Robot (SR) system lorsqu'une tâche nécessite au plus un robot pour sa réalisation.

c) Séquencement des opérations

La décision d'allocation de tâches au système multi-robot peut dépendre uniquement de l'état précédent du système ou bien de la succession des états précédents. On distingue ainsi:

- L'instantaneous assignment (IA) : ici, Toutes les informations sur les robots, les tâches, et l'environnement ne permettent qu'une allocation sans composante temporelle future;
- Le time-extended assignment (TA) : ici par contre, le système possède une mémoire facilitant l'allocation des tâches sur une période donnée.

2.3 Etat de l'art

Un élément essentiel dans la résolution du problème MRTA est la structure des processus de décision, de supervision et de répartition ainsi que les interactions[11]. Cet agencement définit ainsi l'architecture du système, que nous pouvons regrouper en trois familles d'approches: centralisées, décentralisées et distribuées. La figure 3 ci-dessous illustre les trois formes d'architectures étudiées sous forme de graphe.

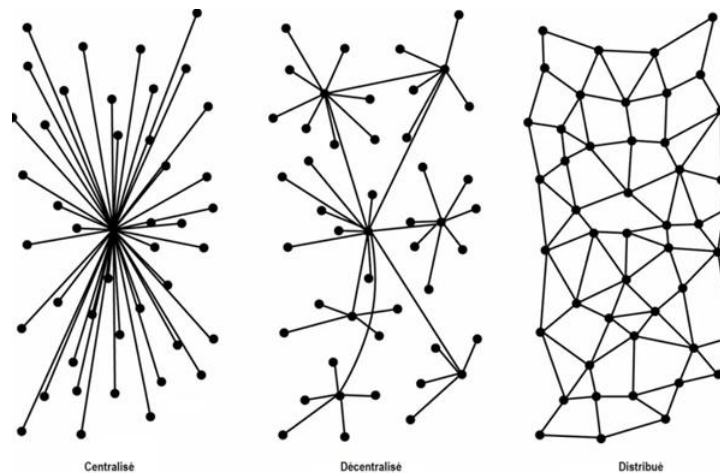


Figure 3 : Illustration des différents types de réseaux[2]

2.3.1 Les approches centralisées

L'approche centralisée est généralement la plus utilisée, car elle est simple à mettre en œuvre. En principe, tous les agents s'organisent autour d'un seul agent qui possède un algorithme d'allocation. Le rôle de cet agent central consiste à:

- Récupérer les informations de tous les agents ;
- Planifier l'allocation des tâches de tous les agents ;
- Transférer le plan d'allocation aux agents ;

Cette famille d'approches a le principal avantage d'être simple à mettre en œuvre, mais reste très

vulnérable du fait de sa dépendance à l'agent central car toute rupture de communication rend impossible la suite de la mission. De plus, l'augmentation d'un grand nombre de tâches ou de robots (passage à l'échelle) fait exploser le coût calculatoire et peut rendre le système inefficace.

Une grande diversité d'algorithmes existe dans la littérature pour résoudre le problème MRTA de façon centralisée. Dans la plupart des cas, les travaux implémentent cette approche en la modélisant comme un problème d'optimisation ou de programmation sous contraintes.

Parmi ces approches figurent en bonne partie les travaux des auteurs Yifan Bai et al.[13], qui proposent une résolution du problème MRTA par l'utilisation d'une métaheuristique: le recuit simulé.

a) Les abstractions faites par Yifan Bai et al.

Afin de proposer une approche de résolution du problème MRTA par combinaison du recuit simulé et de la recherche récurrente basée sur le conflit, les auteurs posent quelques restrictions(R) applicatives au problème:

- R1: Tous les robots esclaves ont les positions de départ et d'arrivée connues du robot coordinateur ;
- R2: Les itinéraires de navigations du système ne doivent présenter aucun conflit; deux robots ont des itinéraires de navigation en conflit dans deux cas:
 - Cas 1 - conflit de chemin: lorsque deux robots allant dans les directions opposées se trouvent sur le même chemin au même instant t ;
 - Cas 2 - conflit de destination: lorsque deux robots se trouvent au même point au même instant.

b) Solution proposée par Yifan Bai et al. : le recuit simulé

Yifan Bai et al.[13] proposent un algorithme à deux couches nommé SA-reCBS (pour Simulated Annealing - recurrent Conflict Based Search) qui sera hébergé dans un robot central, responsable du pilotage des équipes de robots:

- La première couche, Le recuit simulé: qui permet d'assigner une tâche aux robots ;
- La deuxième couche, la recherche basée sur le conflit: qui est responsable d'établir un itinéraire de navigation sans conflit.

c) Résultats obtenus et limites observées

Les travaux de Yifan Bai et al.[13] permettent d'obtenir un modèle d'allocation de tâches à deux couches qui a la particularité d'économiser les ressources consommées par les robots tout en réduisant le temps de calcul du système (durée totale nécessaire au robot central pour établir un plan d'allocation des tâches). Toutefois, ce modèle reste idéaliste puisqu'il assume que l'environnement et le nombre de robots restent constants, alors que dans la réalité, une panne de robots par exemple modifie directement le nombre de robots disponibles, de même que l'environnement où évoluent les robots change dynamiquement (température et relief du terrain).

2.3.2 Les approches décentralisées

L'approche décentralisée permet de répartir la décision d'allocation au sein du système multi-robot et donne ainsi plus de liberté aux agents qui s'organisent localement. Les agents ne dépendent plus d'un seul agent central responsable du processus de décision, mais plutôt aux agents locaux.

Le principal avantage de cette famille d'approches est la robustesse face aux ruptures de communication entre les robots et le système central. De plus, elles offrent théoriquement des solutions meilleures que celles des familles centralisées puisque le plan d'allocation de chaque agent tient compte directement de l'état de l'environnement dans lequel il évolue. Toutefois, cette famille d'approches ont le principal inconvénient d'être plus complexes à mettre en œuvre et à étudier que la famille d'approches centralisées.

L'idée fondamentale de cette famille d'approches réside dans l'emploi d'un ensemble hétérogène d'agents. Nous pouvons par exemple considérer le cas d'une équipe d'intervention constituée de robots sauveurs répartis en un nombre fini de clusters pilotés par des secouristes de façon à ce que chaque secouriste puisse ajuster son fonctionnement aux objectifs de mission du cluster de robots auquel il appartient. Cette approche est bien illustrée par Martina Lippi et Alessandro Marino[8], qui proposent une méthode optimale de résolution du problème MRTA par la programmation linéaire par nombre entier (MILP - Mixed-Integer linear programming).

a) Les abstractions faites par Martina Lippi et Alessandro Marino : MILP

Afin de résoudre le problème MRTA comme un problème MILP, ils émettent un certain nombre de suppositions(S) sur les tâches à réaliser, dont:

- S1: Chaque tâche a un seuil de résolution à partir duquel elle est marquée comme complète, et peut être supervisée par un opérateur humain pour garantir sa résolution ;
- S2: Certaines tâches peuvent nécessiter la collaboration entre les robots, comme s'il s'agissait par exemple pour une équipe de robots de soulever une charge lourde ;
- S3: Certaines tâches ne sont attribuables qu'aux agents humains, comme les tâches de supervisions ou encore celles de transport d'un objet difforme ;
- S4: Certaines tâches ne sont attribuables qu'aux robots uniquement, comme les tâches de manipulation de substances dangereuses à la santé d'un agent humain ;

Ces suppositions permettent de circonscrire le périmètre du problème MRTA, débouchant ainsi sur sa variante ST-MR-IA.

b) Solution proposée : MILP

La proposition de Yifan Bai et al.[13] est éprouvée dans un environnement de simulation, dont l'architecture (voire figure 4) est constituée d'un opérateur humain local, d'un solveur Gurobi et d'une interface réceptacle visuel V-REP. L'opérateur humain envoie des commandes via un contrôleur interfacé avec Matlab. Ce dernier a recours au solveur Gurobi pour trouver une solution au problème d'optimisation et communique avec l'environnement de simulation V-REP. Le retour visuel est fourni à l'humain par V-REP.

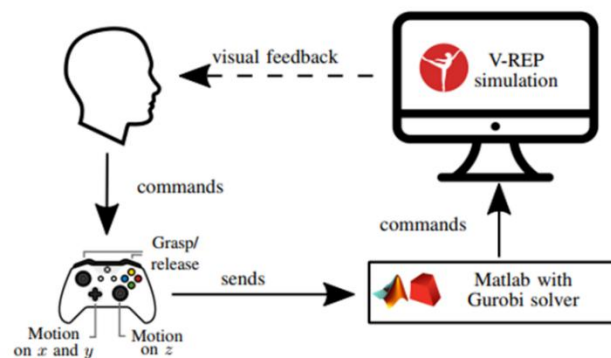


Figure 4 : Architecture du modèle proposé d'allocation des tâches avec un superviseur humain[8]

c) Résultats obtenus et limites observées

Les modèles d'allocation basés sur la collaboration entre les humains superviseurs et les robots exécutants offrent l'atout de résoudre les conflits dus aux nouveaux objectifs arrivant dans le cas des systèmes MRTA temps réel, puisque l'humain intervient sur le système en modifiant ses paramètres en fonction des nouveaux objectifs de la mission. Toutefois, la solution proposée au problème d'allocation des tâches par l'utilisation

du MILP et d'un superviseur humain telle que proposée a pour limitation de nécessiter la présence d'un intervenant humain afin d'assurer la mobilisation des robots dans le système.

2.3.3 Les approches distribuées

Dans cette famille d'approches, chaque agent coopère avec ses voisins afin d'améliorer son plan d'allocation, leur permettant ainsi de se concentrer davantage sur le problème local. Le principal atout de cette famille d'approches est la mise en échelle facile tandis qu'il présente la faiblesse majeure de la difficulté de coordination globale des agents autour d'un objectif complexe (comme les missions qui nécessitent une synchronisation dans la chronologie de la réalisation des tâches).

Pour résoudre le problème MRTA dans le cadre d'une mission de détection et de désamorçage des mines sous-marines, les chercheurs Antoine Milot et al.[2] proposent un modèle d'allocation par enchères permettant aux robots de communiquer efficacement dans un réseau maillé tout en conservant une autonomie de décision sur la mission qui leur est assignée.

a) Les abstractions faites par Antoine Milot et al.

En plus des capacités intrinsèques d'un robot comme la détection des objets et la mobilité dans l'espace, Antoine Milot et al.[2] fixent deux prérequis majeurs pour la résolution du problème MRTA:

- Chaque robot du système doit posséder les capacités par défaut requises pour résoudre chacune des tâches disponibles ;
- Chaque robot est suffisamment autonome pour déterminer comment effectuer les tâches disponibles.

b) Solution proposée : HaucTioN

Encore appelées "Market-Based approaches", les méthodes d'allocation par enchères doivent leur nom aux principes économiques dont ils s'inspirent : la vente aux enchères. Cette approche permet l'élaboration et l'exécution d'un contrat de vente entre les robots du système multi-robot. L'obtention de ce contrat fait face à des enchères. Dans le cas d'un système MRTA, le contrat peut être vu comme le plan d'exécution des tâches par un robot.

i) Les caractéristiques du protocole HaucTioN

Les approches par enchères sont constituées de nombreux algorithmes qui diffèrent les uns des autres selon les paramètres du problème à résoudre. Parmi ces paramètres, nous pouvons citer:

- La nature de l'item mis en vente. Par exemple, une tâche élémentaire, un ensemble de tâches, ou encore une tâche abstraite décomposable ;
- Le nombre d'items mis en vente. On peut ainsi mettre en vente un item ou plusieurs à la fois. Dans le cas où plusieurs items sont en vente, il est nécessaire de définir des règles spécifiques sur le nombre d'items que peut remporter un enchérisseur ;
- La nature centralisée ou décentralisée des enchères, c.-à-d. qui peut tenir le rôle de commissaire-priseur ;
- L'utilisation d'un délais. Par exemple, pour accorder un temps maximum aux enchérisseurs pour placer leurs mises; ce délais impacte le choix des algorithmes d'estimation des mises pour les enchérisseurs. C'est un point très important quand les tâches à estimer sont en ligne car les enchérisseurs doivent être aptes à soumettre une enchère tout en continuant l'exécution de leurs tâches en cours ;
- La qualité des communications considérées. En effet, tous les robots ne pourront pas participer aux enchères du fait de la portée des capteurs des robots et la qualité du signal de communication ;
- La disparité des connaissances entre robots. Avec les contraintes de communication, une diversité des connaissances sur la mission et l'environnement peut apparaître en ligne. Les robots estimant leur mise en fonction de leurs connaissances du MAS. Il peut donc par exemple être nécessaire d'écarter les

robots moins bien informés .

La combinaison de ces caractéristiques donne lieu à des schémas d'enchères, qu'on peut répartir en quatre catégories:

- L'enchère au Single-Item (SI): lorsqu'une unique tâche est mise en vente pour une phase d'enchère
- L'enchère au Sequential Single-Item (SSI): lorsque dans la même enchère, les tâches sont mises en vente et allouées une par une à chaque tour d'enchères ;
- L'enchère au Parallel Single-Item (PSI): lorsque toutes les tâches sont vendues au même instant. Dans ce cas, les robots proposent une mise pour chacune d'entre elles, et l'enchère est résolue en un seul tour.
- L'enchère combinatorial: Lorsque toutes les tâches sont en vente au même instant, et les robots se réservent le droit de miser sur la tâche qui les intéresse à l'instant qui leur convient le mieux.

ii) Formalisme du protocole HaucTioN

Antoine Millot et al. décrivent l'approche par enchères comme suit :

- Une mission devant être accomplie par le système multi-robot. Cette mission est décomposable en sous éléments accomplissables par un ou plusieurs robots ;
- L'équipe a un nombre de ressources limitées ;
- La fonction objectif servant à quantifier les préférences à adopter sur toutes les solutions ;
- Une fonction individuelle d'utilité pour chaque robot ;
- Une correspondance entre la fonction objectif globale et les fonctions d'utilités individuelles ; Cette correspondance permet de définir comment les contributions de chacun font progresser l'objectif principal ;
- Un mécanisme d'enchère pour attribuer les ressources et tâches ;
- Chaque enchère fait l'objet d'un tour d'enchères constitué de quatre étapes essentielles:
 1. L'annonce: un commissaire-priseur ("auctioneer ") met en vente les tâches disponibles ;
 2. L'estimation des mises: les robots enchérisseurs ("bidders") analysent les tâches mises en vente et estiment les mises qu'ils peuvent faire dessus ;
 3. La détermination des gagnants: une fois que les robots ont analysé toutes les estimations ou qu'une condition d'arrêt est atteinte(délais de proposition des mises par exemple), ils communiquent sur les mises effectuées, et s'accordent deux à deux sur l'allocation des tâches à partir du critère d'évaluation des mises. Cette étape porte le nom de WDP (Winner Determination Problem) ;
 4. La récompenses: le commissaire-priseur clôt le tour d'enchères et notifie les vainqueurs.

iii) Fonctionnement d'un tour d'enchères HaucTioN

Le protocole HaucTioN se déroule en quatre étapes, partant de l'arrivée d'une nouvelle tâche dans le système(l'annonce), la collecte des utilités de robots pour l'exécution de la tâche (l'estimation des mises), la mise en avant des robots ayant les meilleures mises (détermination des gagnants), pour finir par la notification des robots vainqueurs (la transmission des récompenses et clôture du tour d'enchères).

La figure 5 ci-dessous résume le fonctionnement du tour d'enchères. Dans cette figure :

- les rectangles représentent des structures de données gérées par les robots ;
- Les flèches en pointillés représentent les données échangées entre les robots ;
- Les cercles indiquent les processus qui agrègent les informations pour construire de nouvelles structures ;
- Les doubles flèches indiquent les appels à un planificateur HTN ;
- Les flèches continues indiquent les interactions entre les structures de données et les processus.

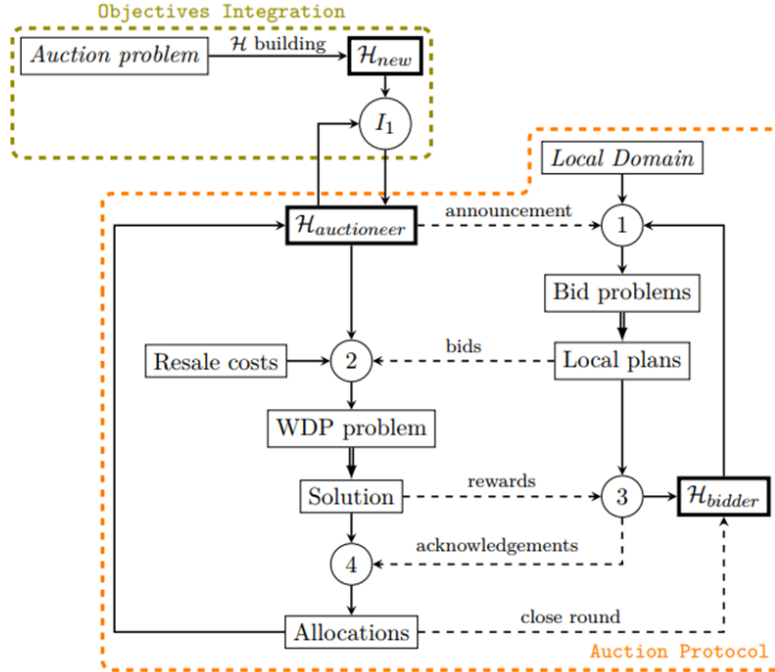


Figure 5 : Vue d'ensemble du protocole HaucTioN.

I. Etape 1: L'annonce

La toute première étape d'un tour d'enchères consiste à transmettre un message d'annonce contenant des informations sur les objets en vente. Elle se fait en deux étapes:

- La détection des tâches disponibles : au premier tour d'enchères, les tâches disponibles sont connues par les agents. Par la suite, il est question de monitorer les nouvelles tâches qui arrivent dans le système (comme l'intégration en temps réel d'un nouvel objectif), ou encore de constater qu'une tâche a été bouclée (c'est le cas d'un objectif atteint) ;
- La formalisation du message d'annonce : une fois les tâches détectées, elles sont modélisées dans le formalisme des MRTA et transmit aux enchérisseurs. Un message d'annonce est ainsi constitué de:
 - auctionnerID : l'identifiant du commissaire-priseur ;
 - auctionID : l'identifiant de l'enchère ;
 - roundID : l'identifiant du tour d'enchères ;
 - t_a , le délai imposé par le commissaire-priseur pour effectuer les mises ;
 - γ : l'article mis en vente, tel que $\gamma=(\mathcal{H}_\gamma, \lambda_\gamma)$, où $\mathcal{H}_\gamma$ est l'ensemble des robots de taille n , et λ_γ est l'ensemble des tâches disponibles .

II. Etape 2: Estimation des mises faites par les robots

Une fois qu'un message d'annonce est reçu, chaque robot doit calculer une mise pour chaque tâche réalisable. La valeur de la mise correspond au coût d'exécution de cette tâche.

Le processus d'estimation d'une mise prend en entrée l'article en vente $\gamma=(\mathcal{H}_\gamma, \lambda_\gamma)$, l'ensemble des labels obtenus à l'annonce Φ , le modèle générique des capacités locales du robot, l'état initial du système, noté Si_0 .

A partir des données d'entrée sont construites des structures intermédiaires:

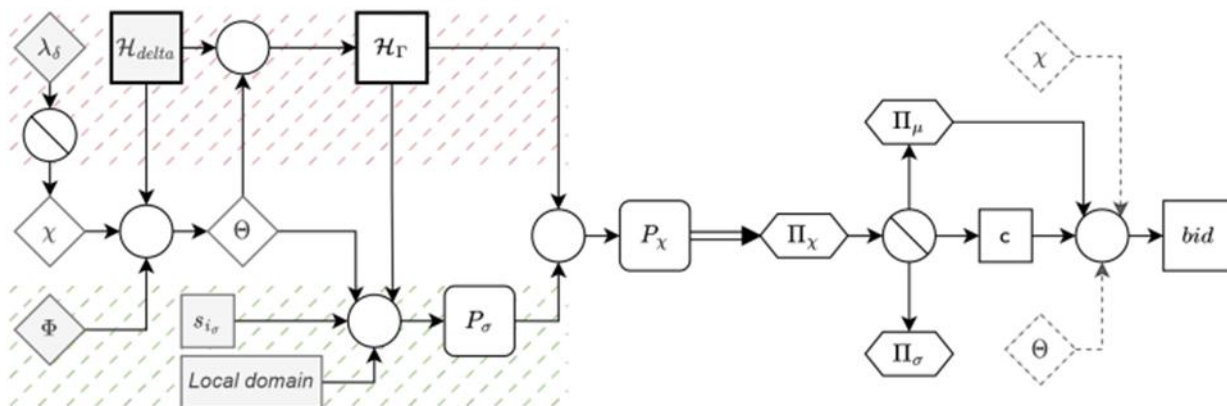
- L'ensemble des labels à estimer, noté Θ ;
- Le multi-robot HTN réduit du problème d'estimation, noté H_Γ ;
- Le problème local au formalisme HTN, noté P_σ ;
- Le problème d'estimation à résoudre, noté P_χ ;
- Le plan solution du problème d'estimation, noté Π_χ , ainsi que son coût et ses parties locales et multi-robots, notés respectivement C , Π_σ et Π_μ ;

La sortie du processus d'estimation des mises est la mise (encore appelée "bid"), et comporte les

- l'ID de l'enchérisseur ;
- L'identifiant de l'enchère auctionID ;
- L'identifiant du tour d'enchères roundID ;
- Le label de la tâche souhaitée X ;
- La valeur c de la mise (encore appelée utilité de l'enchérisseur) ;
- Le plan d'allocation associé à la mise.

La figure 6 ci-dessous illustre le processus d'estimation des mises. Dans cette figure :

- Les blocs dans les parties hachurées en rose et en vert représentent respectivement des éléments propres à l'article en vente et à l'enchérisseur ;
- Les rectangles sont des structures de données gérées par l'enchérisseur ;
- Les losanges représentent des ensembles de labels multi-robots ;
- Les blocs grisés sont des données d'entrée du problème d'estimation ;
- Les cercles pleins indiquent les processus qui agrègent les informations pour construire de nouvelles structures et les cercles barrés indiquent les processus qui déconstruisent une structure en plusieurs autres ;
- Les rectangles arrondis représentent des problèmes de planification hiérarchique ;
- La double flèche indique un appel à un planificateur HTN ;
- Les hexagones représentent des plans solutions d'un problème de planification ;
- Les éléments en pointillés allègent la figure en faisant référence à leurs blocs d'origine.



III. Etape 3: Détermination des gagnants

- Un robot peut remporter au plus une mise: Autrement dit, après avoir proposé sa mise par rapport aux tâches existantes, il ne lui sera attribué qu'au plus une tâche, assurant ainsi qu'un robot ne peut exécuter qu'une seule tâche(ST) ;
- La compatibilité entre les mises retenues: Autrement dit, pour être retenues, les mises de deux robots doivent être compatibles, assurant ainsi qu'une tâche peut nécessiter plus d'un robot(MR).

Le processus de détermination des gagnants (Figure 7) prend en entrée la liste des mises B et l'article en vente $\gamma=(H_\gamma, \lambda_\gamma)$, puis retourne le sous-ensemble $B_w \subset B$ des mises gagnantes. Mais comme à l'étape précédentes, les structures intermédiaires sont formées, notamment:

- L'ensemble des mises de chaque robot, noté B_{r_i} , pour chaque robot $r_i \in R$;
- Les problèmes HTN dédiés à l'encodage des mises des robots, notés P_{r_i} pour chaque robots $r_i \in R$;
- Le problème HTN dédié à l'encodage de la revente, noté P_{resell} . La revente concerne les mises n'ayant reçues aucune mise au tour d'enchères en cours ;
- Le problème de détermination des gagnants, noté P_{wdp} ;
- Le plan solution du problème de détermination des gagnants, noté Π_{wdp} .

La figure 7 illustre le processus de détermination des gagnants. Dans cette figure :

- Les rectangles sont des structures de données gérées par le robot ;
- Les blocs grisés sont des données d'entrée du problème de détermination des gagnants ;
- Les cercles pleins indiquent les processus qui agrègent les informations pour construire de nouvelles structures et les cercles barrés indiquent les processus qui déconstruisent une structure en plusieurs autres ;
- Les rectangles arrondis représentent des problèmes de planification hiérarchique ;
- La double flèche indique un appel à un planificateur HTN ;
- L'hexagone représente un plan solution d'un problème de planification ;
- Les éléments en pointillés représentent une répétition du processus d'encodage des mises n'importe quel robot r_i , avec $r_i \in R = (r_1, \dots, r_n)$.

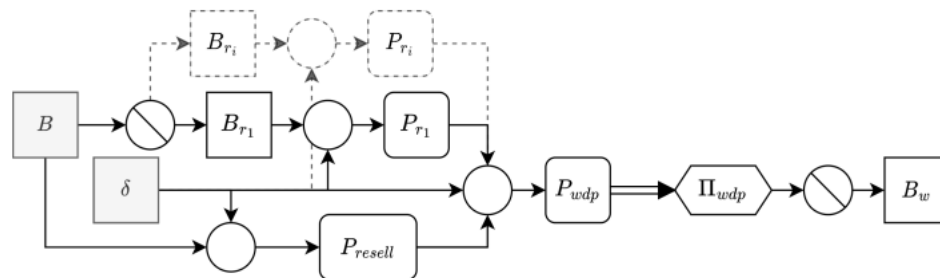


Figure 7 : Processus de détermination des gagnants.

IV. Etape 4: Distribution des récompenses et clôture du tour d'enchères

Afin de pallier aux éventuelles pannes de communication entre les robots, HaucTioN introduit la notion d'accusé de réception. En effet, tout robot gagnant doit envoyer un accusé de réception de sa récompense. Il doit ensuite intégrer les tâches qu'il s'est vu allouer à l'ensemble des tâches qu'il doit accomplir. La clôture de l'enchère est due à l'écoulement du temps t_a (durée du tour d'enchères).

c) Résultats obtenus et limites observées

Le modèle d'allocation de tâches à des robots totalement autonomes tel que proposé par les auteurs permet la réalisation de la mission de façon optimale en ressources consommées et permet en plus de répartir les plans de navigation des robots. Toutefois, le modèle est confronté à de nombreux défis dont le plus important est la non considération des éléments de contexte de la mission comme la dépendance entre les tâches ou encore l'absence de coopération orientée mission entre les robots.

2.4 Comparaison des approches

2.4.1 Comparaison des performances

Bien que les approches suscitées présentent des avantages et des limites dans la conception et l'opérationnalisation, les simulations effectuées par les chercheurs Mohamed Badreldin et al.[9] permettent de noter que les méthodes d'allocations de tâches distribuées (« market based » par exemple) sont meilleures (en terme d'utilité) que celles décentralisées ou encore centralisées, que nous soyons devant un système réduit ou celui possédant un grand nombre de tâches (voire figure 8).

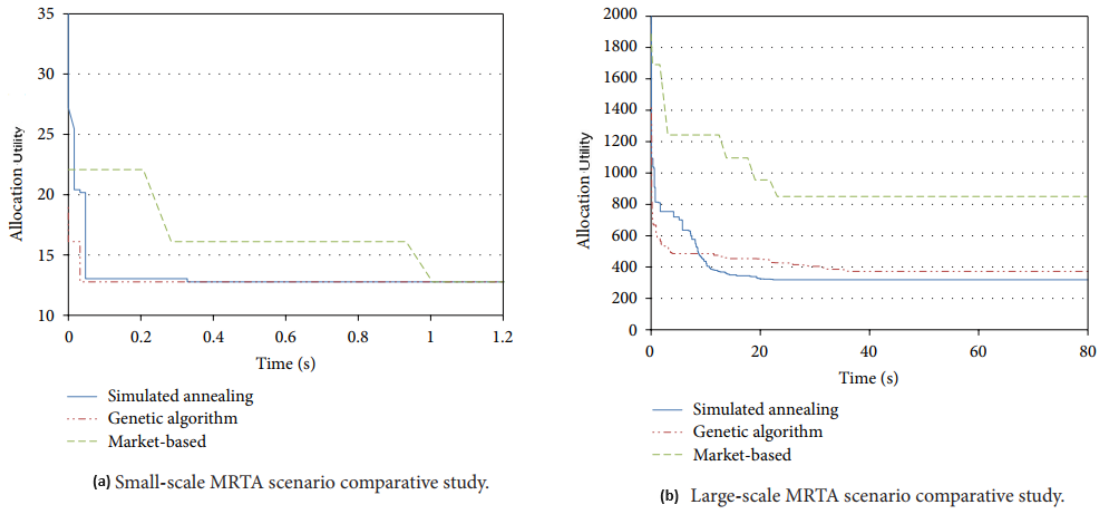


Figure 8 : Comparaison des approches génétiques, de recuit simulé et d'enchères[9].

2.4.2 Tableau récapitulatif

Le tableau ci-dessous regroupe les différentes approches étudiées pour la résolution du problème MRTA ainsi que les forces et faiblesses de quelques algorithmes étudiés dans ces familles.

	Méthodologie	Algorithmes étudiés	Topologie du réseau	Avantages	Inconvénients
Centralis	Un robot maître établit le plan d'allocation et le communique aux robots esclaves qui l'exécutent	SA-reCBS	Etoile	- Rapide à mettre en œuvre - Gain en temps de calcul global	- Difficulté de mise en échelle - Intolérance aux pannes de réseau
Décentralisées	Le système est divisé en clusters de robots. Chaque robot esclave a un maître local auquel il rend compte, et seul les maîtres communiquent entre eux	MILP	Etoile étendu	Bonne résistance aux problèmes de communication	Plus complexe à étudier et mettre en place

Distribuées	Chaque robot communique directement avec tout autre robot du réseau	HaucTioN	Maillée	Grande autonomie et tolérance aux fautes	Faible coopération et conflits de décisions
-------------	---	----------	---------	--	---

Tableau 1 : Tableau récapitulatif des approches étudiées

2.5 Conclusion

Dans ce chapitre, nous avons présenté les notions et quelques travaux récents effectués dans le cadre de l'allocation des tâches à un système multi-robot. Pour chacun des travaux présentés, nous avons relevé les limites associées, et grâce aux travaux de Mohamed Badreldin et al.[9], nous avons relevé que les approches distribuées étaient meilleures, du fait de l'indépendance des agents dans le système et de la résilience aux soucis de communication.

Dans la suite de nos travaux, nous allons nous focaliser sur l'algorithme distribué HaucTioN du professeur Antoine Millot pour adresser la problématique de coordination de la réalisation des tâches dans un système multi-robot.

CHAPITRE 3 : OPTIMISATION DE L'ALGORITHME HAUTION POUR L'ALLOCATION DES TÂCHES DANS UN SYSTÈME MULTI-ROBOT

3.1 Introduction

Dans le chapitre précédent, nous avons présenté les principales approches d'allocation des tâches à un système multi-robot, leurs avantages et leurs limites. Il en ressort que les approches distribuées sont optimales par rapport aux approches centralisées ou celles décentralisées. Dans ce chapitre, nous allons présenter une optimisation de l'algorithme HaucTioN pour une meilleure coordination des tâches dans un système multi-robot.

3.2 Schéma récapitulatif des étapes du protocole HaucTioN

Le protocole proposé dans le chapitre précédent possède des entrées, des sorties et des états intermédiaires spécifiques, pouvant être des données de statut échangées entre les robots comme “le message d'annonce à l'ouverture du tour d'enchères” ou encore le “graphe HTN” qui est une structure de données interne aux robots.

Dans la figure 9 ci-dessous :

- Les pentagones représentent les conditions à satisfaire pour passer d'une étape à une autre ;
- Les box bleus représentent les input tandis que les box jaunes représentent les outputs ;
- Les rectangles plein arrondis quant à eux représentent les procédures/fonctions ;
- Les rectangles creux représentent les blocs de procédures/fonctions.

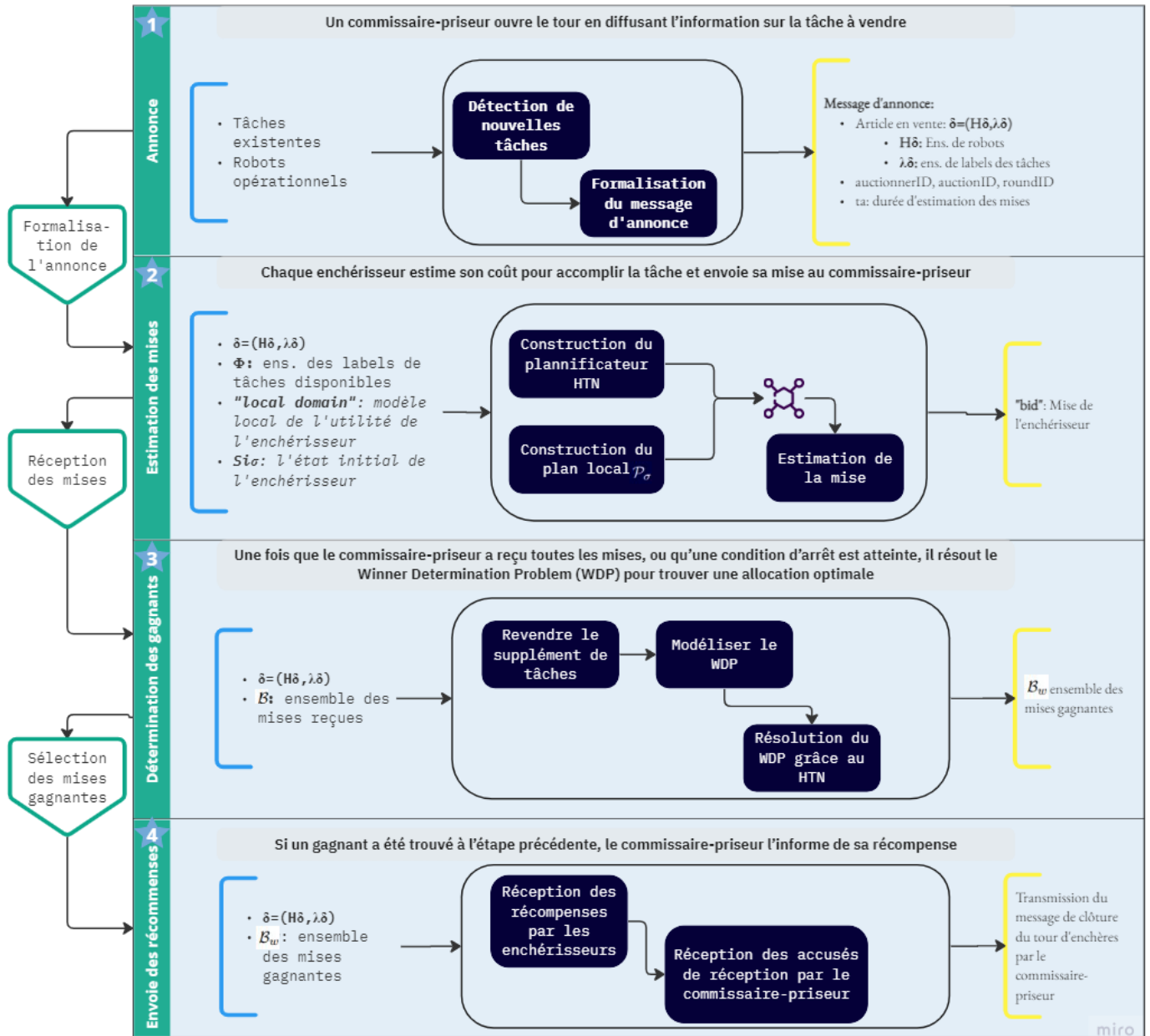


Figure 9 : Schéma logique du protocole HaucTioN.

3.2.1 Limites du protocole HaucTioN

Le protocole HaucTioN, tel que présenté au chapitre précédent, a deux principales limites :

- La complexité de l'étape de détermination des gagnants après une mise. Autrement dit, à chaque tour d'enchères, le coût calculatoire de la détermination du robot qui offre la meilleure mise pour l'exécution d'une tâche est considérablement élevé. à cela s'ajoute les contraintes de communication liées à la nature du système multi-robot (échanges dans un réseau maillé);
- Une gestion peu efficace des contraintes issues de la décision: En d'autres termes, ce protocole ne se préoccupe pas des éventuels incidents pouvant altérer la décision d'allocation, comme la perte de signal entre les robots ou encore l'arrivée d'une nouvelle tâche.

Notre amélioration va permettre au protocole optimisé HaucTioN-K pour combler les limites du protocole de base HaucTioN.

3.3 Présentation du protocole HaucTioN-K

Nous proposons dans cette section notre protocole de régionalisation par le K-Means clustering (voire annexe A) pour réduire la complexité de la détermination des gagnants. En effet, plutôt que d'évaluer les mises de tous les agents du système, celles-ci sont évaluées uniquement à l'intérieur des clusters formés. Un cluster ici peut par exemple représenter des agents situés dans un rayon géographique spécifique, ou encore les agents partageant une caractéristique particulière. Le but de cette manœuvre est de réduire l'échelle des intrants du système à l'étape d'évaluation des gagnants à chaque tour d'enchères.

3.3.1 Principe du protocole HaucTioN-K

Pour atteindre l'objectif de réduction de la complexité de l'étape de détermination des gagnants à chaque tour d'enchères, nous modifions l'étape de détermination des gagnants. Désormais plutôt que d'exécuter le WDP sur les mises effectuées par les enchérisseurs, le commissaire-priseur va d'abord regrouper ces mises reçues en fonction de leurs valeurs (en utilisant l'algorithme K-Means), puis exécutera simultanément k algorithmes de détermination des gagnants d'où le nom K-WDP attribuée à cette étape.

En ce qui concerne les autres étapes du protocole de base HaucTioN, ils restent inchangés dans le protocole optimisé HaucTioN-K.

3.3.2 Fonctionnement du protocole HaucTioN-K

Pour parvenir à ses fins, HaucTioN-K modifie la troisième étape du protocole de base, de façon à le rendre son exécution parallèle plutôt que séquentielle, tout en conservant les étapes trois autres étapes inchangées.

a) Etape 1 : Annonce

L'étape d'annonce du protocole HaucTioN-K reprend le processus du protocole HaucTioN, partant de la détermination (ou détection) des tâches disponibles à la formalisation du message d'annonce. Cette étape est responsable de la détection des nouvelles tâches et sa diffusion au système de robots, faisant ainsi appel aux capteurs et aux connecteurs réseaux des robots.

Algorithm : Etape 1: Annonce

input :

Tâches: $T = \{t_1, t_2, \dots, t_n\}$

Robots: $R = \{r_1, r_2, \dots, r_m\}$

output:

Message :

$M = \{\delta, T_a, AuctionerID, auctionID, RoundID\}$

1 Détection de nouvelles tâches

2 Formalisation du message d'annonce M

3 return (M)

A la fin de cette étape, les enchérisseurs possèdent le message M d'ouverture du tour d'enchères.

b) Etape 2 : Estimation des mises

L'estimation des mises ne subit aussi aucun changement par rapport au protocole de départ.

Après l'annonce d'une nouvelle tâche, chaque enchérisseur se prononce sur sa capacité à la résoudre. Autrement dit, les robots estiment leurs utilités pour la résolution de cette tâche. Cette estimation prend en entrée le message M d'annonce obtenu à l'étape 1, fait appel aux fonctions et structures internes aux robots, et retourne la mise B_{ri} . Tout au long de cette étape, chaque robot enchérisseur construit son réseau hiérarchique de tâche (HTN) et calcule l'utilité relative à l'exécution de cette tâche, tandis que le commissaire-priseur lui veille au respect des critères d'arrêt d'estimations des mises.

Algorithm : Etape 2: Estimation des mises

input :

L'article en vente: $\delta = \{\lambda_\delta, H_\delta\}$

Les labels: Φ

Les capacités locales: D_σ

L'état initial du robot: $S_{\sigma i}$

output:

Mise B

- 1 $H_\Gamma \leq_p H_\delta$
 - 2 $P_\sigma \leftarrow \text{New } P(D_\sigma, S_{i\sigma})$
 - 3 $P_\chi \leftarrow \text{New } P(D_\chi, S_{i\chi}, \text{tni}\chi)$
 - 4 $B \leftarrow \nabla(P_\chi, P_\sigma, H_\Gamma, \delta, \Phi)$
 - 5 return (B)
-

A la fin de cette étape, le commissaire-priseur possède l'ensemble des mises $B = \{B_1, B_2, \dots, B_n\}$ des n robots du MAS et considérant comme mises nulles les mises non reçues (qui peuvent survenir lors d'une panne réseau).

c) Etape 3 : Détermination des K gagnants (K-WDP)

Cette étape conserve son principe tel qu'énoncé dans le protocole initial, mais est désormais utilisée dans un système parallélisé en K branches plutôt qu'en série. En effet, afin d'éliminer le goulot d'étranglement attribué à cette étape dans le protocole initial, la résolution du WDP (problème de détermination des gagnants) est exécutée en parallèle dans le système grâce à l'introduction de l'algorithme K-Means pour diviser en petits blocs les mises $B_k = \{B_{k1}, B_{k2}, \dots, B_{kk}\}$ avant de les résoudre parallèlement en k blocs de WDP: la formation des clusters avec K-Means, l'exécution parallèle du WDP et la réunion des ensembles de mises gagnantes.

i) La formation des clusters avec K-Means

A partir des mises B reçues à l'étape 2 et du nombre K de clusters choisis, le commissaire-priseur exécute l'algorithme K-Means afin de former les K clusters de mises $B_k = \{B_{k1}, B_{k2}, \dots, B_{kk}\}$, où B_{kj} est le cluster de centre j contenant les mises de robots.

En principe, l'algorithme de classification non supervisé K-Means permet de former K groupes d'individus à partir d'une population de taille n et d'un nombre entier K , en se basant à chaque fois sur le calcul de la distance du point considéré aux centres (voire annexe A). On dira alors qu'un robot appartient à un cluster si et seulement si sa mise est plus proche du centre de ce cluster que des centres des autres clusters.

A la sortie de cette étape de clustering, le commissaire-priseur dispose alors des clusters de mise.

ii) l'exécution parallèle du WDP

A partir des K clusters de mises obtenues en (i), le commissaire-priseur déclenche le déroulement en parallèle du WDP afin de déterminer dans chacun d'eux la liste des éventuels gagnants.

En effet, un nombre K de WDP sont enclenchés simultanément (en parallèle). Chacun d'eux a pour objectif de déterminer les mises gagnantes de chaque cluster, en veillant à respecter les contraintes du système MRTA de nature ST-MR-IA comme le fait qu'un enchérisseur ne gagne qu'au plus une mise ou encore le fait que les mises des enchérisseurs puissent porter sur le même article en vente.

Elle prend en entrée les clusters de mises $B_k = \{B_{k1}, B_{k2}, \dots, B_{kk}\}$, et retourne la liste des mises gagnantes

$B_{k_w} = \{B_{k_w1}, B_{k_w2}, \dots, B_{k_wk}\}$, de chaque groupe $B_{k_{wi}}$ du tour d'enchères.

iii) La réunion des ensembles de mises gagnantes

La réunion des ensembles de mises gagnantes est une étape propre à l'optimisation proposée, et qui découle directement de la parallélisation de l'étape précédente (WDP).

Elle prend en entrée les k groupes de mises gagnantes $B_{k_w} = \{B_{k_w1}, B_{k_w2}, \dots, B_{k_wk}\}$, compare les mises et retient la mise maximale de chaque groupe et les réunit dans un ensemble décroissant de mises gagnantes finales $B_w = \{B_{w1}, B_{w2}, \dots, B_{wk}\}$.

Algorithm : Etape 3: K-WDP

input :

Mises: $B = \{B_1, B_2, \dots, B_n\}$

Article en vente: $\delta = \{H_{\delta}, L_{\delta}\}$

output:

Mises gagnantes: $B_w = \{B_{w1}, B_{w2}, \dots, B_{wn}\}$

```

1 // Clustering des mises
   $B_k \leftarrow \text{K-Means}(B)$ 
2 // Détermination des K gagnants
   $B_{wk} \leftarrow \text{K-WDP}(B_k, \delta)$ 
3 // Réunion des mises gagnantes
   $B_w \leftarrow \bigcup_{i=1}^k B_{wi}$ 
4 return ( $B_w$ )

```

A la fin de cette étape, le commissaire-priseur détient la liste des gagnants du tour d'enchères.

d) Etape 4 : Distribution des récompenses et clôture du tour d'enchère

Tout comme dans le protocole initial, le commissaire-priseur distribue les récompenses aux enchérisseurs, puis grâce à un accusé de réception, il veille à la distribution des récompenses si non quoi l'article sera revendu au prochain tour d'enchères. La récompense distribuée aux robots est proportionnelle aux mises effectuées $R_i = \varepsilon B_i$, et son coefficient ε est propre à chaque commissaire-priseur.

3.3.3 Schéma général du protocole HaucTioN-K

Le schéma 10 ci-dessous présente une vue synoptique du protocole HaucTioN-K. Dans cette figure :

- Les pentagones représentent les conditions à satisfaire pour passer d'une étape à une autre ;
- les box bleus représentent les input ;
- Les box jaunes représentent les outputs ;
- Les rectangles plein arrondis représentent les procédures/fonctions ;
- Les rectangles creux représentent les blocs d'instruction.

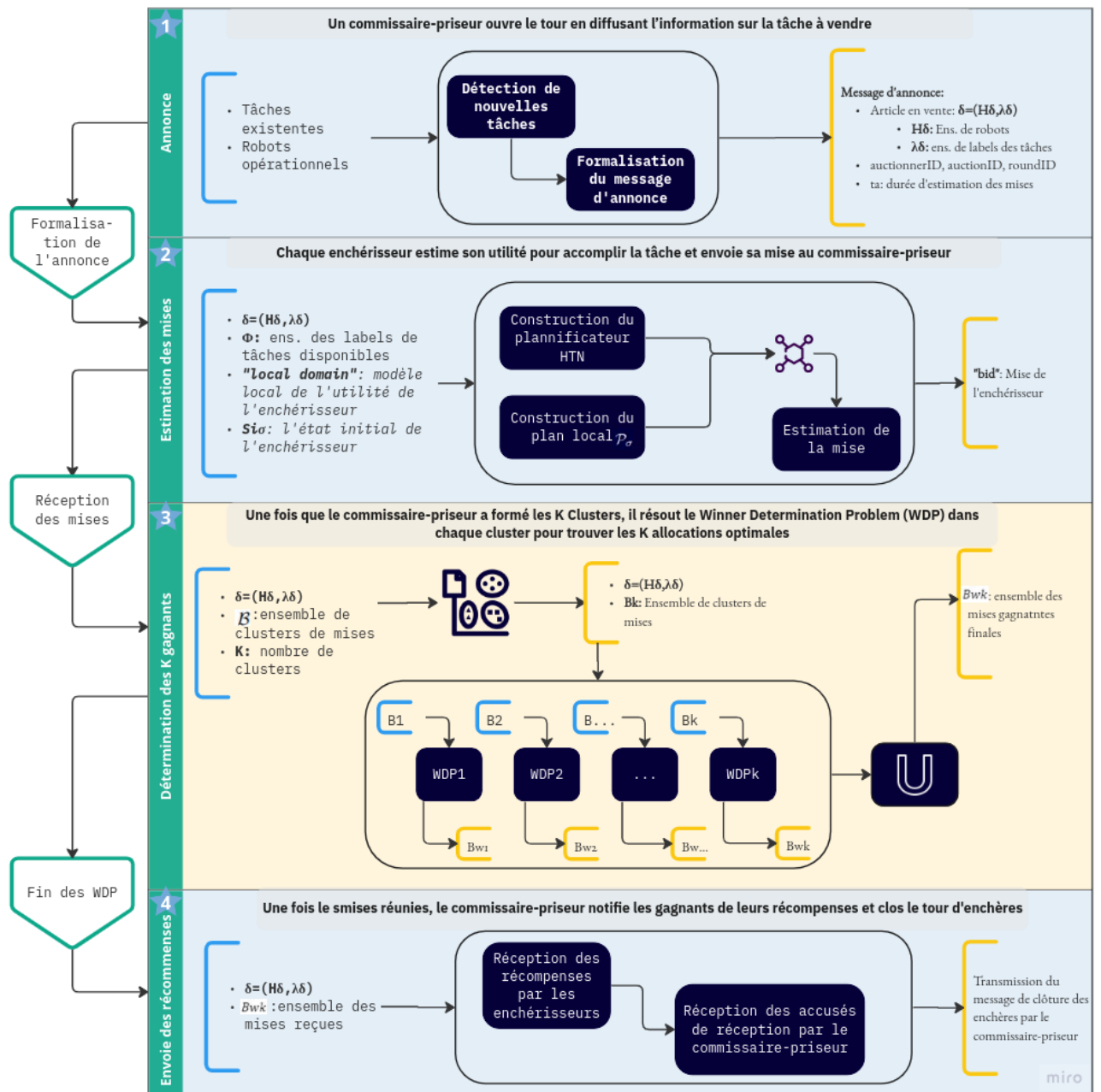


Figure 10 : Schéma récapitulatif du protocole HaucTioN-K.

3.3.4 Validation du protocole HaucTioN-K

Les auteurs du protocole HaucTioN ont prouvé sa validité en procédant notamment à une expérimentation sur le cas de détection et de neutralisation de mines dans un milieu sous-marin. Puisque HaucTioN-k est un protocole dérivé de lui par modification de l'étape de détermination des gagnants, nous procéderons à sa validation en analysant la complexité temporelle et les spécifications (entrées, sorties, préconditions et postconditions), notamment celles de l'étape modifiée.

a) Analyse des spécifications de HaucTioN-K

A l'aide de préconditions et postconditions bien établies (voire tableau 2), chaque étape du protocole possède son propre déclencheur et ses conditions d'arrêt. De plus, avec des entrées et des sorties câblées les

unes aux autres, les étapes sont consécutivement interdépendantes. Nous pouvons en déduire que le protocole HaucTioN-K est stable; car pour toute donnée admissible, elle produit une solution unique et cette solution dépend continûment des données. Dans le tableau 2 ci-dessous :

- En bleu sont les étapes du protocole de base HaucTioN ;
- En jaune celles du protocole HaucTioN-K ;

#	Etape		Entrées	Sorties	Précondition	Postcondition
1	Annonce		• N Robots • M Tâches	Message d'annonce	• $M \neq 0$ • $N \neq 0$	Le message d'annonce doit contenir au moins un article en vente
2	Estimation des mises		Message d'annonce	Mises B		• Les mises non estimées sont considérées comme nulles • $2 \leq \text{Card}(B)$
3	Détermination des gagnants (K-WDP)	Clustering des mises	• Mises B • Nombre de clusters K	K groupes de mises	• Au moins K mises non nulles • $2 \leq k \leq \text{Card}(B)$	Au moins K clusters de mises non nulles
		Parallel WDP	K groupes de mises	Les gagnants des K groupes	Au moins une mise gagnante trouvée	K ensemble de mises, possiblement vides
		Réunion des mises gagnantes	Les gagnants des K groupes	La liste des gagnants		
4	Distribution des récompenses		La liste des gagnants	Récompenses		Accusé de réception reçu par le commissaire-priseur miro

Tableau 2 : Spécifications des étapes du protocole HaucTioN-K.

b) Analyse de la complexité de HaucTioN-K

Le protocole HaucTioN-K innove face à son protocole mère en apportant une touche de parallélisation de l'étape de détermination des gagnants.

Dès lors, la différence de complexités des deux protocoles se joue sur l'étape 3.

Équation 1 : Formulation de la complexité du protocole HaucTioN

$$\begin{aligned}
 \mathcal{O}(\text{HaucTioN}) &= \sum_{i=1}^4 \mathcal{O}(\text{Step}_i) \\
 &= [\mathcal{O}(\text{Step}_1) + \mathcal{O}(\text{Step}_2) + \mathcal{O}(\text{Step}_4)] + \mathcal{O}(\text{Step}_3) \\
 &= [\mathcal{O}(\text{Step}_1) + \mathcal{O}(\text{Step}_2) + \mathcal{O}(\text{Step}_4)] + \mathcal{O}(\text{WDP}_{n,m})
 \end{aligned}$$

Équation 2: Formulation de la complexité temporelle du protocole HaucTioN-K

$$\begin{aligned}
 \mathcal{O}(\text{HaucTioN-K}) &= \sum_{i=1}^4 \mathcal{O}(\text{Step}_i) \\
 &= [\mathcal{O}(\text{Step}_1) + \mathcal{O}(\text{Step}_2) + \mathcal{O}(\text{Step}_4)] + \mathcal{O}(\text{Step}_3) \\
 &= [\mathcal{O}(\text{Step}_1) + \mathcal{O}(\text{Step}_2) + \mathcal{O}(\text{Step}_4)] + \mathcal{O}(\text{K-WDP}_{n,m})
 \end{aligned}$$

Hypothèses :

- Le protocole HaucTioN converge, selon Antoine Milot et al.[2] ;
- Le protocole HaucTioN-K converge aussi, puisqu'il dérive de deux protocoles convergents (HaucTioN et K-Means).

Par conséquent, nous pouvons effectuer une opération asymptotique entre leurs complexités.

Équation 3 : Calcule de la différence asymptotique de complexité entre les protocoles HaucTioN et HaucTioN-K

$$\begin{aligned}
\Delta_{\text{complexité}} &= \mathcal{O}(\text{HaucTioN-K}) - \mathcal{O}(\text{HaucTioN}) \\
&= \mathcal{O}(\text{K-WDP}_{n,m}) - \mathcal{O}(\text{WDP}_{n,m}) \\
&= [\mathcal{O}(\text{K-Means}) + \mathcal{O}(\text{//WDP}) + c] - \mathcal{O}(\text{WDP}_{n,m}) \\
&\quad \text{where //WDP is the parallel WDP} \\
&\quad c \text{ is the complexity of WDPs result's reunion} \\
&= [\mathcal{O}(t * d * k * n) + \frac{1}{k} \mathcal{O}(\text{WDP}_{n,m}) + c] - \mathcal{O}(\text{WDP}_{n,m}) \\
&\quad \text{where } t \text{ is the value of K-Means iteration} \\
&\quad d \text{ the data dimension} \\
&\quad k \text{ the number of centroids} \\
&\quad n \text{ the data length} \\
&= [\lambda \mathcal{O}(n) + \frac{1}{k} \mathcal{O}(\text{WDP}_{n,m}) + c] - \mathcal{O}(\text{WDP}_{n,m}) \\
&\quad \text{where } \lambda = t * d * k \\
&= [(\lambda \mathcal{O}(n) + c) + \frac{1}{k} \mathcal{O}(\text{WDP}_{n,m})] - \mathcal{O}(\text{WDP}_{n,m}) \\
&\cong \frac{1}{k} \mathcal{O}(\text{WDP}_{n,m}) - \mathcal{O}(\text{WDP}_{n,m}) \\
&\quad \text{assuming } \mathcal{O}(n) \ll \mathcal{O}(\text{WDP}_{n,m}) \\
&= \mathcal{O}(\text{WDP}_{n,m}) \left(\frac{1}{k} - 1 \right) \\
&< 0 \\
&\quad \text{because } \mathcal{O}(\text{WDP}_{n,m}) > 0 \text{ and } \left(\frac{1}{k} - 1 \right) < 0
\end{aligned}$$

L'analyse asymptotique ci-dessus montre que la différence de complexité des deux protocoles est négative ; par conséquent, la complexité temporelle du protocole HaucTioN-K est inférieure à celle du protocole de base HaucTioN.

3.3.5 Avantages et limitations du protocole HaucTioN-K

Le protocole HaucTioN-K offre quatre principaux atouts:

- La réduction de la complexité du protocole HaucTioN. En éliminant le goulot d'étranglement observé à l'étape 3 grâce à la parallélisation des opérations ;
- HaucTioN-K est plus adapté pour des missions temps réel que son modèle de base, puisqu'en regroupant les mises des enchérisseurs dans des clusters, il facilite la réalisation des missions aux tâches hétérogènes ;
- HaucTioN-K est plus résistant aux pannes de communication. En effet, en cas de pannes d'un robot, les autres robots du même cluster rachèteront la tâche à la prochaine vente.

La principale limitation du protocole HaucTioN face à son protocole de base est celle attribuée à la capacité du MAS à exécuter l'algorithme de clustering K-Means. En effet, Les robots du MAS qui exécutent déjà le protocole de base HaucTioN doivent posséder suffisamment de ressources intrinsèques (RAM, processeur) pour dérouler le processus K-MEANS et le K-WDP sans faillir aux autres opérations qui lui étaient autrefois attribuées.

3.4 Conclusion

Dans ce chapitre, nous avons présenté le protocole de décision HaucTioN-K, qui est une amélioration du protocole HaucTioN proposé par Antoine Millot et al. Nous avons également montré que ce modèle est stable, cohérent et de complexité inférieure à sa version initiale. Dans le chapitre suivant, nous allons effectuer une étude expérimentale du protocole HaucTioN-K.

CHAPITRE 4 : EXPERIMENTATION DU PROTOCOLE HAUTION-K

4.1 Introduction

Dans le chapitre précédent, nous avons présenté le protocole HaucTioN-K, en précisant ses apports et sa principale limitation. Dans ce chapitre, nous effectuerons une étude expérimentale du protocole HaucTioN-K.

4.2 Description de l'environnement d'expérimentation

Pour mener une étude expérimentale du protocole HaucTioN-K, nous mettons en place deux environnements afin de mieux mesurer les performances du système MRTA:

- L'environnement technique constitué de matériels et de logiciels ;
- L'environnement de simulation permettant.

4.2.1 Architecture technique

Comme support matériel, nous utilisons une machine réelle disposant de 3 cœurs, d'un processeur Intel ® Core(™) i3-1005G1 CPU @ 1.20GHz(4 CPUs), ~1.2GHz et de 4GB de RAM.

Le support logiciel quant à lui est exclusivement fait de MATLAB ® & Simulink, licence étudiante, Version R2016b, release 9.1.0.441655, 64-bit, tournant sur un système Ubuntu 20.4.

4.2.2 Architecture de simulation

L'environnement de simulation présente un fonctionnement sur trois couches:

- Une couche backend propre à chaque composante, constituée du:
 - protocole de décision ;
 - protocole de supervision ;
 - HTN (Hierarchical Task Network) ;
 - module de gestion des communication ;
- la couche MiddleWare encore appelée couche de simulation, commune aux composantes du système, et permet de simuler les couches fonctionnelles des robots et leurs interactions avec l'environnement ;
- la couche frontend, permettant d'observer l'évolution du système, et constitué :
 - du module de graphisme et monitoring ;
 - des composantes de simulation des interactions entre les composantes.

La figure 11 ci-dessous présente un schéma récapitulatif de l'architecture de simulation.

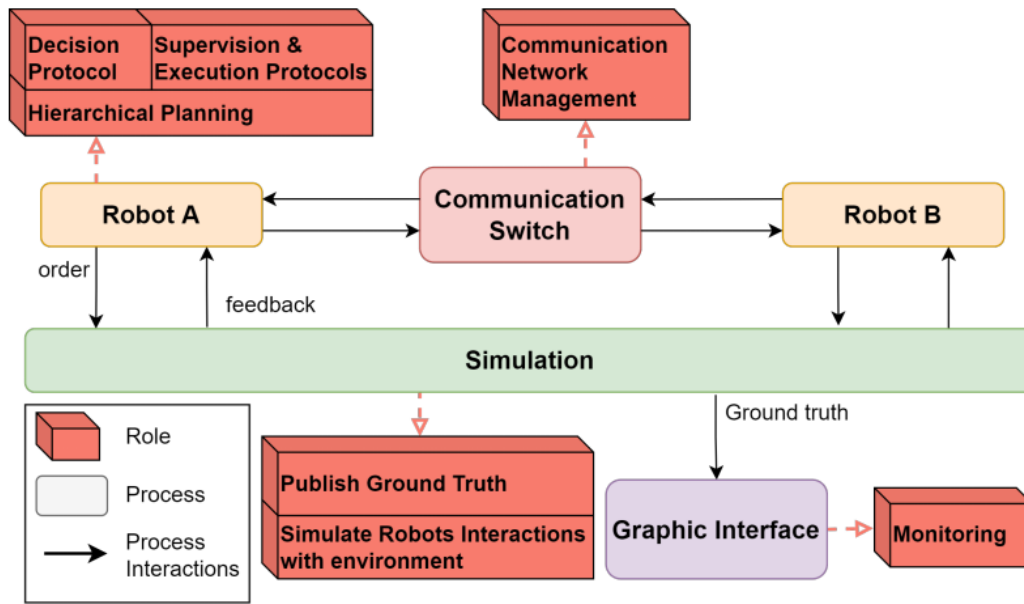


Figure 11 : Schéma de l'architecture de simulation

4.2.3 Les données de simulation

Pour mettre en œuvre le protocole HaucTioN, nous testons trois cas de figures, selon que le nombre de robots est égal au nombre de tâches, ou encore le nombre de tâches largement supérieur au nombre de robots, puis inversement.

A des fins de simulations, les robots sont représentés par les équations de vitesses (fonction objectif) de la forme $Y_t(x) = a_t x^2 + b_t x + C_0$, où x représente le coût de réalisation de la tâche en cours en considérant les paramètres a , b et c du robot à cet instant t .

Cas de figure	Nombre d'agent N	Nombre de tâches M
Cas 1	50	10
Cas 2	10	10
Cas 3	10	50

Tableau 3 : Tableau récapitulatif des scénarios de simulation

4.2.4 Les métriques d'évaluation

Pour mener à bien cette simulation, nous constituons un environnement représenté par un ensemble de marqueurs et légendes afin de faciliter la représentation des protocoles. Nous avons par exemple:

- Le système est peuplé de n robots qui évoluent dans le plan orthonormé de repère $(O; X; Y)$. Ces robots sont représentés par des étoiles mobiles ayant la même couleur que leur trajectoire ;
- Le système est aussi peuplé de m tâches représenté par des points noirs autour desquels nous pouvons éventuellement avoir un cercle concentrique (à l'intérieur duquel la tâche peut-être considérée comme réalisée).
- Le temps est mesuré en secondes(s), et la dimension en mètres(m).

Les principales métriques d'évaluation du système MRTA sont l'utilité, de la récompense et du coût.

a) La récompense (reward)

Il s'agit du gain obtenu par l'accomplissement de la tâche par le robot. Dans la pratique, cette récompense est octroyée à un robot par le commissaire-priseur et est proportionnelle à l'utilité de ce dernier. Aussi, il permet au système MRTA d'étiqueter les robots du système en fonction de leur rendement au cours de leur évolution dans l'environnement de travail.

b) Le coût (cost)

Le coût de réalisation C_{rt} d'une tâche T par un robot R est une valeur numérique positive qui indique la quantité de ressource exigée à ce robot pour réaliser complètement la tâche. Celle-ci peut correspondre par exemple à l'énergie électrique consommée, de la distance parcourue ou de la masse en kilogrammes d'un objet à transporter.

c) L'utilité (utility)

L'utilité d'un robot est une valeur numérique grâce à laquelle un robot exprime son aptitude à réaliser une tâche. Idéalement, l'objectif est d'optimiser directement les performances globales du système, mais cette quantité est souvent difficile à mesurer lors de l'exécution du système. Par conséquent, l'utilité d'un robot doit être mesurée par rapport à un objectif spécifique. Ses valeurs dépendent de la quantité Q_{rt} de ressources mises à disposition pour réaliser la tâche T et des ressources dépensées (C_{rt}) par ce robot à l'exécution de cette tâche.

$$U_{RT} = \begin{cases} Q_{RT} - C_{RT} & \text{if } R \text{ is capable of executing} \\ & T \text{ and } Q_{RT} > C_{RT} \\ 0 & \text{otherwise} \end{cases}$$

4.2.5 Notion de fonction objectif

Afin de pouvoir prendre des décisions, les robots ont besoin de raisonner sur des modèles mathématiques formels. Ces modèles définissent s'ils sont en capacité d'effectuer la tâche, mais peuvent également caractériser d'autres informations. Par exemple, il est possible de déterminer à quel prix (coût) et pour quel degré de réussite (récompense) un robot peut réaliser une tâche. L'éligibilité d'un robot pour une tâche dépend donc non seulement d'un filtrage classique sur les capacités requises, mais également d'une notion de rentabilité. L'établissement d'un modèle mathématique encapsulant ces informations permet de définir une fonction objectif qui pourra être minimisée ou maximisée selon des critères définis par l'opérateur afin d'optimiser l'accomplissement de la mission. Généralement, la définition d'un tel modèle se fait non seulement autour des notions de coût et de récompenses, mais aussi des notions de priorité de la tâche et d'aptitude (à quel point le robot est pertinent pour réaliser une tâche, s'il s'agit par exemple de déplacement sur un terrain plat, le robot le plus rapide sera le plus indiqué).

Quelques exemples de fonction objectif :

- Minimiser le coût ou maximiser l'utilité du pire agent ;
- Minimiser la somme des coûts individuels / Maximiser la somme des utilités individuelles ;
- Minimiser le coût moyen par tâche.

4.3 Utilisation du protocole HaucTioN

Les figures ci-dessous permettent de visualiser l'évolution des trois métriques (récompense, utilité, coût) du système MRTA avec le protocole HaucTioN dans les trois cas de figures annoncés ci-dessus:

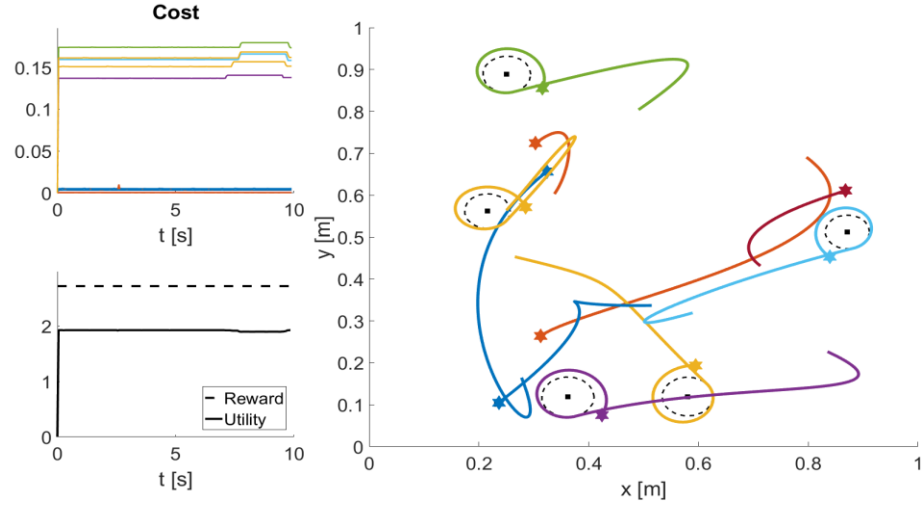


Figure 12 : Utilisation du protocole HaucTioN pour $\{N=10 \text{ et } M=10\}$

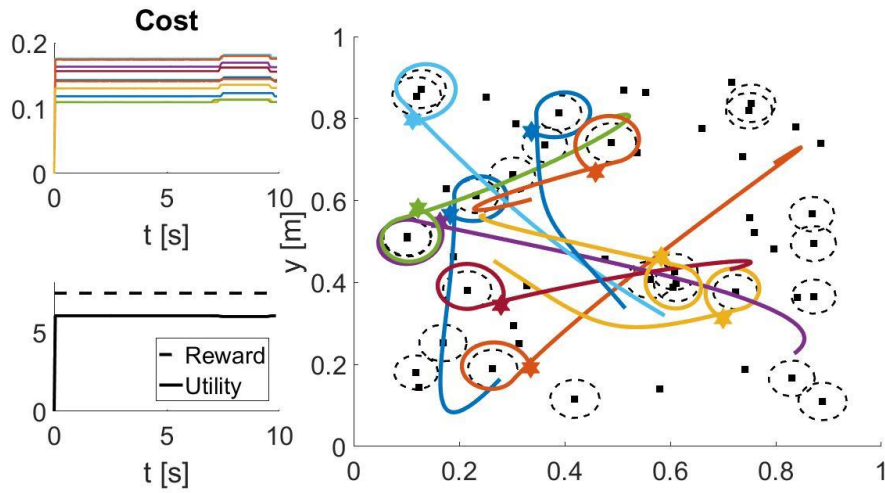


Figure 13 : Utilisation du protocole HaucTioN pour $\{N=10 \text{ et } M=50\}$

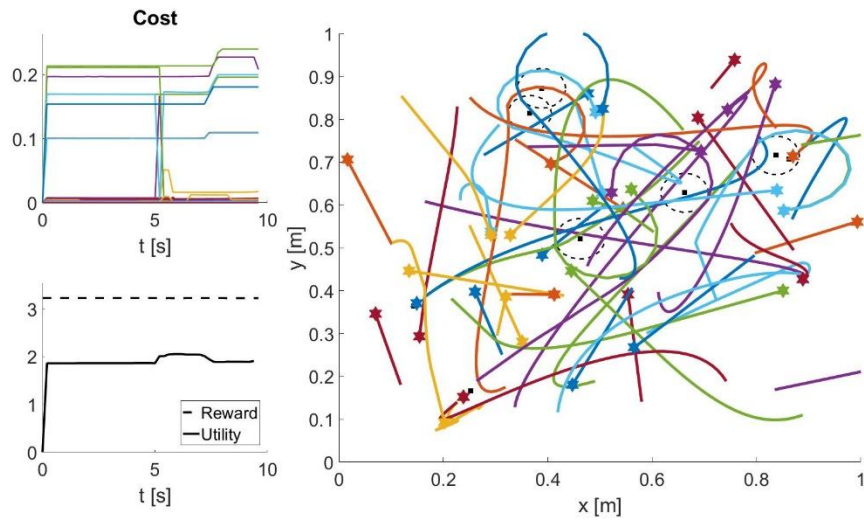


Figure 14 : Utilisation du protocole HaucTioN pour $\{N=50 \text{ et } M=10\}$

4.4 Utilisation du protocole HaucTioN-K

Les figures ci-dessous permettent de visualiser l'évolution des trois métriques (récompense, utilité, coût) du système MRTA avec le protocole HaucTioN-K dans les trois cas de figures annoncés ci-dessus:

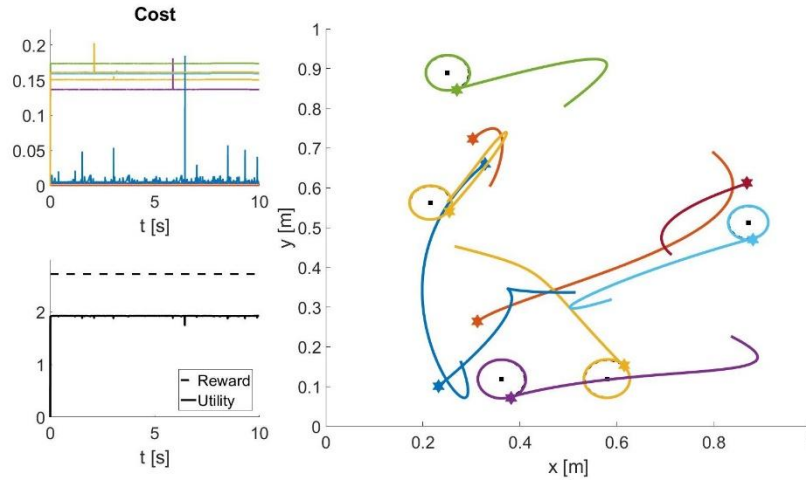


Figure 15 : Utilisation du protocole HaucTioN-K pour $\{N=10 \text{ et } M=10\}$

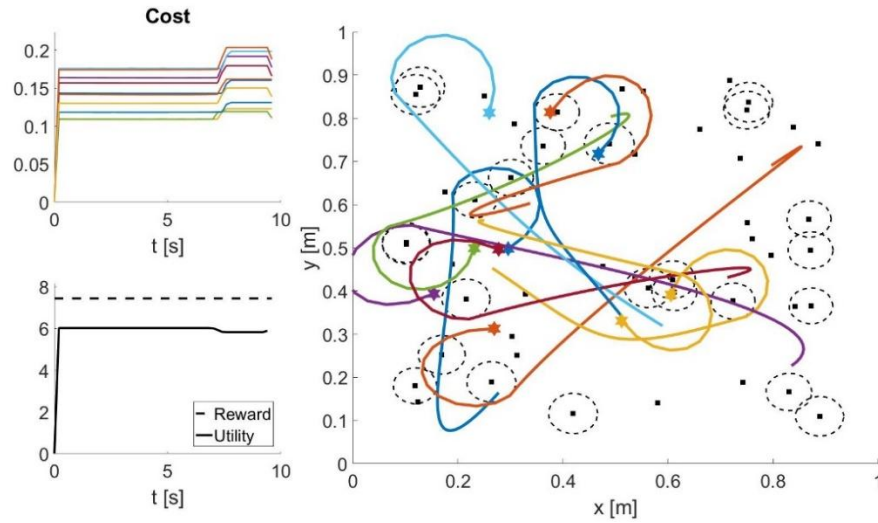


Figure 16 : Utilisation du protocole HaucTioN-K pour $\{N=10 \text{ et } M=50\}$

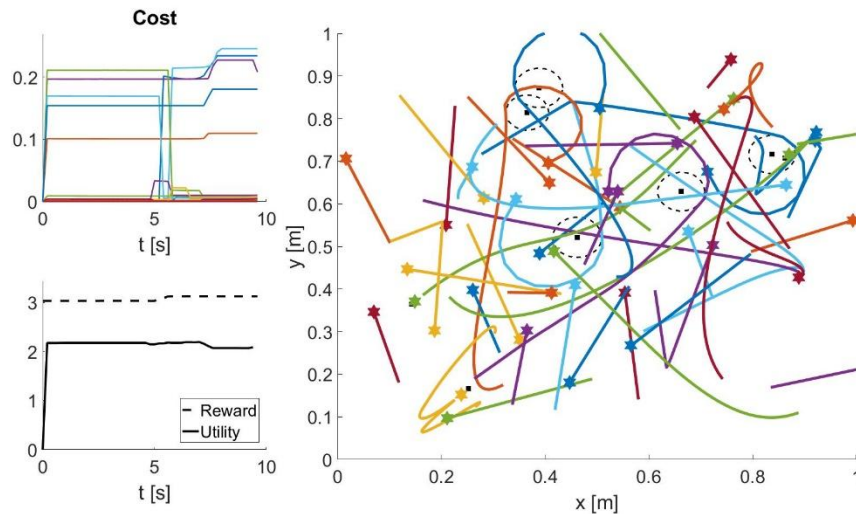


Figure 17 : Utilisation du protocole HaucTioN-K pour $\{N=50 \text{ et } M=10\}$.

4.5 Validation expérimentale du protocole HaucTioN-K

Les expérimentations réalisées ci-dessus permettent de relever des comportements différents du système d'allocation des tâches pour les mêmes entrées.

4.5.1 Analyse des métriques: utilité, récompense et coût

A partir des trois cas de figures expérimentales, nous avons dressé un tableau de comparaison des métriques, permettant de d'observer que les métriques d'utilité et de récompenses du protocole HaucTioN-K sont supérieures à celles du protocole de base HaucTioN. Toutefois, le coût d'allocation varie de façon significative, allant parfois jusqu'à s'annuler pour certains agents ou atteindre des pics pour d'autres, montrant ainsi l'efficacité dans la coordination du multi-agent. Toutefois, dans deux des trois cas de figures de la simulation, les consommations deviennent rapidement constantes tout au long des tours d'enchères, comme l'illustrent le tableau 4 et la figure 18 ci-dessous.

	Utilité		Récompense		Coût	
	HaucTioN	HaucTioN-K	HaucTioN	HaucTioN-K	HaucTioN	HaucTioN-K
N=10; M=10	1.892	1.984	2.785	2.92	0.158	0.15
N=10; M=50	5.32	6.71	5.25	6.82	0.176	0.153
N=50; M=10	1.908	2.243	3.125	3.115	0.127	0.078

Tableau 3 : Tableau de comparaison des métriques utilité et récompense des protocoles HaucTioN et HaucTioN-K

Par ailleurs, nous relevons que le coût des allocations varie de façon beaucoup plus uniformément dans le protocole amélioré HaucTioN-K.

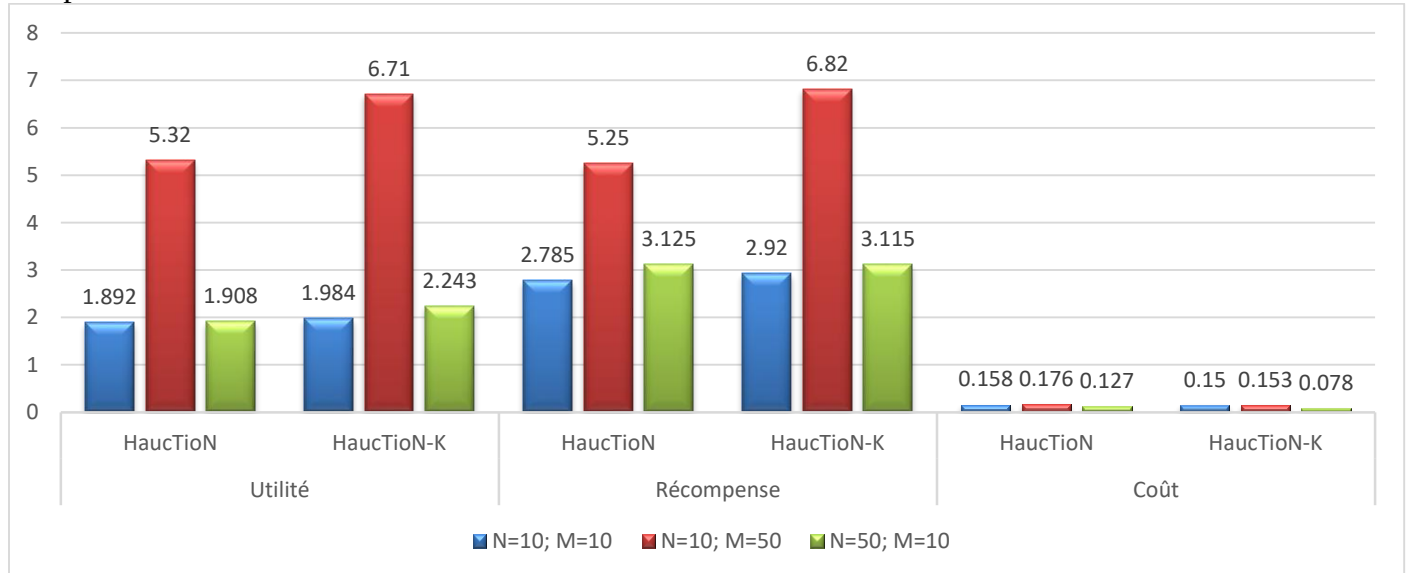


Figure 18 : Graphique de comparaison des protocoles HaucTioN et HaucTioN-K

4.5.2 Interprétation des variations de métriques

Dans l'étude expérimentale ci-dessus, nous observons la réduction des valeurs des métriques de récompense, d'utilité et de coût, mais aussi une réduction globale des variations de coût entre deux allocations. Cette observation permet de déduire qu'en face d'un problème d'allocation de tâches à un système multi-agent, le protocole HaucTioN-K est meilleur que le protocole HaucTioN en terme d'utilité, de récompense et du coût de ressources consommées.

4.6 Conclusion

Dans ce chapitre, nous avons expérimenté l'allocation des tâches à un système d'agent en utilisant les protocoles HaucTioN et sa dérivée HaucTioN-K, sur des jeux de données variées afin de mettre en lumière les trois circonstances principales d'utilisation d'un système multi-robot. Dans chacune de ces circonstances, le protocole HaucTioN-K s'est montré bien plus optimal que le protocole initial du fait de la réduction du coût d'allocation, par une augmentation de l'utilité et de la récompense du système multi-robot.

CHAPITRE 5 : CONCLUSION GENERALE ET PERSPECTIVES

5.1 Conclusion générale

Les missions de sauvetage pendant les incidents comme les incendies et les inondations connaissent une montée croissante de l'utilisation des drones et des robots divers. Afin d'accroître leur efficacité dans le sauvetage de biens et des vies humaines lors d'une mission dans un environnement à risque, il se pose un problème à savoir lequel robot effectue laquelle tâche afin d'atteindre les objectifs de la mission tout en réduisant son coût global.

L'allocation des tâches dans un système multi-robot vise à attribuer un ensemble de tâches à un ensemble de robots qui évoluent dans un environnement sous contraintes. Elle se fait par des familles de méthodes que nous avons regroupées en familles centralisées, décentralisées ou encore distribuées. La littérature montre que les méthodes de la famille distribuée sont meilleures en terme d'utilité et d'opérationnalité par rapport aux deux autres; l'une d'entre ces méthodes est la méthode d'allocation par enchères, qui déroule le processus de vente aux enchères d'un article dans un marché; en effet, le robot qui détecte une tâche nouvelle ou non achevée prend le rôle de commissaire-priseur, initialise le tour d'enchères et attribue la tâche en question au robot qu'il jugera le plus apte à l'exécuter. Le robot qui remporte le tour d'enchères n'est pas seulement celui qui a la mise maximale, car le commissaire-priseur le sélectionne en tenant aussi compte de sa connaissance du système multi-robot. Malheureusement, lorsqu'il est appelé à se prononcer sur le robot qui va effectuer la tâche, le commissaire-priseur résout un problème de décision dont la complexité croît rapidement avec la taille du système (le nombre de tâches, le nombre de robots et les contraintes d'allocation). Dans la pratique, le protocole HaucTioN proposé par Antoine Millot et al.[2] fait face à un goulot d'étranglement, réduisant considérablement son efficacité.

Aux vues de cette limite, nous avons proposé un protocole HaucTioN-K, qui est dérivé du protocole de base HaucTioN, en introduisant la parallélisation des opérations de l'étape de détermination des gagnants afin de réduire la taille des entrées du problème de détermination du gagnant (WDP – Winner Determination Problem) du tour d'enchères. HaucTioN-K et son protocole de base ont trois étapes similaires. L'étape 1 appelée l'annonce, qui permet au commissaire-priseur d'informer le reste des robots sur l'introduction en vente d'une tâche nouvelle ou inachevée. L'étape 2 appelée l'estimation des mises au cours de laquelle les enchérisseurs prononcent leurs mises dans un certain délai. L'étape 4 appelée la clôture du tour d'enchères, au cours de laquelle le commissaire-priseur notifie les gagnants du tour d'enchère. L'étape 3 quant à elle, appelée détermination des gagnants, est le nœud qui distingue les deux protocoles. En effet, lorsque HaucTioN détermine directement les mises gagnantes à cette étape et fait face au goulot d'étranglement, HaucTioN-K se sert de l'algorithme de classification non supervisée K-Means pour former K groupes de mises avant d'exécuter son algorithme WDP en parallèle dans chacun de ces clusters.

L'analyse des spécifications du protocole HaucTioN-K a relevé qu'il reste stable et cohérent. Par ailleurs, sa complexité temporelle asymptotique s'est révélée inférieure à celle de son protocole de départ. Par suite, l'étude expérimentale réalisée dans un environnement de simulations, a permis de montrer grâce à trois cas de figure (d'abord autant de robots que de tâches, puis plus de robots que de tâches et enfin, plus de tâches que de robots) que le protocole HaucTioN est optimale par rapport à son protocole de base.

5.2 Perspectives

Au cours de notre travail, nous avons relevé deux limites attribuées au protocole d'allocation des tâches HaucTioN : le goulot d'étranglement à l'étape de détermination des gagnants pour laquelle nous avons adressé une solution, et la gestion efficace des contraintes issues de la décision qui fera l'objet d'une étude prochaine afin de doter au protocole HaucTioN-K la faculté d'analyser l'impact de la décision prise par chaque robot sur l'évolution du multi-robot.

Par suite, à des fins d'analyse théorique du modèle HaucTioN-K, nous procéderons à une analyse qualitative. Il sera question, au-delà des chiffres, de mettre en exergue des régularités et d'expliquer la signification d'un éventuel comportement isolé du système dans les circonstances normales.

Aussi, l'étude expérimentale du protocole HaucTioN-K reste encore éloignée de la réalité des opérations de sauvetage dans un environnement d'incident. Dans la suite de nos travaux, il sera question de l'éprouver dans un environnement qui change dynamiquement (suite à la perte d'un robot par exemple) et même de le déployer dans un environnement réel d'incident.

BIBLIOGRAPHIE

- [1] Aakriti Agrawal, Senthil Hariharan, Amrit Singh Bedi, Dinesh Manocha, “DC-MRTA: Decentralized Multi-Robot Task Allocation and Navigation in Complex Environments”, The International Conference on Intelligent Robots and Systems (IROS), 2022.
- [2] Antoine Milot, Estelle Chauveau, Simon Lacroix, Charles Lesire Cabaniols, “Allocation par enchères et planification hiérarchique pour un système multi-robot, application au cas de la chasse aux mines”, JFSMA 2022: 30èmes Journées Francophones sur les Systèmes Multi-Agents, Juin 2022
- [3] Bhivraj Suthar, Rajesh Mahadeva, Saurav Dixit, Vinay Kumar, K. Arun Rishab, Arora, and Suniana Ahuja, “Robotic Drone Arm for civil structures inspection: Challenges and Future Directions”, International Conference on Music Perception and Cognition, 2023
- [4] Biran P. Gerkey, Maja J. Mataric, “A formal analysis and taxonomy of task allocation in multi-robot systems”, The International Journal of Robotics Research, 2004
- [5] Camélia Slimani, “Towards I/O-efficient Edge Intelligence algorithms : K-means and Random Forests”, Université de Bretagne Occidentale – Brest, 2023
- [6] G. Ayorkor Korsah, M. Bernardine Dias, Anthony Stentz, “A Comprehensive Taxonomy for Multi-Robot Task Allocation”, The International Journal of Robotics Research, 2013
- [7] Longtian Fu et Yumei Yu, “Design of Topic Clustering Algorithm Based on K-Means” Scientific Journal of Intelligent Systems Research, 2019
- [8] Martina Lippi, Alessandro Marino, “A Mixed-Integer Linear Programming Formulation for Human Multi-Robot Task Allocation”, The International Conference on Robot and Human Interactive Communication (RO-MAN), 2021
- [9] Mohamed Badreldin, Ahmed Hussein, and Alaa Khamis, “A Comparative Study between Optimization and Market-Based Approaches to Multi-Robot Task Allocation”, Advances in Artificial Intelligence, 2013
- [10] Shinkyu Park, Yaofeng Desmond Zhong, and Naomi Ehrich Leonard, “Multi-Robot Task Allocation Games in Dynamically Changing Environments”, the International Conference on Robotics and Automation (ICRA), 2021
- [11] Siddhartha Cheruvu, Yi Chen, Aaron Arul Maria John , “Multi-Robot Task Allocation in Disaster Response”, The Arizona State University, 2020
- [12] Tom M MITCHELL et al. “Machine learning”, Burr Ridge, IL : McGraw Hill, 1997
- [13] Yifan Bai, Christoforos Kanellakis, George Nikolakopoulos, “SA-reCBS: Multi-robot task assignment with integrated reactive path generation”, Cornell University, 2023

ANNEXE A : EXPLICATIONS SUR K-MEANS

A. Concepts généraux du Machine Learning

Définition 1. Le Machine Learning ou Apprentissage Automatique désigne la capacité d'un programme d'apprendre, c'est-à-dire modifier son comportement sur la base d'une expérience, sans que cette modification ne soit explicitement programmée[5].

Définition 2. Le Data-Set ou Jeu de Données en français désigne une collection de valeurs quantitatives et/ou qualitatives. Chaque valeur est associée à une variable et une observation. Ainsi, un data-set peut être vu comme étant une matrice où les lignes sont des observations, et les colonnes des variables qui caractérisent ces observations[5].

Dans la littérature, les algorithmes de ML sont classés selon les caractéristiques des données d'entrée (data-set) sur lesquels ils agissent. En somme, cette classification dépend de la présence - ou non - d'étiquettes dans le data-set qui représentent l'information que l'on veut prédire. Nous distinguons alors plusieurs catégories d'algorithmes de ML dont l'apprentissage supervisé (ici, les observations du jeu de données sont étiquetées par les valeurs que l'on cherche à prédire), l'apprentissage semi-supervisé (le data-set ici contient une proportion de données étiquetées et une autre proportion non étiquetée), l'apprentissage par renforcement (au cours duquel un agent reçoit en entrée l'état courant de son environnement et prend une décision sur l'action à exécuter en conséquence) et l'apprentissage non supervisé (opposé à la version supervisée, car les observations ne sont pas étiquetées). Le choix d'un algorithme de classification repose est influencé par divers éléments tel que la popularité, sa complexité temporelle et son adaptabilité aux gros volumes de données.

B. L'algorithme K-Means

La formulation du problème du K-Means, traduit en K-moyennes a été proposée la première fois par Hugo Steinhaus[12].

Définition 3. Soit K le nombre de clusters à obtenir. Il s'agit de trouver l'affectation des observations à K clusters de manière à minimiser la variance intra-cluster donnée par :

$$\arg \min_{C_1, C_2, \dots, C_K} \sum_{k=1}^K \|\vec{x} - \vec{\mu}_k\|_2^2$$

Où $\|\cdot\|_2$ représente la norme euclidienne, C_i représente le cluster i et le vecteur $\vec{\mu}_i$ représente le point central de ce cluster donné par l'équation (E1) suivante :

$$\vec{\mu}_i = \frac{1}{|C_i|} \sum_{\vec{x} \in C_i} \vec{x}$$

Définition 4. La distance Euclidienne entre deux points x et y de dimension d est définie comme suit :

$$d(x, y) = \|\vec{x} - \vec{y}\|_2 = \sqrt{\sum_{i=0}^d (x_i - y_i)^2}$$

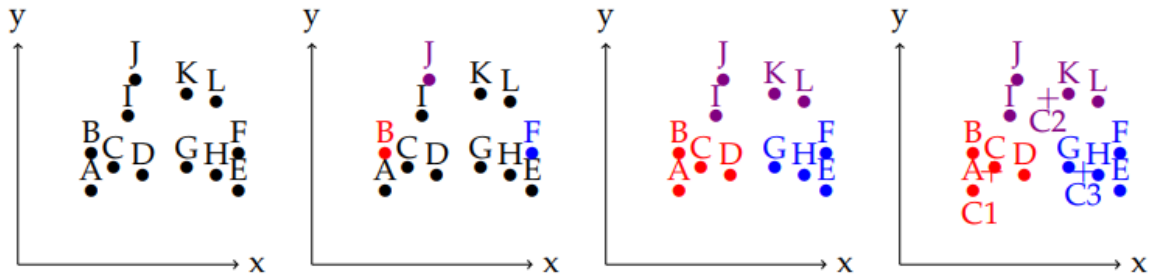
Algorithm : Pseudo Algorithme du K-means

Data: D : Data-set**Result:** K centres

```
1 Initialiser les K centres
2 while Non convergence do
3   for  $i \leftarrow 0$  to  $N - 1$  do
4     | Affecter l'observation  $i$  du data-set au centre le plus proche
5   | Recalculer les centres selon la formule (E1).
```

La première étape consiste à initialiser les centres (ligne 1). Dans la littérature, cela est fait soit de manière aléatoire en échantillonnant K observations du data-set pour les considérer comme centres initiaux, ou suivant une heuristique d'initialisation qui vise à maximiser la distance entre les centres initiaux. Par la suite, chaque observation du data-set est parcourue et affectée au centre le plus proche en terme de distance euclidienne (ligne 4). Une fois toutes les observations parcourues et affectées, les centres de clusters sont recalculés tel que chaque nouveau centre du cluster i est le barycentre des observations qui lui sont affectées selon l'équation (E1). Le processus s'arrête lorsque la condition de convergence est atteinte. Cette dernière est caractérisée par le fait que les centres ne varient plus entre deux itérations consécutives, ou que le nombre d'itérations atteint un nombre maximal it_{max} .

La figure ci-dessous illustre le fonctionnement de l'algorithme K-Means pour la classification d'un jeu de données en trois clusters.



C. Limites et complexité de K-Means

La principale difficulté dans l'utilisation de l'algorithme de clustering K-Means réside dans la définition en amont du nombre K de partitions. Et même s'il existe des méthodes complémentaires pour optimiser sur certains jeux de données, elles restent faillibles ; les jeux de données avec une forme elliptique dans un espace 2D poseront par exemple problème à classifier.

La complexité temporelle de l'algorithme K-Means est donné par : $\Theta(\text{K-Means}) = \Theta(n * t * k * d)$, avec n la taille du jeu de données, t le nombre d'itération maximale, k le nombre de clusters et d la dimension des données[7].