



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

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : *Génie Logiciel*

Par :

AROUSI Abdellah Mohamed

Sur le thème

Les Systèmes Multi-Robots coopératifs: comparaison entre les stratégies de coopération

Soutenu publiquement en juillet 2019 à Tiaret devant le jury composé de :

Mr. CHADLI Abdelhafid	MCB Université de Tiaret	Président
Mr HATTAB Noureddine	MAA Université Tiaret	Examineur
Mr MOKHTARI Ahmed	MAA Université Tiaret	Encadreur

Remerciement

*je remercie ALLAH qui m'a accordé la volonté et le courage pour
achever ce travail.*

*J'exprime mes remerciements et ma gratitude à mon encadreur
MOKHTARI Ahmed pour son soutien, sa patience, son honnêteté et sa
disponibilité à tout moment*

*je tiens à remercier Monsieur CHADLI Abdelhafid, d'avoir
accepté, de présider et d'honorer avec sa présence ma soutenance.*

*je remercie également Monsieur HATTAB Nouredine, pour l'intérêt
qu'il a porté au travail effectué en acceptant de participer à ma
soutenance en tant qu'examineur.*

*Enfin, je tiens à exprimer ma profonde gratitude à l'ensemble
des enseignants et du corps administratif qui ont contribué tout au long de
mon cursus.*

*Et à tous ceux qui ont contribué de près ou de loin à la réalisation de ce
travail.*

Dédicaces

A mon père

En toi mon père ; je vois un père dévoué à sa famille. Ta présence en toute

Circonstance m'a maintes fois rappelé les sens de la responsabilité.

A ma mère

En toi ; je vois la maman parfaite ; toujours prête à se sacrifier pour le bonheur

De ses enfants.

Merci pour tout.

A mes frères sid ahmed , mohamed , djamel , jimmi , nassreddine et sœurs
bouchra , zhour , chahra et mes amis houari , yacin , sofian je vous dédie ce

travail

A kablia fatiha qui m'a toujours soutenu et

m'encourager...

A ma famille.

ilyess.

Résumé

les system multi robot représente un système complexe dans sa réalisation nécessite la coopération de plusieurs robots . il existe plusieurs stratégies de coopération ,l'objectif de Ce travail est de comparer ses différents stratégies pour cela on a créer un système de simulation de robot de déminage est tester ce système avec les différents stratégies , D'après les résultats on a constater qu'il n'existe pas une stratégies universel mais les performances de la stratégie dépend des paramètres de système : le nombre de robot , l'étendu de l'environnement .

Mots clés : Système Multi-Agents, Système Multi-robots, coopération ,Simbad.

Abstract

The multi-robot system represents a complex system in its realization requires the cooperation of several robots. There are several strategies of cooperation, the objective of this work is to compare its different strategies for that we create a demining robot simulation system for testing this system with different strategies, according to the results we have found that there is no universal strategy but the performance of the strategy depends on the system parameters: the number of robots, the extent of the environment.

Keywords: Multi-Agents Systems, Multi-Robots System, Simbad.

	Pages
Chapitre 01: Les systèmes multi Agents	
Agent.....	01
Mesures de performances et rationalités	02
L'objectif chez un agent.....	04
La communication chez un agent.....	05
Cycle de vie d'un agent.....	06
Catégories d'un agent.....	08
Architecture d'un agent.....	10
Les systèmes multi agent.....	10
Le rôle des systèmes multi agent.....	11
Caractéristiques des systèmes multi agent	12
L'environnement des systèmes multi agent.....	13
Composition d'un SMA.....	15
L'interaction entre les agents.....	17
Schémas d'interaction.....	18
Langage de communication entre agents.....	18
Protocole de communication.....	19
Organisation.....	19
Application des SMA.....	19
Chapitre II Les systèmes multi robots	
Introduction	22
La robotique.....	23
Les types de robots.....	24
Les classes robots.....	25
Les composants des robots.....	25
Architecture des robots.....	29
Navigation des robots.....	30
Les systèmes multi-robots.....	30
Pourquoi un système multi robots.....	31
Les domaines d'utilisation de système multi robots.....	32
Catégories des systèmes multi- robots.....	33
Les problèmes de coordination dans SMR.....	33
Coopération des robots mobiles	34
La relation robot-tache-environnement	36
Les taches coopératives multi-robots.....	36
Les stratégies de coopération.....	37

L'architecture centralisée	38
L'architecture décentralisée.....	40
Architecture centralisée versus décentralisée	42
Chapitre 03 : conception	
Introduction	45
Navigation des robots	45
Descriptions de l'environnement.....	47
Présentation des stratégies.....	47
Présentation des techniques	53
Conclusion	
Chapitre 04 : implémentation	
Présentation des technologies de développement	54
Architectures générale de notre système.....	56
Présentation des interfaces.....	57
Simulation centralisée.....	57
Simulation décentralisée.....	58
Simulation hybride	59
Les résultats de simulation.....	59
Conclusion	

Introduction Général

La principale ligne actuelle de recherche pour la robotique mobile est de permettre à un système robotique mobile de se déplacer d'une manière autonome dans son environnement pour accomplir un certain nombre de tâches, ces tâches sont par exemple de se déplacer vers une cible fixée ou mobile, d'éviter les obstacles, de suivre une route.

Lors de notre étude nous avons identifié un ensemble de problèmes qui sont parfois complexes et elles nécessitent une grande coopération entre des groupes de robots mobiles.

Objectifs :

Dans notre travail, nous avons fixé un ensemble d'objectifs à atteindre afin de répondre à nos besoins fixés au préalable, nous citons par la suite nos objectifs qui sont :

- comparaison entre les stratégies de coopération des systèmes multi robots.
- appliquer une étude approfondie sur le secteur militaire (le déménagement des robots) .

Organisation du mémoire

Notre mémoire est organisée sous forme de quatre sections qui sont :

Chapitre 1 : Les systèmes multi -agents.

Ce chapitre est consacré à décrire les concepts généraux sur les systèmes multi agents.

Chapitre 02 : Les systèmes multi-robots.

Nous abordons dans ce chapitre un ensemble de concepts clés sur les systèmes multi-robots et la coopération entre eux.

Chapitre 03 : Conception.

A travers ce chapitre, nous avons clarifié et détaillé les stratégies de coopération des systèmes multi-robot ainsi nous avons modélisé les stratégies par des différents diagrammes UML qui montrent le fonctionnement réel.

Chapitre 04 : Implémentation.

Ce chapitre, est consacré pour montrer l'ensemble des technologies utilisées pour la réalisation de notre système, que nous avons illustré à travers des prises de captures d'écran qui montrent notre système. Finalement nous présentons un ensemble de graphes de comparaison qui ont pour but de choisir la meilleure stratégie.

Chapitre I
Les Systèmes Multi-Agents

I. Les systèmes multi-agents

Introduction

Le thème des systèmes multi-agents (SMA), s'il n'est pas récent, est actuellement un champ de recherche très actif. Cette discipline est à la connexion de plusieurs domaines en particulier de l'intelligence artificielle, des systèmes informatique distribués et du génie logiciel. C'est une discipline qui s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités autonomes et flexibles appelées agents, que ces interactions tournent autour de la coopération, de la concurrence ou de la coexistence entre ces agents.

Ce chapitre commence par définir ce qu'est un agent, ses différents types et son architecture, puis s'intéresse aux systèmes multi-agents(SMA), leurs propriétés pour aborder par la suite les différentes interactions et la communication entre les agents.

À la fin, un aperçu des différents domaines d'applications des SMA est donné.

Agent

Le concept d'agent est apparu avec les premiers programmes destinés à s'exécuter sur le long terme, au sein d'un environnement dans lequel ils jouissent d'une certaine autonomie d'action.

On ne peut attribuer une définition unique à la notion d'agent, car le terme agent est utilisé dans de très différentes applications par des communautés venant d'horizons divers.

Cette dernière a, depuis son apparition, été enrichie et affinée, notamment par *Wooldridge et Jennings 1995*, *Ferber1999* ainsi que d'autres mais ils ne se sont pas mis d'accord sur une seule description. Un consensus s'est établi autour d'une liste de propriétés et d'un schéma d'exécution présentés ci-après.

Définition

On appelle agent une entité physique ou virtuelle :

- [Ferber95] :
Un agent est une entité réelle ou virtuelle, évoluant dans un environnement, capable de le percevoir et d'agir dessus, qui peut communiquer avec d'autres agents, qui exhibe un comportement autonome, lequel peut être vu comme la conséquence de ses connaissances, de ses interactions avec d'autres agents et des buts qu'il poursuit.
- [Demazeau96] :
Un agent est une entité réelle ou virtuelle, dont le comportement est autonome, évoluant dans un environnement, qu'il est capable de percevoir, sur lequel il est capable d'agir et d'interagir avec les autres agents.

- [Woolridge 98] :

Un agent est un système informatique capable d'agir de manière autonome et flexible dans un environnement. Flexibilité signifie réactivité, pro-activité, adaptabilité et capacités sociales.

- a. Un agent est un système informatique, situé dans un environnement, et qui agit d'une façon autonome pour atteindre les objectifs (buts) pour lesquels il a été conçu (Wooldrige et Jennings, 1995).
- b. Les agents intelligents sont des entités logicielles qui réalisent des opérations à la place d'un utilisateur ou d'un autre programme, avec une sorte d'indépendance ou d'autonomie, et pour faire cela ils utilisent une sorte de connaissance ou de représentation des buts ou des désires de l'utilisateur." (L'agent IBM)
- c. Un agent est une entité qui fonctionne continuellement et de manière autonome dans un environnement où d'autres processus se déroulent et d'autres agents existent." (Shoham, 1993);
- d. Un agent est une entité autonome, réelle ou abstraite, qui est capable d'agir sur elle-même et sur son environnement, qui, dans un univers multi-agents, peut communiquer avec d'autres agents, et dont le comportement est la conséquence de ses observations, de ses connaissances et des interactions avec les autres agents (Ferber, 1995).
- e. qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser).
- f. qui possède des ressources propres.
- g. qui est capable de percevoir (mais de manière limitée) son environnement,
- h. qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune).
- i. qui possède des compétences et offre des services.
- j. qui peut éventuellement se reproduire.
- k. dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.

Chacun des termes de cette définition est important. Une entité physique est quelque chose qui agit dans le monde réel: un robot, un avion ou une voiture sont des exemples d'entités physiques. En revanche, un composant logiciel, un module informatique sont des *entités virtuelles*, car elles n'existent pas physiquement.

La propriété la plus importante d'un agent est son autonomie. Celle-ci se rapporte au contrôle de son exécution : un agent peut dire "non" à une requête, il décide lui-même d'agir ou non, et de la nature de ses actions. C'est d'ailleurs ce qui le différencie de tous les concepts semblables : "objets", "modules logiciels", "processus" ou d'un simple sous-programme.

Cette autonomie sous-entend qu'un agent a des buts qui lui sont propres, et qu'il peut les privilégier à la résolution d'une requête extérieure.

Mais l'autonomie n'est pas seulement comportementale, elle porte aussi sur les ressources. Pour agir, l'agent a besoin d'un certain nombre de ressources: énergie, CPU, quantité de mémoire, accès à certaines sources d'informations, etc. Ces ressources sont à la fois ce qui rend l'agent non seulement dépendant de son environnement, mais aussi, en étant capable de gérer ces ressources, ce qui lui donne une certaine indépendance vis-à-vis de lui.

L'agent est ainsi à la fois un *système ouvert* (il a besoin d'éléments qui lui sont extérieurs pour survivre) et un *système fermé* (car les échanges qu'il a avec l'extérieur sont très étroitement règlementés).

Les agents n'ont qu'une représentation partielle de leur environnement, c'est-à-dire qu'ils n'ont pas de vision globale de tout ce qui se passe. C'est d'ailleurs ce qui se passe dans les réalisations humaines d'envergure (la fabrication d'un Airbus par exemple) dans lesquelles personne ne connaît tous les détails de la réalisation, chaque spécialiste n'ayant qu'une vue partielle correspondant à son domaine de compétence.

Mesure de performance et rationalité chez un agent

Un agent rationnel est un agent capable de toujours effectuer la ou les actions appropriées pour chaque cas. Le fait d'analyser si une action est appropriée ou non nous amène à la performance d'un agent, et de la mesure de cette performance.

a. Mesure de performance :

Dans son environnement, l'agent reçoit des percepts et peut donc générer une séquence d'actions. Cette séquence d'actions définit en quelque sorte son comportement pour la tâche à effectuer. Et une manière simple de définir le succès d'un comportement est de voir si l'environnement est bien dans l'état recherché, ou passe bien par les états voulus, après l'application du comportement par l'agent. On en déduirait donc le bienfondé ou non du comportement en fonction de ces différents états de l'environnement.

b. Rationalité:

Plusieurs facteurs permettent de déterminer la rationalité à un instant donné :

- L'ensemble des connaissances (acquises ou innées) dont l'agent dispose sur son environnement.
- L'ensemble des actions qu'il peut effectuer dans cet environnement.
- La séquence de percepts à l'instant donné.
- La mesure de performance définissant le succès ou non d'une tâche.

Suite à l'énumération des quatre facteurs permettant de déterminer ce qui est rationnel ou non, nous pouvons clairement définir ce qu'est un agent rationnel tel que S. Russell et P. Norvig l'ont proposé : « Pour chaque séquence possible de percepts, un agent rationnel doit sélectionner une action susceptible de maximiser sa mesure de performance, compte tenu des observations fournies par la séquence de percepts et de la connaissance dont il dispose. » (Russell et Norvig 2006).

L'objectif chez un agent

Chez un agent, l'objectif est ce qui le pousse à agir. Nous pouvons nous demander ce que ferait un agent sans fonction, sans objectif à satisfaire. Il est possible qu'il ne fasse rien, il est possible qu'il agisse de manière chaotique ou incohérente, mais son comportement ne paraîtrait en rien rationnel pour l'environnement dans lequel il se trouve du fait de son absence de raison d'être.

La communication chez un agent

Chez les agents, la communication est une forme d'interaction. Cette possibilité de communiquer permet l'échange d'informations, et donc de connaissances. La communication rend également possible la coopération entre agents, par exemple quand plusieurs agents ont des objectifs complémentaires. Bien entendu, pour que la communication soit effective chez les agents, il faut qu'ils aient un protocole commun de communication ainsi qu'un canal commun et ce, afin d'assurer la cohérence dans les échanges d'informations.

Cycle de vie d'un agent

Le comportement d'un agent résulte de ses perceptions, de ses connaissances, de ses compétences, et bien évidemment de son but. Il suit un cycle de vie en trois étapes se répétant indéfiniment tout au long de son exécution :

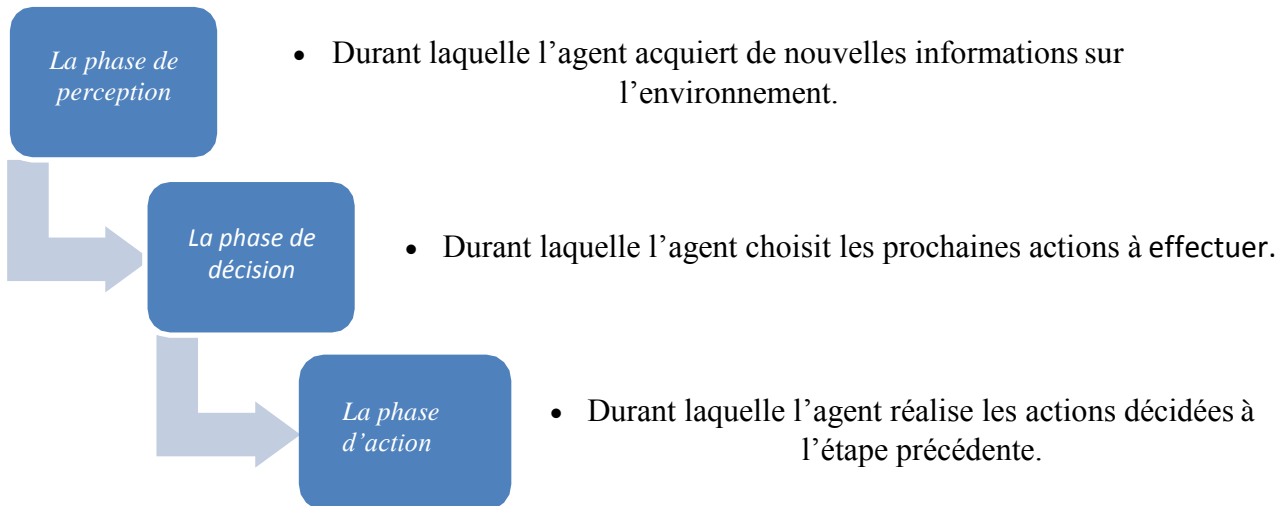


Figure I :Cycle de vie d'un agent

L'agent est ainsi une sorte "d'organisme vivant" dont le comportement, qui se résume à communiquer, à agir et, éventuellement, à se reproduire, vise à la satisfaction de ses besoins et de ses objectifs à partir de tous les autres éléments (perceptions, connaissances, représentations, actions, communications et ressources) dont il dispose.

Enfin, notons que cette définition s'appuie fortement sur la notion d'environnement, qui, quand elle ne bénéficie pas d'un consensus dans la communauté. Intuitivement, l'environnement peut être décrit comme tout ce qui est extérieur à l'agent et qu'il peut percevoir, il sera plus détaillé ci-après.

Catégories d'agents

Les agents sont reparties en deux grandes catégories : les agents réactifs, et les agents cognitifs. La différence entre ces deux catégories d'agents concerne la représentation que peut avoir un agent de son environnement.

A- Agents réactifs

La catégorie des agents réactifs, décrite par R. A. Brooks, comprend tous les agents dont la représentation de l'environnement n'est que sub-symbolique (Brooks, 1991). Cela signifie que la représentation provient uniquement des perceptions de l'agent, du monde « visible » à l'instant courant. Il n'y a pas de raisonnement à proprement parler dans cette catégorie d'agents puisque l'on est dans une configuration du type stimulus/action : stimulus--> percept--> action. L'agent réagit aux évènements dans l'environnement mais n'a pas de mémoire et ne peut donc ni prendre le passé en compte, ni prévoir au-delà du court terme.

Le principe d'un agent réactif permet la construction de systèmes composés de nombreux petits agents, qui sont des automates. Les interactions des agents entre eux permettent l'émergence de structures d'une couche d'abstraction supérieure qui sont potentiellement observables (Lestel et al. 1994).

- Pas de représentation explicite
- architectures simples
- stimulus -> réponse
- Organisation implicite/induite
- auto-organisation, ex : colonie de fourmis
- Communication via l'environnement

ex : perception/actions sur l'environnement, phéromones de fourmis

- Grand ou très grand nombre d'agents
- Redondance
- Robustesse
- Emergence
- Validation expérimentale

B- Agents cognitifs

Contrairement aux agents réactifs, la catégorie des agents cognitifs comprend tous les agents disposant d'une représentation symbolique de leur environnement. Les agents cognitifs sont capables de raisonner sur leur représentation symbolique. Il existe des algorithmes pour manier les symboles, et donc les représentations symboliques, mais ils sont très complexes et limitent les capacités de ce type d'agents. Très souvent, cette classe d'agents s'inspire de théories de la cognition, principalement humaine. Ce sont ces différentes théories de la cognition qui induisent les comportements des agents.

- Représentation explicite
- Architectures complexes, souvent modèle logique (ex : BDI, Agent0)
- Organisation explicite
- allocation et dépendances tâches
- partage des ressources
- protocoles de coordination/négociation
- Communication explicite, point à point, élaborée (ex : KQML)
- Petit/moyen nombre d'agents
- Certaines validations formelles possibles

Architecture d'un agent

Les agents possèdent généralement une architecture modulaire bien déterminée. Les modules utilisés varient selon les applications, mais trois se retrouvent systématiquement :

1. **Le module perception** prend en compte les événements provenant de l'environnement. Il peut s'agir d'une simple boîte à lettre dans le cas d'un agent communicant, ou d'un module gérant des capteurs pour un agent situé.
2. **Le module décision** implémente le raisonnement de l'agent. Il comprend les connaissances de l'agent et des règles pour les exploiter, et décider d'action à entreprendre en fonction du but de l'agent. Il peut s'agir de simples règles conditionnelles pour un agent réactif comme d'algorithmes d'apprentissage et d'un moteur d'inférence pour un agent cognitif.
3. **Le module action** réalise les actions que le module décision lui demande. Il peut s'agir de simples envois de messages dans le cas d'agents communicants comme d'actions plus complexes mettant en œuvre des effecteurs. C'est ce qui est vu depuis l'extérieur comme le comportement de l'agent.

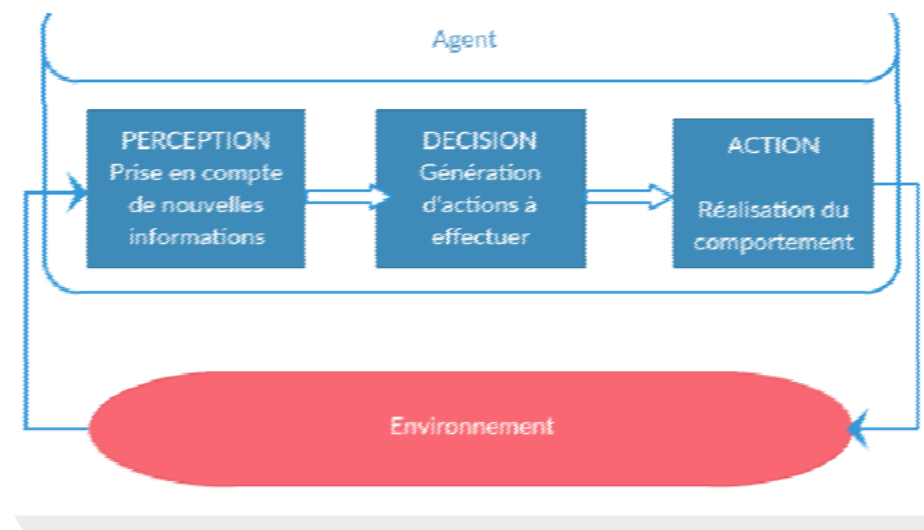


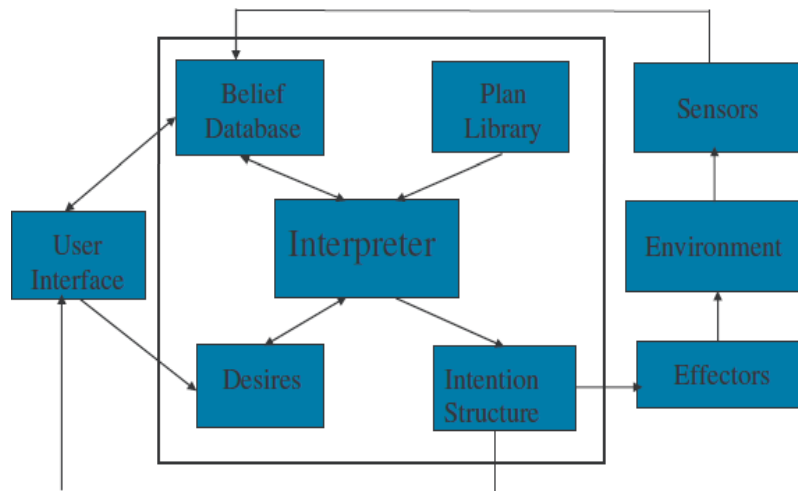
Figure 02: Un agent et son environnement

La figure I.1 schématise ces modules et leur interaction avec l'environnement. Cette architecture de base est souvent enrichie de modules propres à l'approche, par exemple d'un module spécifique pour les connaissances, les compétences ou encore l'apprentissage. De nombreuses implémentations sont en fait hybrides, possédant des caractéristiques à la fois réactives et cognitives et des modules d'interaction à la fois situés et communicants. Cette modularité fait que la programmation par composants est particulièrement appropriée au développement d'agents, et donc de SMA.

La notion d'agent étant maintenant bien définie, il est temps de s'intéresser aux systèmes multi-agents.

Architecture

- Les agents logiques : fonctionnement basé sur des déductions logiques.
- Les agents réactifs : fonctionnement basé sur une simple correspondance entre les situations et les actions (interagissent avec leur environnement mais sans raisonner dessus).
- Les agents BDI : l'agent décide des actions à entreprendre à partir de ses états internes qui sont exprimés sous la forme de croyances (Belief), de désirs (Désire) et d'intentions (Intention).



BDI agent architecture (selon Kiny).

Figure III Architecture BDI

- Les agents multi-niveaux : les connaissances internes de ce type d'agents sont organisées en différents niveaux d'abstractions, permettant ainsi différents niveaux de traitement.

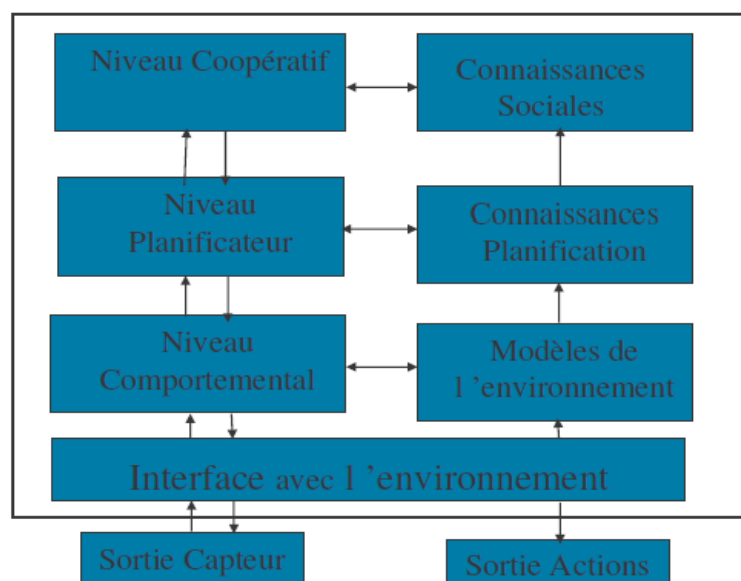


Figure4 Archetecture multi -niveau

Systemes multi-Agents

La motivation du domaine des SMA est de réaliser des entités artificielles (des programmes, des robots) qui interagissent entre elles et avec leur environnement. Une attention particulière est alors portée sur l'aspect multi-agents. Le passage de l'individu (l'agent) au collectif (le système multi-agent) a nécessité l'extension et la généralisation de divers concepts issus de l'intelligence artificielle tels que l'apprentissage multi-agent, le raisonnement sur les connaissances de groupes ou encore la décision collective.

Définition d'un SMA

« Un Système Multi-Agents (SMA) comporte plusieurs agents qui interagissent entre eux dans un environnement commun. Certains de ces agents peuvent être des personnes ou leurs représentants (avatars), ou même des machines mécaniques. S'il y a moins de trois agents, on parle plutôt d'interaction homme/machine, ou machine/machine que de systèmes multi-agents. » Anne Nicole.

« On appelle système multi-agent (ou SMA), un système composé des éléments suivants :

Un environnement E , c'est à dire un espace disposant généralement d'une métrique.

Un ensemble d'objets.

Ces objets sont situés, c'est à dire que, pour tout objet, il est possible à un moment donné, d'associer une position dans E .

Ces objets sont passifs, c'est à dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.

Un ensemble A d'agents, qui représentent les entités actives du système.

Un ensemble de relations R qui unissent les objets et les agents entre eux.

Un ensemble d'opérations Op permettant aux agents A de percevoir, produire, consommer, transformer et manipuler les objets de O .

Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers. »Ferber.

Rôles des SMA

• Résoudre un problème de manière distribuée : systèmes multi-experts.

- Les actions des agents sont des transformations d'objets liées à la description d'un problème.

- agents plutôt rationnels

- Simulation de phénomènes complexes.

-Les agents simulent des actions physiques, biologiques ou sociales qui produisent des modifications du monde représenté. Ex : simulation de la pêche dans le delta du Niger, des épidémies, écosystèmes (proies / prédateurs).

- Agents plutôt réactifs.

- Gérer et maintenir un environnement de travail.

- Les actions physiques ou sociales réalisées par les agents sont des actions réelles, elles évoluent dans le temps et modifient le monde : robots footballeurs, agents négociant un rendez-vous au profil de l'utilisateur.

- Agents plutôt cognitifs et sociaux.

Caractéristiques des SMA

Chaque agent a des informations ou des capacités de résolution de problèmes limitées, ainsi chaque agent a un point de vue partiel ;

-Il n'y a aucun contrôle global du SMA ;

-Les données sont décentralisées ;

-Le calcul est asynchrone.

-L'approche SMA ne s'intéresse pas à la recherche de solutions optimales. Elle recherche des solutions satisfaisantes pour des problèmes complexes . [Mr CHADLI -c1]

Pour concevoir un SMA, il faut définir:

- Un modèle de SMA

- Le modèle de chacun des agents qui vont entrer en action (niveau microscopique);

- Définir leur environnement et leurs interactions (niveau macroscopique) ;

- Définir les organisations sociales (niveau macro) qui les structurent.

- Un modèle concret de SMA

- Qui crée, initialise les agents,

- Installe leur organisation et

- Lance les agents qui doivent intervenir pour une exécution particulière.

Environnement d'un SMA

L'environnement est un concept très important lorsque l'on parle de systèmes multi-agents. C'est de lui que le système tire de l'information, et c'est lui qui supporte l'activité des agents. Il n'existe cependant pas de définition qui fasse consensus

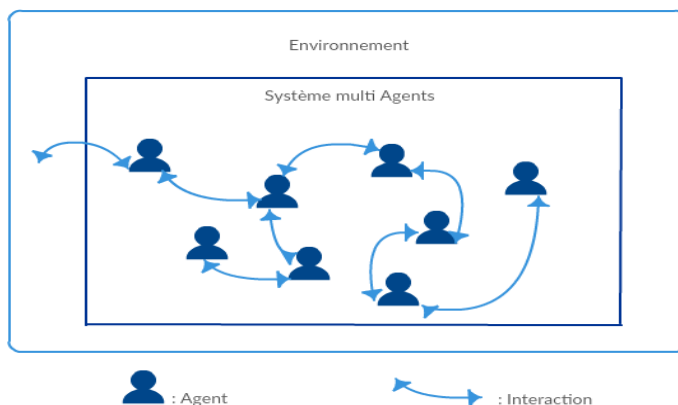


Figure 5: Un SMA et son environnement

Différents environnements peuvent être identifiés selon le point de vue adopté. comprendre tout ce qui lui est extérieur. De même, l'environnement d'un agent comprend tout ce qui lui est extérieur, mais cela inclut les autres agents du système. Revenant à notre exemple, l'environnement d'un joueur est celui de l'équipe, plus tous ses coéquipiers. Dans un SMA hétérogène, tous les agents n'ont pas les mêmes capacités de perception. Ainsi, il est possible que seule une partie des agents d'un SMA interagissent avec l'environnement du SMA, les autres étant internes au système, comme c'est le cas pour la figure ci-dessus.

Les interactions entre un SMA et son environnement sont couplées. Lorsque le SMA effectue une action sur l'environnement, celui-ci est modifié. La perception de cette modification agit comme un signal de retour sur le SMA. Notons que l'agent qui effectue l'action et celui qui perçoit la modification ne sont pas forcément les mêmes.

On peut distinguer plusieurs facettes de l'environnement d'un agent. Il y a d'un côté l'environnement physique, composé de ce que les capteurs des agents peuvent percevoir, et de l'autre côté, l'environnement social, composé des autres agents connus. Dans l'exemple de la simulation d'une équipe de basketball, l'environnement physique est par exemple composé des paniers et de la balle, et l'environnement social des autres joueurs et du coach.

Enfin, il existe une variété d'environnements aussi vaste que les applications le permettent.

L'environnement peut être *continu* ou *discret*, *épisode* ou pas, *déterministe* ou non, *statique* ou *dynamique* [Russell et Norvig 2010]. Autrement dit, il n'y a pas de restriction sur la nature de l'environnement pour y plonger un SMA, autre que celle de l'existence de moyens de perception et d'action.

La création d'un SMA demande donc de définir son environnement et de fournir un moyen d'interaction avec celui-ci. D'autres éléments sont nécessaires et sont décrits dans ce qui suit.

Composition d'un SMA

Les agents et l'environnement sont les composantes les plus visibles d'un SMA, mais d'autres sont tout aussi cruciales. [Demazeau 1995] propose par exemple une vue des SMA en quatre composantes élémentaires :

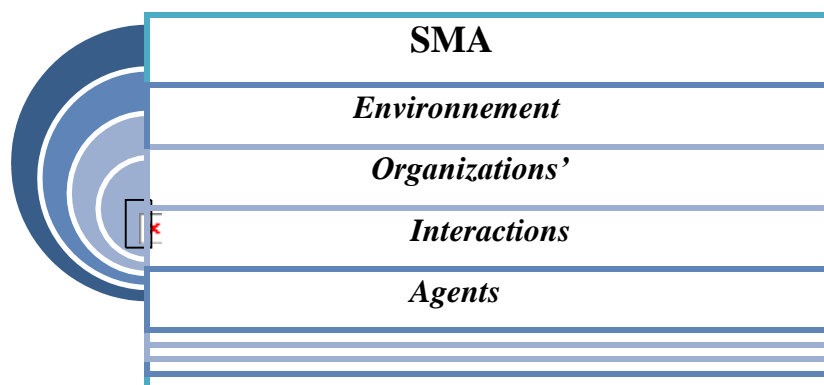


Figure 6 : Composition d'un SMA

- **Les agents** incluant la description complète de leur architecture et de leurs comportements.
- **L'environnement** incluant la description des ressources avec lesquelles peuvent éventuellement interagir les agents.
- **Les interactions** incluant l'ensemble des moyens assurant la transmission d'informations entre les agents, en particulier les protocoles de communication et les langages utilisés. Nous

avons vu que les agents pouvaient communiquer soit par le biais de modifications sur l'environnement, soit directement par envoi de messages. La spécification de ces communications va avoir un impact important sur le comportement global du système.

- **Les organisations** incluant les éléments permettant de structurer un ensemble d'agents, par exemple sous forme de hiérarchie ou d'un simple réseau de relations.

L'interaction entre les agents

« Une interaction est la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques... Les interactions sont non seulement la conséquence d'actions effectuées par plusieurs agents en même temps, mais aussi l'élément nécessaire à la constitution d'organisation sociales » (Ferber).

- Interagir pour gérer, communiquer, se coordonner, coopérer, négocier...
- Interaction passive par modification de l'environnement ;
- Interaction intentionnelle par action visible sur l'environnement ou par l'intermédiaire de messages que les agents s'envoient les uns aux autres.
- Nécessité de modèles formels d'interactions.

Différentes formes :

- Interaction directe :
Un agent communique par envoi de messages asynchrone vers un autre agent ou ensemble d'agents
- Interaction indirecte :
La communication est réalisée au travers de l'environnement

Interaction et coopération entre agents

Un système multi-agent (SMA) se distingue d'un groupe d'agents indépendants par le fait que les agents interagissent afin de réaliser collectivement une tâche ou d'atteindre ensemble un but particulier.

Ainsi un agent peut avoir connaissance du fait qu'il n'est pas seul dans son environnement, et qu'il peut interagir avec d'autres entités. Ce qui nous mène à la notion d'interaction qui est définie comme suit: « Une *interaction* est une mise en relation dynamique de deux ou plusieurs agents par le biais d'actions réciproques» [FERB 95].

Cette interaction peut être coopérative ou pas. Nous pouvons caractériser le système par le type de coordination mis en œuvre qui peut aller de coopération nulle (*antagonisme* totale) vers une coopération totale.

Aussi, des agents coopératifs peuvent changer leurs buts pour répondre aux besoins d'autres agents afin d'assurer une meilleure coordination entre eux. Cela peut entraîner des coûts de communications élevés. Les agents antagonistes ne vont -par contre- pas coopérer et dans ce cas, leurs buts individuels risquent d'être bloqués et les coûts de communication sont bien évidemment minimaux.

Chaque agent peut être caractérisé par trois dimensions : ses buts, ses capacités à réaliser certaines tâches et les ressources dont il dispose.

Les interactions des agents d'un SMA sont motivées par l'interdépendance des agents selon ces trois dimensions suivantes:

- Leurs buts peuvent être compatibles ou non.
- Les agents peuvent désirer des ressources que les autres possèdent.
- Un agent peut disposer d'une capacité nécessaire à un autre agent pour l'accomplissement d'un des plans d'action de ce dernier.

La manière avec laquelle s'accomplissent les interactions permet d'obtenir des situations variées au niveau du SMA, la figure qui suit illustre l'interaction d'un agent avec d'autres agents ainsi qu'avec son environnement.

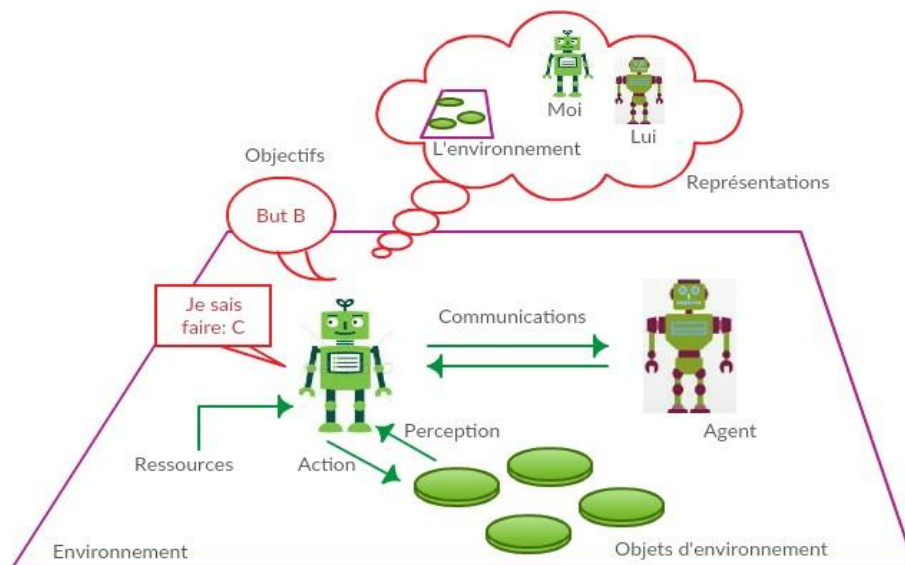


Figure 7: Représentation d'un agent en interaction avec son Environnement et les autres agents

Concrètement, l'ensemble de ces situations sont gérées au sein du SMA par des mécanismes de *coordination*. Nous trouverons ci-dessous sa définition dans la littérature :

«La coordination est le processus de construction de programmes en assemblant els parties actives. Un modèle de coordination est la colle qui lie des activités séparées en un ensemble, c'est aussi la gestion des dépendances entre les activités»

La coordination est fortement liée aux relations qui peuvent s'établir entre les agents et plus généralement à la façon dont les agents sont organisés.

Plusieurs schémas d'interactions sont nécessaires

Les principaux sont :



la collaboration :

Manière de répartir le travail entre plusieurs agents



la coordination d'actions

Manière dont les actions des différents agents doivent être organisés dans le temps et l'espace



Les coopérations

- forme générale d'interaction pour les agents capables d'avoir un projet explicite
- Qui fait quoi, quand, ou, et avec quels moyens
- Les techniques de négociation sont utilisées pour limiter les effets des conflits qui apparaissent

Coopération = collaboration + coordination + résolution de conflits

Langages de Communication entre agents (ACL)

Plusieurs formalismes d'ACL ont été proposés depuis 1990 mais les plus importants sont :

- KQML (Knowledge Query & Manipulation Language): standard de Tom Finin et al. (1993).

- ACL-FIPA nouveau standard de la FIPA (Fondation for Intelligent Physical Agents) en 1999.

Dans les ACL, le contenu du message n'est pas spécifié en termes ACL mais dans un formalisme laissé au choix du développeur ; c'est pourquoi, le langage de codage des messages et le choix d'ontologies pour la représentation des données est fournie dans l'entête du message (attributs language et ontology).

- KIF : Knowledge Interchange Format (Langage formel de dialogue interne)

Protocoles de communications:

Des normes de communications fixant les règles d'interactions peuvent être utilisées, par exemple :

Achieve-RE : un Initiateur envoie un message, le receveur peut répondre par not-understood, refuse ou agréé . Suite a l'accord (agréé), le receveur retourne un message de type inform (réponse) ou failure.

FIPA - Contrat NET : Protocole d'établissement d'un contrat (de vente par exemple).

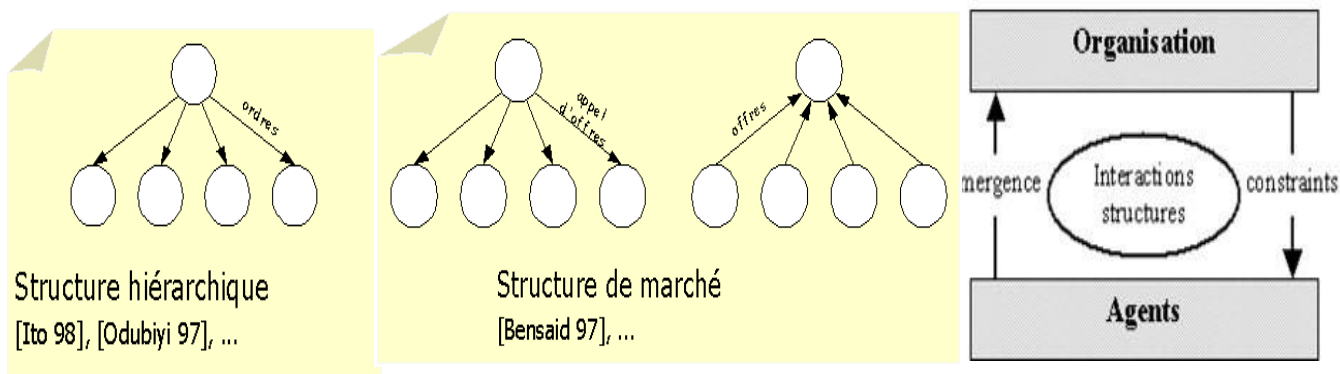
FIPA - Propose : un Initiateur propose à Participant d'effectuer une action .

FIPA - Subscribe : un Initiateur envoie un message à un Participant de souscription de service.

Organisations

Les SMA peuvent être :

- organisés a priori : en structure hiérarchique, de marche, de communauté, de société [Grislin95], [Mandiau99]
- organisés par émergence : la structure de l'organisation apparaît suite aux interactions entre agents



Application des SMA

Les applications des systèmes multi-agents couvrent de plus en plus de domaines.

Citons les systèmes d'information coopératifs, la simulation sociologique, les outils documentaires adaptés au Web, les robots autonomes coopératifs, jeux vidéo (multi-joueurs), résolution distribuée de problèmes, etc. Néanmoins les systèmes multi-agents développés actuellement peuvent être classés en trois catégories [Gleizes 04], [Ferber 06], [Rejeb 05] comme suit

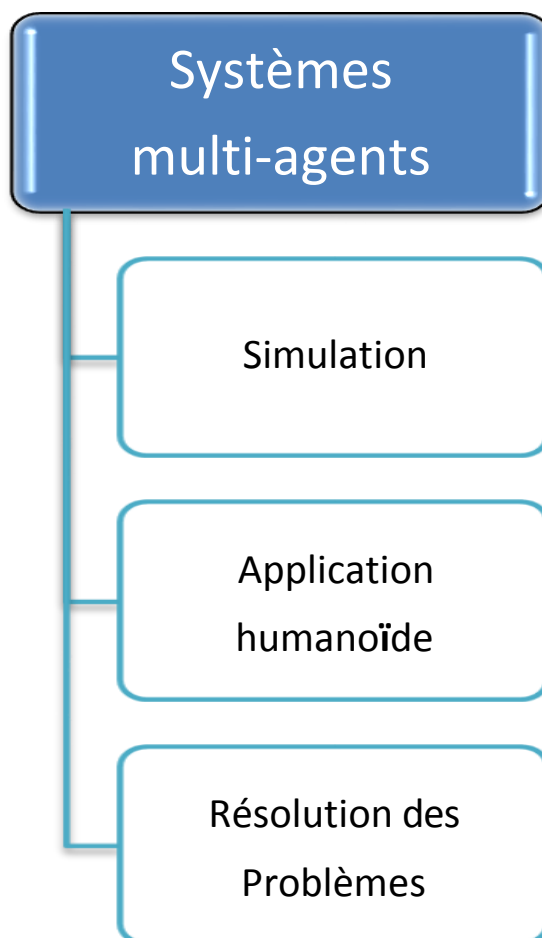


Figure 8 : Applications des SMA

-
- **Les simulations:** dont l'objectif est la modélisation de phénomènes du monde réel, afin d'observer, de comprendre et d'expliquer leur comportement et leur évolution. Les systèmes multi-agents ont trouvé rapidement un champ extrêmement propice à leur développement dans le domaine de la modélisation de systèmes complexes ne trouvant pas de formalisation mathématique adaptée. Dans le domaine des sciences du vivant d'abord, ensuite dans celui des sciences humaines et sociales, les SMA ont montré qu'il était possible de modéliser au niveau micro les comportements d'entités élémentaires et d'étudier au niveau macro le résultat global de l'interaction de ces entités. En effet, la simulation multi-agents permet de tester rapidement le changement de certaines hypothèses ; elle permet aussi d'intégrer de nouveaux agents et d'éditer, sur un plan pratique, les résultats pour comparer les expérimentations les unes aux autres, de plus elle permet de préserver l'hétérogénéité du système à simuler.

 - **Les applications dans lesquelles les agents jouent le rôle d'êtres humains.** La notion d'agent simplifie la conception de ces systèmes et amène de nouvelles problématiques centrées utilisateur telles que la communication, la sécurité... etc. Comme par exemple les systèmes de ventes aux enchères dans laquelle les agents jouent les rôles de commissaire-priseur et d'acheteurs, représentent une classe d'applications de cette catégorie.

 - **La résolution de problèmes :** telle qu'elle avait été définie en Intelligence Artificielle, étendue à un contexte distribué. Dans ce cadre, l'objectif est de mettre en œuvre un ensemble de techniques pour que des agents, pertinents pour la résolution d'une partie ou l'ensemble du problème, participent de manière efficace et cohérente à la résolution du problème global.

Conclusion

Dans ce chapitre, nous avons essayé de définir le plus précisément possible, ce qu'un agent et ce qu'un système multi-agents tout en mettant l'accent sur les concepts qui s'y rapportent tel que celui de l'environnement, d'interaction et de communication.

Chapitre II

Les Systèmes Multi-Robots

Introduction:

Le robot doit son nom à Karel CAPEK. Dérivé de "robota", il signifie corvée ou travail forcé en tchèque et dans les langues slaves. Ce mot fut alors utilisé pour la première fois dans la pièce de théâtre RUR 1 en 1921 où le robot cherche à détrôner l'homme en prenant la place des ouvriers dans les usines.

Mais il a fallu attendre le début des années 1950, pour que les premiers robots mobiles équipés de capteurs voient le jour. Ainsi, William Grey Walter mit au point des tortues (appelées les Tortues de Bristol)

. Chacune a un œil photo électrique qui lui permet de détecter les sources de lumière. Des amplificateurs transmettent le signal reçu par la plaque photo électrique aux moteurs qui permettent à la tortue de s'approcher de la source de lumière.



Figure 9 : Les tortues de Bristol par William Grey Walter.

Le premier robot mobile à comporter un planificateur d'actions (et donc une notion d'intelligence) s'appelle Shakey et fut développé au Stanford Research Institute SRI à la fin des années 1960. Il fut équipé d'une caméra, d'un télémètre et connecté à deux ordinateurs via une liaison radio

En France, le premier robot mobile est né au LAAS-CNRS en 1977. Il se localisait dans une pièce en fusionnant les données de plusieurs types de capteurs (ultrasons, odométrie et télémètre) (cf. Figure 9).

Les avancées technologiques réalisées à la fin du siècle dernier et jusqu'à nos jours ont donné naissance à de nouveaux matériaux, des ordinateurs plus rapides, plus petits et moins chers, etc. Ceci a propulsé la recherche en robotique mobile. Ainsi, les robots se déplacent maintenant sur le sol, dans l'air, sous l'eau et même dans l'espace

La robotique :

Quand les chercheurs ont commencé à réfléchir sur la conception de robots, on appelait « robotique » la science des robots et/ou l'art de concevoir et fabriquer des robots. Les succès des robots industriels, qui travaillaient initialement en poste isolé et qui ont été intégrés à des chaînes de production comme des machines parmi d'autres, ont conduit le public à élargir le sens du terme « robotique » et à le rendre désormais presque synonyme d'automatisation. Le correspondant de robotique devrait naturellement être « automatique » mais ce n'est pas le cas dans le langage courant. Pour compenser ce hiatus se sont créés le mot « robotisation » et le verbe « robotiser » pour désigner la plupart des automatisations et leur mise en œuvre, même en l'absence de véritables robots. Dans certains champs d'applications, des mots calqués sur robotique sont apparus. Il en est ainsi de « domotique » qui désigne tout ce qui concerne l'automatisation dans l'habitat ou « productique » pour tout ce qui a trait aux moyens de production. Aujourd'hui, la robotique est donc l'art d'automatiser des systèmes plus ou moins complexes.

Les robots :

1/ Un robot est un système (mécatronique) électronique et informatique agissant physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné. Cette machine est polyvalente et capable de s'adapter à certaines variations de ses conditions de fonctionnement. Elle est dotée de fonctions de perception, de décision et d'action. Ainsi, le robot devrait être capable d'effectuer des tâches diverses, de plusieurs manières, et accomplir correctement sa tâche, même s'il rencontre de nouvelles situations inattendues.

2/ C'est une machine pouvant manipuler des objets en réalisant des mouvements variés dictés par un programme aisément modifiable. Programmer un robot consiste, dans un premier temps, à lui spécifier la séquence des mouvements qu'il devra réaliser. Certains robots sont dotés de "sens" ; c'est-à-dire d'un ensemble plus ou moins important d'instruments de mesure et d'appréciation (caméra, thermomètre, télémètre, ...) permettant au programme du robot de décider du mouvement le mieux adapté aux conditions extérieures. Par exemple: si un robot mobile muni d'une caméra est amené à se déplacer dans un local inconnu, on peut le programmer pour qu'il contourne tout obstacle qui entraverait sa route. On

essaie également de doter des robots d'un dispositif d'intelligence artificielle afin qu'ils puissent faire face à des situations imprévues et nouvelles (le robot pourrait acquérir une certaine "expérience")

Les types de robots :

Il existe deux grandes familles de robots qui sont :

- Les robots manipulateurs.
- Les robots mobiles.

A. Les robots manipulateurs :

Un robot manipulateur est en forme d'un bras et se compose d'un certain nombre de segments qui est conçu pour manipuler ou déplacer des matériaux, outils et pièces sans contact humain direct. Ils sont des dispositifs qui permettent aux humains d'interagir avec des objets dans un environnement en toute sécurité. Les robots manipulateurs sont utilisés dans des applications industrielles pour s'effectuer efficacement des tâches telles que l'assemblage, soudage, traitement de surface, et le forage.

B. Robot mobile:

Un robot mobile est celui qui peut se déplacer dans son environnement de façon indépendante. Pour ce faire, le robot doit pouvoir naviguer, et la portée et la précision de navigation requise varie en fonction de la taille du robot et du type de sa tâche

Les classe robots:

Les robots sont répartis dans les classes suivantes :

Classe 1 : robot de manipulation manuelle : un robot à plusieurs degrés de liberté actionné par l'opérateur.

Classe 2 : robot à séquence fixe : robot de manipulation qui effectue les étapes successives d'une tâche en fonction d'une méthode prédéterminée, qui est difficile à modifier.

Classe 3 : robot à séquence variable : le même type de robot de manipulation que dans la classe 2, mais les étapes peuvent être modifiées facilement.

Classe 4 : robot playback : l'opérateur humain effectue la tâche manuellement en conduisant ou en commandant le robot, qui enregistre les trajectoires. Cette information est rappelée lorsque cela est nécessaire, et le robot peut effectuer la tâche en mode automatique.

Classe 5 : robot à commande numérique : l'opérateur humain alimente le robot avec un programme de mouvement plutôt que de lui enseigner la tâche manuellement.

Classe 6 : robot intelligent : un robot avec des moyens de comprendre son environnement, et la capacité de mener à bien une tâche malgré les changements dans les conditions ambiantes dans lesquelles celui-ci doit exécuter sa tâche.

Les composants des robots

Un robot, en tant que système, se compose des éléments, qui sont intégrés ensemble pour former un ensemble. La plus part des robots contient les éléments suivants: [Man]

- **Manipulateur:** c'est le corps principal du robot qui comprend les jonctions, les articulations, et d'autres éléments de structure du robot. Il convient de noter ici que le manipulateur seul n'est pas un robot.



Figure 10 : Un bras manipulateur

• **Effecteur finale:** cette partie est reliée à la dernière jonction (main) d'un manipulateur qui gère généralement les objets, établit des connexions à d'autres machines ou effectue les tâches requises.



Figure 11: Effecteur finale d'un robot

• **Actionneurs:** les actionneurs sont les «muscles» de manipulateurs. Le contrôleur envoie des signaux aux actionneurs, qui, à son tour, déplacent les articulations du robot et des jonctions, les types communs des actionneurs sont les servomoteurs, les moteurs pas à pas, les actionneurs pneumatiques et les vérins hydrauliques. Les actionneurs sont sous le contrôle du contrôleur.



Figure 12: Différents actionneurs d'un robot

• **Capteurs:** les capteurs sont utilisés pour recueillir des informations sur l'état interne du robot ou pour communiquer avec l'environnement extérieur. Comme chez l'humain, le dispositif de commande de robot doit connaître l'emplacement de chaque lien du robot afin de connaître la configuration du robot. Toujours comme vos principaux sens de la vue, le toucher, l'ouïe, le goût, et la parole, les robots sont équipés de dispositifs sensoriels externes comme un système de vision, le toucher et les capteurs tactiles, synthétiseur de parole, et grâce à eux le robot peut communiquer avec le monde extérieur.

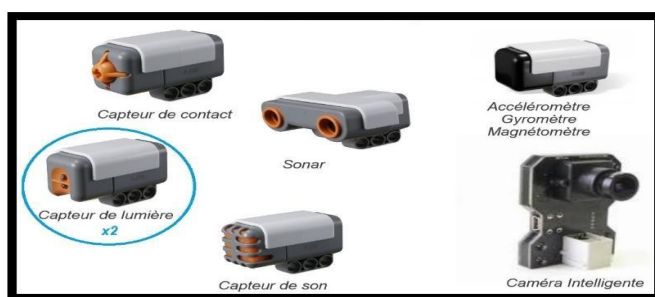


Figure 13: Différents capteurs d'un robot

• **Contrôleur:** le contrôleur est plutôt proche de votre cerveau; même si elle n'a pas la puissance du cerveau; il contrôle toujours vos mouvements. Le contrôleur reçoit les données de l'ordinateur (le cerveau du système), commande les mouvements des actionneurs, et coordonne les mouvements avec les informations envoyées par les capteurs



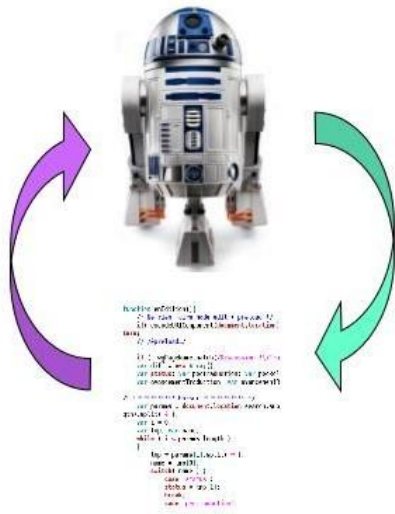
Figure 14 : Un contrôleur utilisé pour commander un robot

- **Processeur:** le processeur est le cerveau du robot. Il calcule les mouvements des articulations du robot, détermine combien et à quelle vitesse chaque joint doit se déplacer pour atteindre l'emplacement et la vitesse souhaitée, et supervise les actions coordonnées du contrôleur et les capteurs. Dans certains systèmes, le contrôleur et le processeur sont intégrés ensemble en une seule unité, et dans d'autres cas, ce sont des unités séparées.



Figure 15 : Le cerveau du robot

• **Logiciel :**



Trois groupes de logiciels sont utilisés dans un robot. L'un est le système d'exploitation qui exploite le processeur. Le second est le logiciel robotique qui calcule la motion nécessaire de chaque joint du robot basée sur des équations mathématiques. Ces informations sont envoyées au dispositif de commande. Ce logiciel peut être utilisé à différents niveaux, de la langue machine aux langues sophistiqués utilisés par les robots modernes. Le troisième groupe est la collection d'application - orientée les routines et les programmes développés pour utiliser le robot ou ses périphériques pour des tâches spécifiques telles que l'assemblage, le chargement de machines, la manutention et les routines de vision.

Architecture des robots

L'architecture des robots est un dispositif mécanique réalisant des tâches en fonction d'un programme qui lui a été assigné. Généralement, les robots ont une architecture que l'on peut schématiser dans la plupart des cas de la sorte [BEA06] :

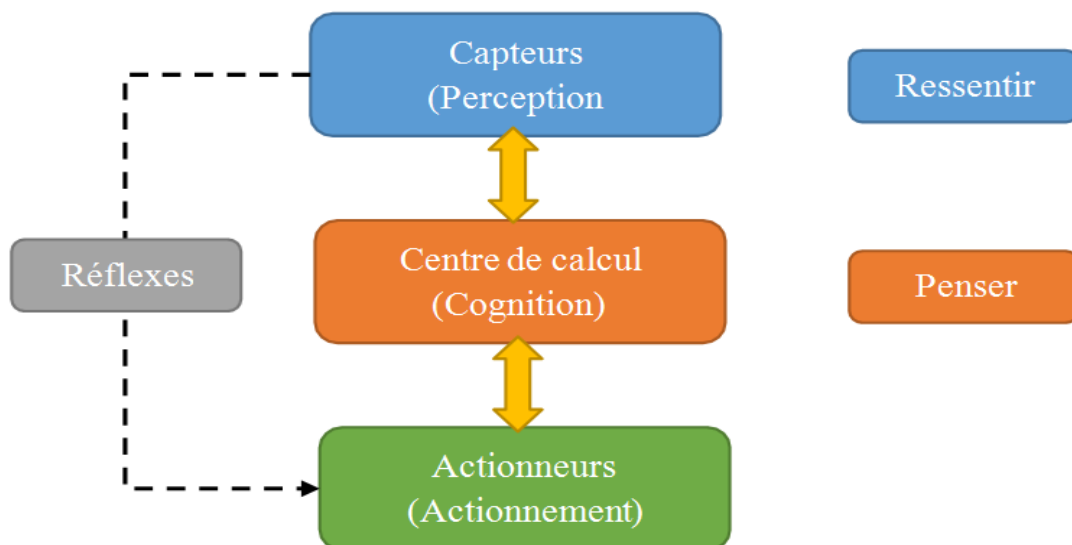


Figure 16: Architecture des robots

Navigation des robots :

D'après **Levitt et Lawton (1990)**, la navigation est définie comme le procédé permettant de répondre aux trois questions suivantes :

1. "Où suis-je ?".
2. "Où sont les autres lieux par rapport à moi ?".
3. "Comment puis-je atteindre ces autres lieux depuis l'endroit où je me trouve?".

Donc, la navigation est la science [WIK 2006] et l'ensemble des techniques qui permettent de :

- ✓ connaître la position (coordonnées) d'un mobile par rapport à un système de référence, ou par rapport à un point fixe déterminé.
- ✓ calculer ou mesurer la route à suivre pour rejoindre un autre point de coordonnées connues en respectant un certain nombre de contraintes et de critères qui découlent de plusieurs facteurs, qui dépendent généralement des caractéristiques du robot, de l'environnement, et du type de tâche à exécuter.
- ✓ calculer toute autre information relative au déplacement de ce mobile (distance et durée, vitesse de déplacement, heure estimée d'arrivée, etc.).

Les systèmes multi-robots (SMR):

Un système multi-robots est un système constitué d'un ensemble de robots, homogènes ou hétérogènes, mobile ou fixe, ces robots sont capables de communiquer entre eux et coopérer pour améliorer l'efficacité d'exécution des tâches ou encore pour permettre d'exécuter des tâches impossibles à exécuter de façon individuelle.

Ces dernières décennies, l'idée d'utiliser un groupe de robots mobiles dans un seul environnement a attiré un grand nombre de roboticiens, notamment dans le domaine de la commande. L'intérêt porté à ce champ d'investigation est justifié par toutes les tâches possibles et variées qu'offre l'utilisation d'un groupe de robots par rapport à un seul. Les avancées technologiques réalisées dans le domaine de la communication entre systèmes ont

grandement contribué aux résultats obtenus. La littérature est si riche qu'il est difficile de rassembler un état de l'art exhaustif sur le contrôle des SMR.

Le domaine de la robotique distribuée a ses origines dans les années 1980, lorsque plusieurs chercheurs ont commencé à étudier les questions dans plusieurs systèmes de robots mobiles. Avant cette date, la recherche s'est concentrée sur les systèmes robotiques simples qui n'impliquaient pas les composants robotiques de résolution de problèmes

Un système multi robot est un système constitué d'un ensemble de robots, homogènes ou hétérogènes, mobile ou fixe, ces robots sont capables de communiquer entre eux et coopérer pour améliorer l'efficacité d'exécution des tâches ou encore pour permettre d'exécuter des tâches impossibles à exécuter de façon individuelle.

Pourquoi un système multi robots :

Un système multi robots peut effectuer des tâches difficiles ou même impossibles à accomplir par un seul robot. Une équipe de robots fournit une certaine redondance, elle contribue à l'accomplissement d'une tâche de manière collaborative

Ce système offre plusieurs avantages par rapport aux systèmes mono-robot ; les motivations les plus communes pour le développement de solutions de systèmes multi robots sont les suivants :

- La complexité de la tâche à réaliser est trop élevée pour un seul robot donc elle Nécessite une coopération d'un ensemble de robots.
- La construction de plusieurs robots simples avec des ressources bornées est beaucoup plus facile que la construction d'un seul robot complexe et puissant.
- L'introduction de plusieurs robots augmente la robustesse grâce à la redondance.
- La rapidité : pour obtenir un niveau de performance élevé les tâches sont effectuées en parallèle. A titre d'exemple : l'exploration parallèle d'un environnement inconnu par un ensemble de robots mobiles, dans le but de faire soit une cartographie de l'environnement, soit la réalisation d'une tâche de fourragement. Ces deux tâches peuvent être réalisées par un seul robot, mais l'ajout d'autres robots va faire en sorte d'accélérer l'exécution des tâches en question.
- La robustesse : les performances du contrôle dans un SMR ne sont pas trop affectées en cas de défaillance d'un robot.
- La flexibilité : il est possible d'exécuter les tâches désirées, de diverses manières. Ceci est induit principalement par la redondance des entités robotiques

Les domaines d'utilisation de système multi robots :

L'un des objectifs de l'utilisation des robots est de remplacer ou d'aider les êtres humains dans la réalisation des certaines tâches. C'est particulièrement souhaitable pour des tâches dangereuses comme l'exploitation spatiale et sous-marine, l'exploration de l'intérieur d'une centrale nucléaire ou encore ennuyeuses tel que la surveillance d'un site, les applications militaires. Les robots peuvent effectuer des tâches automatiquement, mais ils sont aussi dotés d'une certaine intelligence.

- ❖ Robotique médicale : La robotique se développe et se perfectionne au grand bénéfice de la médecine, un robot médical est un appareil utilisé soit pour former les médecins, pour aider les médecins dans leurs tâches ou les remplacer ; en chirurgie les robots peuvent remplacer les chirurgiens qui les contrôlent à distance et aussi pour remplacer un organe du corps d'un malade, en général c'est un appareil aide à soigner des patients.
- ❖ Robotique industrielle : Les robots industriels ont été développés pour intervenir dans les milieux dangereux (nucléaire), ils servent aussi dans le maniement d'objets lourds, des robots de peinture et de soudure dans l'industrie automobile, ils sont recommandés pour des tâches répétitives et précises.
- ❖ Robotique militaire : Un robot militaire est un robot autonome ou contrôlé à distance conçu pour des applications militaires, les robots sont plus précis dans leurs tirs et aussi beaucoup plus dure à détruire qu'un soldat. De plus il y a des avantages d'un point de vue économique car les robots n'ont besoin d'aucune formation puisqu'ils fonctionnent à partir de logicielles, et ils n'ont pas besoin de nourriture ou de soins médicaux, mais aussi les prix des robots de guerre sont excessivement élevés on a comme exemple le déminage puisque notre objectif est la comparaison entre les différents stratégies de coopération nous avons choisie le déminage comme domaine d'application :

Le déminage est un ensemble d'actions visant à l'élimination des mines terrestres ou navales d'une zone. Le déminage est la détection des mines, puis leurs enlèvements sans perdre des vies humaines.



Figure 17 : démineur humaine

Catégories des systèmes multi-robots

- ✚ Système multi-robots homogènes : Un système multi-robots homogènes est un ensemble de robots qui ont les mêmes caractéristiques, ils doivent coopérer entre eux
- ✚ Un SMR hétérogènes: est un ensemble de robots mixtes, c'est-à-dire les robots ne sont pas identiques, il peut y avoir des robots fixes et des robots mobiles ; dans ce cas chaque robot s'occupe de sa propre sous tâche.

Les problèmes de la coordination dans les systèmes multi-robots :

Un système multi-robots est considéré comme un système distribué, puisque les tâches sont distribuées sur l'ensemble des robots du système. L'un des problèmes des systèmes multi-robots est la coordination, ce problème est séparé en quatre parties, la communication, la computation, la configuration, et la coordination

- La communication : définir comment les agents communiquent entre eux.
- La computation : elle définit la mise en œuvre du comportement des agents individuels. elle détermine ainsi ce qui est communiqué.
- La configuration : elle définit la structure d'interaction et elle indique quel sont les agents existants dans le système et quel sont les agents qui peuvent communiquer entre eux.
- La coordination : elle définit les modes d'interaction, c'est-à-dire déterminé à quel moment certaines communications ont lieu.
- la coopération : Dans mon travail j'ai focalisé mon intention sur la coopération entre les entités du système multi-robots.

Coopération des robots mobiles

La robotique collective est synonyme d'existence d'un groupe de robots. Elle stipule non seulement la mise en place d'un contrôle individuel pour chaque robot, mais impose aussi l'utilisation de stratégies de contrôle appropriées afin que l'assemblage de toutes ces entités robotiques engendre des configurations cohérentes et efficaces pour la réalisation des tâches désirées.

Les stratégies de coopération permettent a un robot mobile de se déplacer pour rejoindre un but, réaliser une mission ou une tâche. Nous pouvons distinguer plusieurs types de stratégies, cette classification permet de distinguer les stratégies sans modèles internes et les stratégies avec modèle interne.

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.

L'autonomie, l'intelligence et la relation robot-tâche-environnement.

A. L'autonomie :

Il y a deux principales définitions de l'autonomie [Webster 81] :

- comportement basique initialement programmé sans aide extérieur.
- avoir le pouvoir de l'auto-gouvernance.

Les robots qui contiennent des contrôleurs et des blocs d'alimentation sont autonomes dans le premier cas, mais ça reste une faible autonomie. Cependant, pour faire face à des situations imprévues et s'adapter à des environnements changeants, la puissance de l'auto-gouvernance («forte autonomie») est nécessaire. L'autonomie dans ce cas signifie que la machine est capable de déterminer sa ligne de conduite par son propre raisonnement, plutôt que d'en suivre un pré-défini par des instructions. La forte autonomie exige la capacité de planifier et d'apprendre de l'expérience

Un robot mobile autonome, a alors la capacité de se déplacer dans son environnement pour effectuer un certain nombre de tâches différentes, et est aussi capable de s'adapter aux changements de son environnement, apprendre de l'expérience et de modifier son comportement en conséquence, et de construire des représentations internes de son monde qui peut être utilisée pour des processus de raisonnement comme la navigation.

B. L'intelligence:

Nous pouvons lire dans les manuels de vulgarisation scientifique et les magazines à propos des machines intelligentes, et des robots intelligents une définition utilisable et claire de l'intelligence. L'intelligence se réfère au comportement, et l'aune à laquelle nous la mesurons est très dépendante de notre propre éducation, notre compréhension et notre point de vue. Alan Turing a écrit en 1947 :

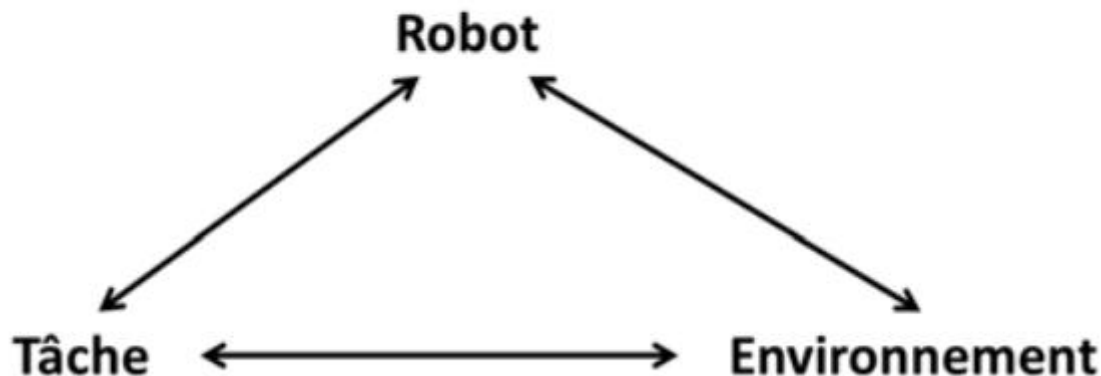
“La mesure dans laquelle nous considérons que quelque chose se comporte d'une manière intelligente est déterminée autant par notre propre état d'esprit et de formation que par les propriétés de l'objet considéré. Si nous sommes en mesure d'expliquer et de prédire son comportement, nous pouvons tenter d'imaginer son intelligence. Par conséquent, le même objet peut être vu comme intelligent par un homme, mais le jugement d'un autre peut dire qu'il ne l'est pas, le deuxième homme ayant découvert les règles de son comportement”.

L'observation de Turing décrit l'intelligence comme un objectif inaccessible, parce qu'au moment où nous comprenons le fonctionnement d'une machine, ou pouvons prédire son comportement, il n'est plus question d'intelligence. Cela signifie que tant nous progressons dans notre compréhension de la conception et l'analyse des systèmes artificiels, l'intelligence évolue avec nous, tout en restant ainsi hors de notre portée.

C. La tripartite Robot Tâche Environnement

Le contrôle des robots mobiles nécessite la juxtaposition de trois phases : la perception, la décision, et l'action. La perception construit un modèle même simple de l'environnement du robot, la décision utilise ce modèle pour générer des consignes de mouvement, et finalement l'action transforme ces consignes en commandes adéquates pour les effecteurs du robot. Un contrôle bien mené nécessite donc la maîtrise de ces trois phases [Adouane 05].

Comme l'intelligence, le comportement d'un robot ne peut pas être évalué indépendamment de l'environnement du robot et de la tâche que le robot doit effectuer. Le robot, la tâche et l'environnement dépendent les uns des autres et s'influencent mutuellement



Les tâches coopératives multi-robots:

Pour qu'une tâche soit qualifiée de coopérative, les robots chargés de l'accomplir doivent coordonner leurs actions en interagissant dans le même environnement, et en partageant impérativement des objectifs et des ressources communs : pousser ou porter le même objet, cartographier un environnement, etc. sont des exemples de tâches coopératives dans lesquels chaque robot doit tenir compte des actions des autres en accomplissant la sienne [Adouane 05].

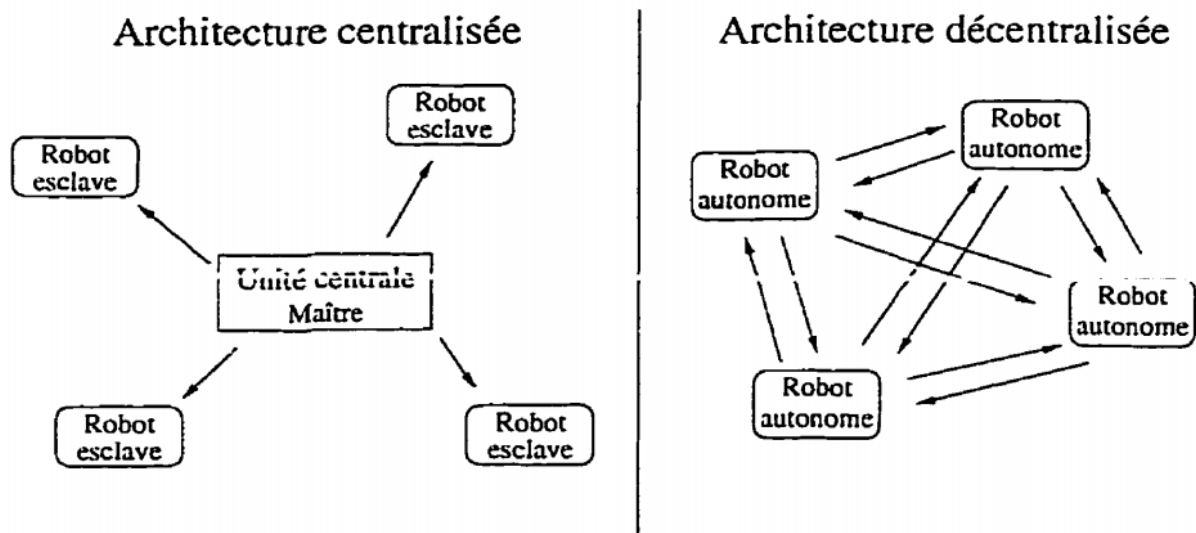
La tâche coopérative peut être considérée comme l'objectif à atteindre par le système multi-robots. Cet objectif change selon les applications désirées ou les types de robots utilisés.

Les stratégies de coopérations :

Les systèmes multi-robots **ont** reçu beaucoup d'attention dans la communauté scientifique ces dernières années. Lorsque plusieurs robots travaillent en collaboration pour accomplir une même tâche,

le contrôle du système est plus délicat surtout lorsque les robots doivent interagir entre eux, c'est-à-dire lorsque leurs actions imposent des contraintes restrictives sur les actions des autres robots.

On distingue deux principaux types d'organisation pour le contrôle d'équipes de robots (Noreils, 1993) : les architectures centralisées et les architectures décentralisées



Architectures centralisées et décentralisées

Figure 18 : Architectures centralisées et décentralisées

Dans les architectures centralisées, la prise de décision est faite par une entité unique se basant sur des informations globales (Parker, 1993). Dans ce cas, les robots du système sont perçus comme les actionneurs d'un même robot

la principale limitation réside dans la communication entre robots. En effet, dans de tels systèmes, le contrôle de chaque robot nécessite d'obtenir les informations provenant de ses senseurs, de traiter ces informations pour choisir les actions à initier, et d'envoyer au robot les commandes nécessaires à son contrôle. Ces informations doivent être transmises au robot via un **canal** dont la capacité peut affecter les performances du système. En effet, le temps nécessaire à la transmission de l'information affecte le temps de réponse des robots et le succès de la tâche.

De plus, l'information peut être altérée ou détruite par la transmission en raison de perturbations. En général, on cherche à limiter le flot de communication entre robots par l'utilisation de protocoles (Mataric, Nilsson et Simsarian, 1995; Wang, 1994), par une structure de communication appropriée (Chen et Luth, 1993; Hashimoto, Oba et Eguchi, 1993) ou même en estimant l'état ou les intentions des autres robots en faisant usage de modèles probabilistiques (Fukuda et Sekiyama, 1994).

Enfin, le contrôle de tous ces robots par un système central devient rapidement complexe et exigeant lorsque le nombre de robots contrôlés augmente

● **L'architecture centralisée**

- - un seul point de contrôle
- - tout les informations disponibles
- - difficulté de communication
- - un robot supervise quelques autres robots,
- - qui eux-mêmes en supervisent d'autres. . .
- - pyramide de responsabilité ;
- - besoin de robustesse des niveaux élevés.

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 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 les quels 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

Les architectures centralisées ont été essentiellement motivés par deux facteurs [Khoshnevis 98] :

- l'ambition d'alléger la structure physique des robots : débarrasser les robots de capteurs embarqués, et la puissante unité de calcul. Ainsi, réduire considérablement leur coût
- . — le désir d'une meilleure connaissance de l'environnement global par la concentration des capteurs au niveau d'une unité centrale, ce qui peut assurer une meilleure prise de décision par rapport à des robots se basant que sur des informations locales fournies par des capteurs embarqués.

Dans ce type d'architecture, le contrôle est délocalisé par rapport à la structure physique des robots et se trouve au niveau d'une unité centrale ou planificateur central dans [Noreils 93]) qui gère et garantit l'exécution de la tâche. Cette unité contient les parties sensorielles (capteurs) afin de collecter les informations de l'environnement, et calcul les consignes de commande (positions, vitesses) qu'elle envoie aux entités robotiques. Elle est aussi responsable de prendre les décisions pour la réalisation de la tâche globale et les communiquer aux robots. Elle doit donc avoir une puissante capacité calculatoire pour satisfaire à toutes ces exigences. Dans les architectures de contrôle centralisées, la manière de coordonner les actions entre le superviseur et les robots peut être comparée à celle de groupements sociaux,

L'architecture décentralisée

- - information locale ;
- - communication locale ;
- - grande robustesse aux pannes ;
- - faible réactivité à un changement d'objectif du système ;
- - difficulté d'obtention d'un comportement global.

Pour les systèmes décentralisés, la difficulté majeure provient du fait que les robots n'ont pas une connaissance de toutes les informations sur l'état globale du système.

Chaque unité ne peut donc pas connaître, en tout temps, l'état des autres robots de même que leurs intentions. est par conséquent difficile pour un robot de choisir ses actions en accord avec celles du reste de l'équipe afin de faire réussir la tâche,

L'idée centrale de ce travail est donc de développer une architecture utilisant principalement des informations locales pour faire face aux changements dynamiques de l'environnement, tout en utilisant des informations globales pour bien coordonner les actions des robots entre eux.

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 termes de fiabilité, de flexibilité, d'adaptabilité et de robustesse.

Néanmoins, les solutions proposées par ce type d'architecture sont souvent sous-optimales.] 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.] 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. est un exemple type de système multirobots 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 coordonner l'exécution des tâches. De nombreux travaux de la littérature cités dans les sections précédentes appartiennent à cette catégorie de système décentralise

Alors on centralise ou on distribue

Chaque type d'architectures de contrôle présente ses avantages et ses inconvénients. Il est alors difficile voire impossible de privilégier un type plutôt qu'un autre sans connaître la nature de la mission confiée au SMR ni le contexte de sa réalisation

Modules	Architecture centralisée	Architecture distribuée
Perception et localisation	L'unité centrale utilise des capteurs centralisés et calcule la localisation absolue dans l'environnement de chaque robot. Les robots ne connaissent pas leur environnement.	Les robots utilisent leurs capteurs embarqués et se localisent de façon relative par rapport à leurs consignes.
Décision	L'unité centrale décide, calcule et génère directement des variables de commande (vitesses) qu'elle envoie aux robots.	Les robots génèrent leurs propres consignes de type chemin, trajectoire, points de passages, etc.
Action	Les entités robotiques appliquent directement les variables de commande calculées par l'unité centrale à leurs moteurs.	Les robots assurent le respect des consignes qu'ils ont générées à travers des lois de commande avec un asservissement en boucle fermée sur ces consignes.

– Architecture centralisée versus distribuée [Adouane 05].

Dans les architectures centralisées, l'unité centrale dispose d'informations globales de l'environnement, ce qui aide à la coordination d'actions entre les éléments du SMR, et à la prise de décision. La limitation de ce type d'architectures réside dans cette même unité centrale du fait qu'elle doit gérer le contrôle et la communication de toutes les entités robotiques. De plus, le SMR dépend entièrement de cette unité, ce qui signifie qu'un défaut du superviseur central implique l'arrêt complet du système. Les architectures de contrôle distribuées quant à elles, permettent de tirer profit de la distribution des ressources : chaque robot se contente de prendre une décision le concernant, en fonction des informations locales de l'environnement délivrées par ses propres capteurs et de sa communication avec ses voisins. Cette distribution allège la tâche des unités de calcul. De plus, les défauts de fonctionnement sont mieux gérés et la panne d'un robot ne signifie pas l'arrêt complet du SMR. En revanche, l'absence d'un superviseur global et d'une information globale sur l'environnement rend difficile la coordination des robots et l'optimisation de l'exécution de la mission

. Le tableau, [Adouane 05] résume les principales différences observées entre ces deux types d'architectures centralisées et distribuées. En pratique, beaucoup de SMR n'ont pas un contrôle strictement centralisé ou strictement distribué et utilisent des architectures de contrôle hybrides centralisées/distribuées pour bénéficier des avantages des deux approches. des robots pourtant décisionnellement autonomes, sont soumis néanmoins ponctuellement à un planificateur central

Conclusion

Dans ce chapitre, nous avons présenté la robotique, les composants d'un robot, et ces types. Par la suite on a mis le point sur les systèmes multi-robots leurs domaines d'utilisation ainsi que comment les robots puissent coopérer dans tels systèmes. Le chapitre qui suit est consacré pour la conception

Chapitre IV
Conception

Introduction

Un robot est censé à remplacer l'être humain dans la résolution des problèmes et la réalisation des tâches.

Les problèmes à résoudre sont parfois assez complexes et nécessitent la coopération entre un groupe de robots mobiles; on choisit comme domaine d'application le secteur militaire et en particulier le déminage (Voir chapitre II).

Le déminage est une tâche complexe qui demande la coopération entre les différents robots qui constitue le système multi robot.

Dans ce chapitre nous allons présenter notre conception qui se base sur l'utilisation des différents stratégies de coopération (Centralisé décentralisée et hybride) en détaillant notre système multi robots et chaque stratégie utilisée.

Systeme multi robot :

Le système multi robots choisi pour notre conception est hétérogène composé de:

1. Robot localisateur chargé de localiser les mines.
2. Robot démineur chargé de déminer.
3. Robot coordinateur chargé d'assurer la coordination entre les différents robots du système.

Les deux premiers types de robots sont des robots mobiles qui peuvent être vu comme étant une entité autonome qui perçoit son environnement et agit sur ce dernier.

Pour cela, on explique dans la section suivante la navigation des robots.

Navigation des robots

Afin de permettre à un robot d'atteindre une position souhaitée (dans notre cas la position des mines), il existe deux grandes familles de méthodes. Les méthodes *sans trajectoire explicite* (champs de potentiels, réseaux de neurones et logique floue) qui cherchent plutôt à contrôler le mouvement global du robot de manière à le guider vers son but. L'autre famille, celle des méthodes de *suiti de trajectoire*, permet au robot de suivre « du mieux possible » une trajectoire de référence donnée, connaissant les contraintes cinématiques [MORETTE, 2009].

Nous avons opté pour la méthode sans trajectoire explicite. Et ce, en rendant les obstacles répulsifs pour le robot dans le but qu'il s'en écarte et reste à bonne distance, et rendre par contre le reste de l'environnement (le vide) attractif pour qu'il y navigue.

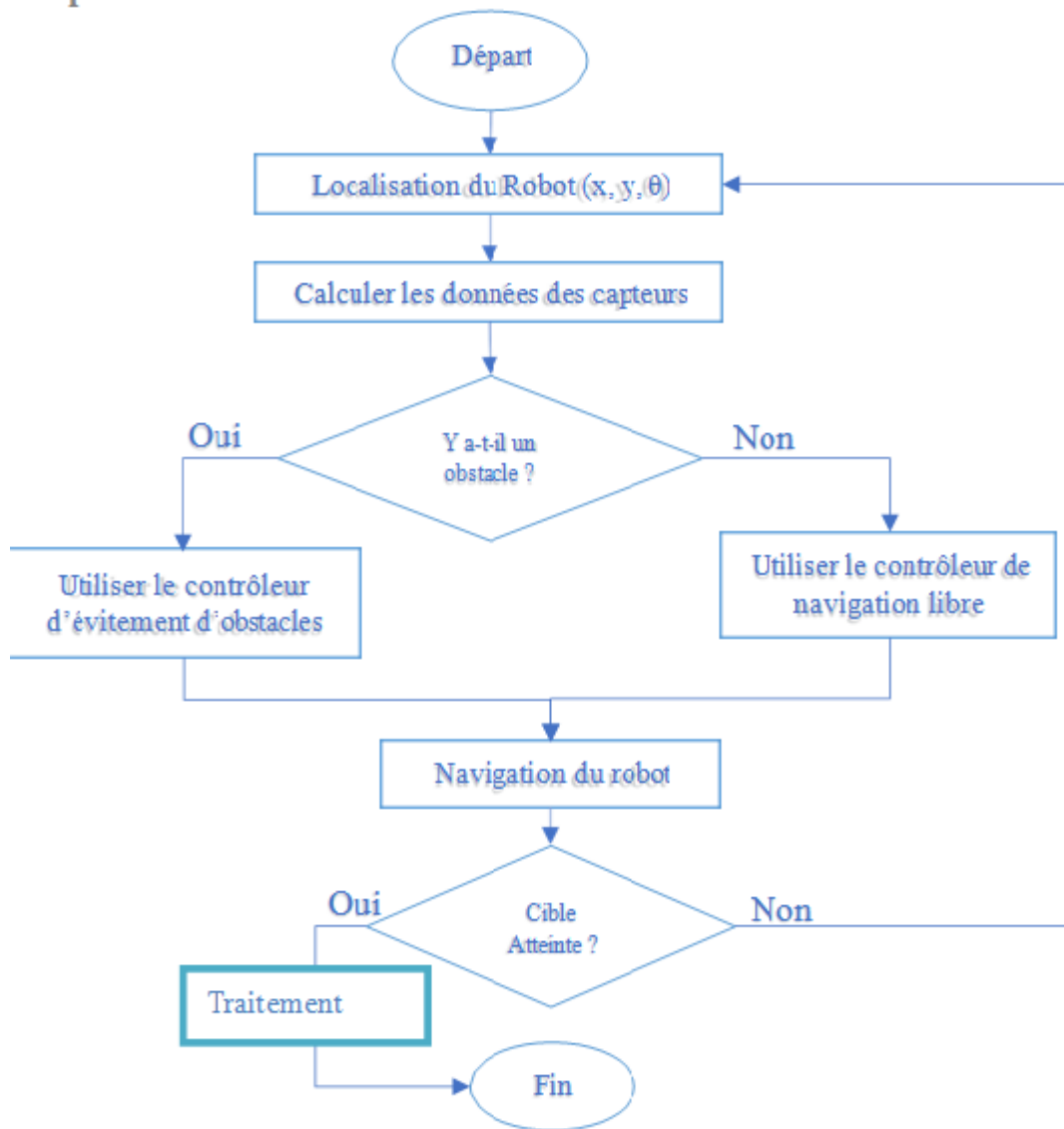


Figure 19 : organigramme générale de robot de la navigation

Description de l'environnement :

- L'environnement : l'espace de navigation est un champ de mines
- Les robots : dans notre cadre on trois types de robots
 1. Robot localisateur
 2. Robot coordinateur : il ya deux types de robot coordinateur .coordinateur principal et coordinateur de chaque groupe
 3. Robot démineur : chaque robot démineur a un nom, position, action et groupe
 - Objets : on a deux types d'objets obstacle et les mines :
 - Obstacles : les murs de l'environnement.
 - Les mines : chaque mines a un identifiant, position, action

Présentation des stratégies

On présente trois stratégies : centralisée, décentralisée et hybride avec plusieurs techniques on détaille la fonction de chaque robot.

La stratégie Centralisée :

Dans cette stratégie on a trois types de robot : localisateur, démineur et coordinateur.

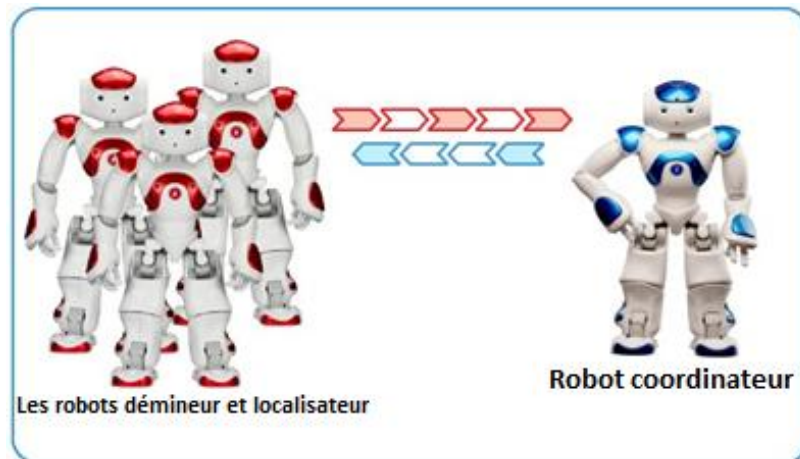


Figure 20 : Architecture Centralisée

1. Le robot localisateur: est capable naviguer dans le champ des mines et une fois il détecte une mine il retourne ses coordonnées au robot coordinateur pour le traiter.

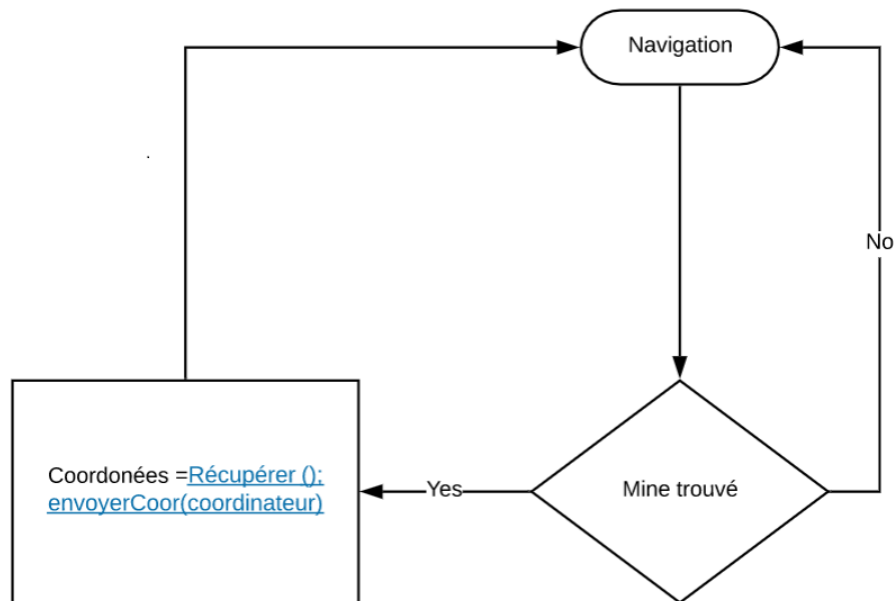


Figure 21 : Organigramme de robot localisateur

2. Le rôle de robot coordinateur est de recevoir les coordonné des mines envoyé par les localisateurs ensuite il commece la recherche et la sélection de robot démineur en lui envoyant les coordonnées de la mine.

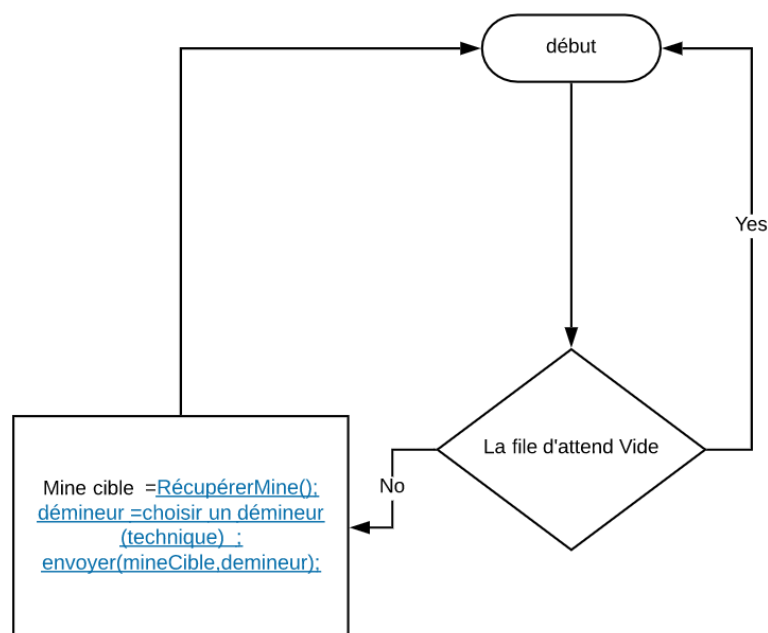


Figure 22 : Organigramme de robot coordinateur

3. Le role de robot démineur est de déminer en recevant les coordonnées de la mine cible il navigue vers et la détruire.

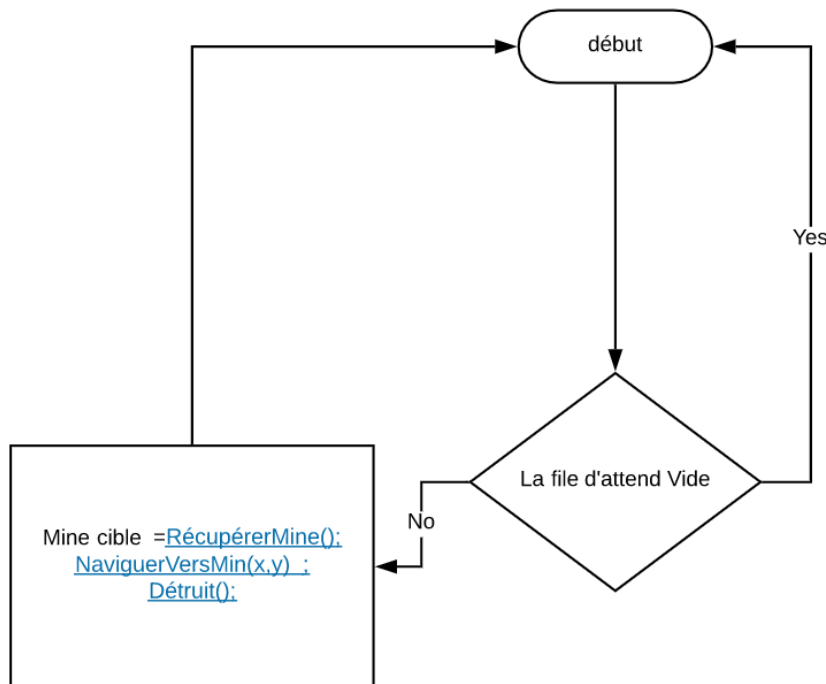


Figure 22 : Organigramme de robot coordinateur

La figure .ci-dessous est un diagramme de séquence de la stratégie précédemment expliqué (Centralisée). Le diagramme de séquences est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML. Dans notre cas, ce diagramme représente l'interaction qui entre le robot

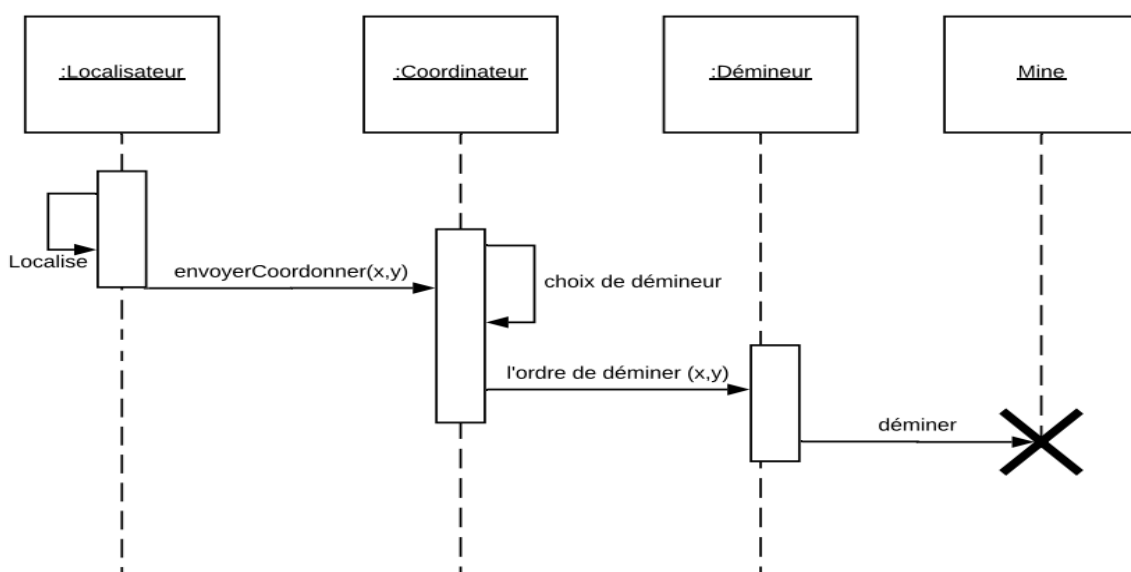


Figure 23 : Diagramme de séquence de la stratégie centralisée

La stratégie Décentralisée :

Dans cette stratégie on utilise seulement les robots localisateurs et les robots démineurs.

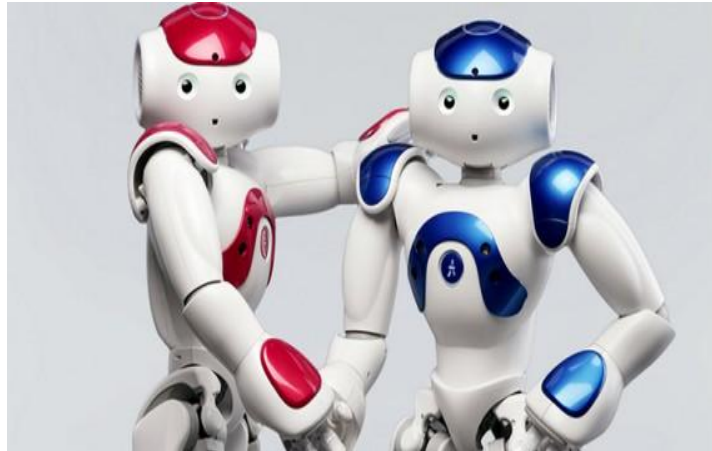


Figure 24 : Robot localisateur et démineur

Le robot localisateur localise les mines et envoie les coordonnées au robot démineur tout dépend de la technique choisie.

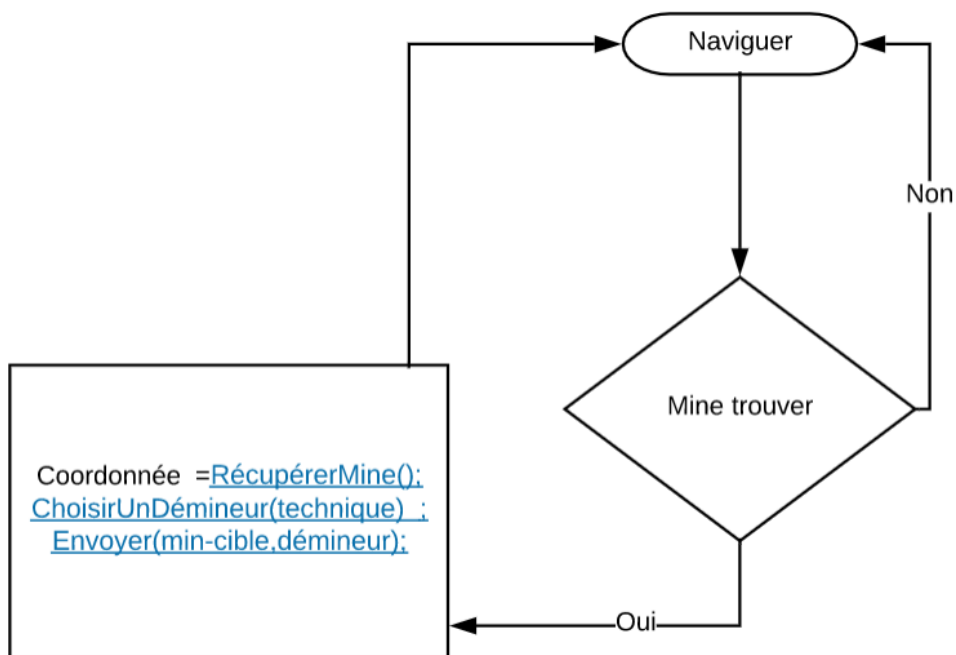


Figure 25 : Organigramme de robot localisateur

La figure .ci-dessous est un diagramme de séquence de la stratégie précédemment expliquée (Décentralisée).

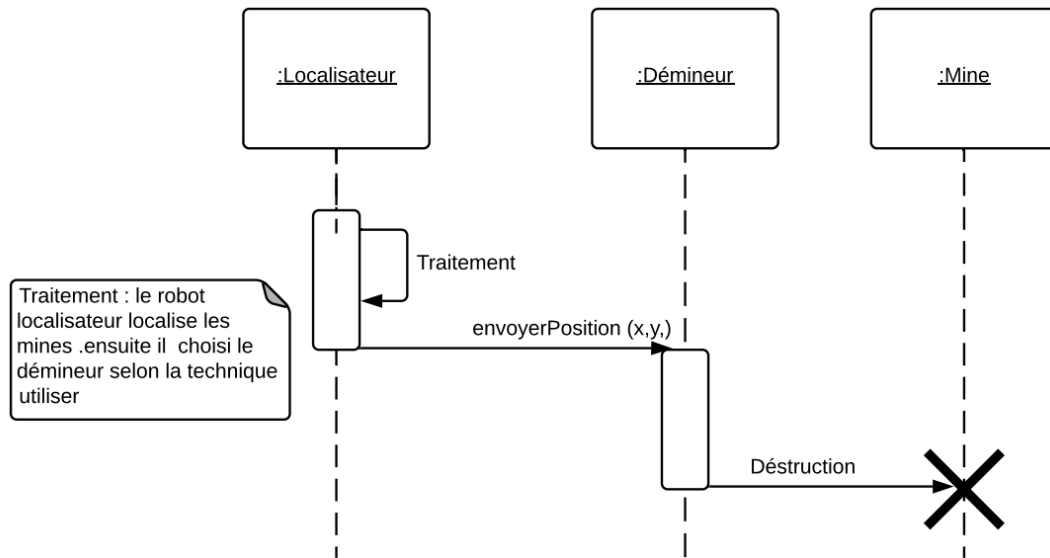


Figure 26 : Diagramme de séquence de la stratégie Décentralisée

La stratégie hybride :

Dans cette stratégie il ya une combinaison locale et globale entre les deux stratégies précédentes. En plus des robots localisateurs, démineurs, coordinateur on se sert du robot coordinateur principal.

Ci dessous les rôles de chaque robot :

- Le robot localisateur : localise les mines et envoie les coordonnées au robot coordinateur.
- Le robot coordinateur reçoit les coordonnées des mines. Et transmet au robot coordinateur principal. Et envoie les coordonnées au démineur.
- Le robot coordinateur principal choisit quel est le coordinateur le plus proche de la mine et envoie les coordonnées de la mine.

La figure ci-dessous est un diagramme de séquence de la stratégie précédemment expliquée (hybride).

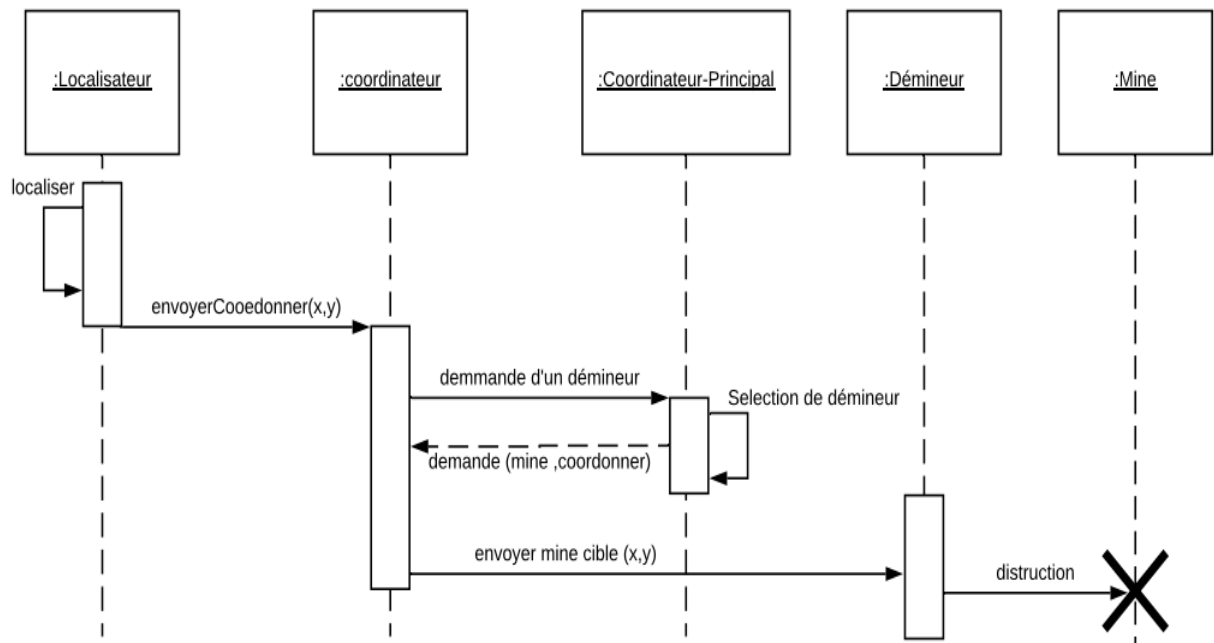


Figure 27 : Diagramme de séquence de la stratégie hybride

Présentation des techniques :

le Dans le choix du démineur on s'est basés sur trois techniques différentes :

- Choix du plus proche démineur : le robot coordonateur va choisir le robot le plus proche à la mine en fonction de la distance calculée entre le robot démineur et de la mine.
- Le choix de démineur libre : le robot coordonateur fait le choix du robot démineur selon leur position (plus proche) et l'état (libre ou occupé).
- Le choix de l'équilibrage de charge : le robot coordonateur choisit le robot destructeur qui a détruit le moins de mines

Chapitre
Implémentation

Introduction :

Ce chapitre a pour but d'apprécier les résultats de la comparaison et rendre tangible sa conception en visualisant les résultats. Pour ce faire, Nous commençons par la présentation des technologies employées pour passer ensuite à l'exposition de l'architecture générale de notre application et fini par la présentation de ses interfaces.

Présentation des technologies de développement utilisées :

Cette partie est consacrée à la présentation des différentes technologies nous ayant servi d'outils tout au long du développement de notre application.

A. Environnement de développement :

Nous présentons dans ce qui suit le cadre de la réalisation de l'application ainsi que les outils qui ont été utilisés durant la phase de sa conception et son implémentation.

B. Environnement matériel :

Le travail a été réalisé dans un environnement Windows 8.1 sur une machine Dell ayant Intel® Core™ i5, 2.53 GHZ comme microprocesseur avec 8Go de RAM et 500Go de capacité de disque dur.

C. Environnement logiciel :

Modélisation :

- **Langage : UML (Unified Modeling Language)** est un langage universel de modélisation graphique et textuel qui permet de présenter et de manipuler les concepts objet avec une considérable précision grâce aux différents diagrammes qu'il propose. Il facilite la compréhension de représentations abstraites complexes.
- **Outil : Creately** est un logiciel de création de diagrammes et de conception en ligne, opéré par *Cinergix* et employé par des équipes autour du monde pour communiquer plus efficacement. Il propose différentes sortes de diagrammes tels que les organigrammes et les multiples diagrammes UML.

Implémentation

- **Netbeans IDE:** est un environnement de développement intégré (IDE) open source conçu par Sun Microsystems. Écrit en XML, Java, NetBeans prend en charge divers langages de programmation à l'exemple de Java, PHP, C/C++, Python et JavaScript. Il fournit toutes les facilités d'un IDE moderne (éditeur graphique d'interfaces, refactoring, éditeur de pages Web, éditeur en couleurs). Disponible sous Windows, Linux, Solaris et Mac OS X, il est indépendant des systèmes d'exploitation et constitue une riche plateforme qui permet le développement d'applications sur serveurs (applications web et JEE), mobiles et Services Web XML.
- **Java :** Java est un langage de programmation évolué orienté objet créé par Sun Microsystems en 1995. Il est compilé et basé sur une architecture logicielle très particulière nécessitant une machine virtuelle Java étant donné qu'il est Multi plateformes. Il est accompagné d'un ensemble énorme de bibliothèques standard couvrant de très nombreux domaines, notamment des bibliothèques graphiques. Sa stabilité, sûreté et surtout sa portabilité font de lui un langage très utilisé. [SIM]
- **JFreechart :** JFreeChart est une bibliothèque open source qui permet d'afficher des données statistiques sous la forme de graphiques. Elle possède plusieurs formats dont le camembert, les barres ou les lignes et propose de nombreuses options de configuration pour personnaliser le rendu des graphiques. Elle peut s'utiliser dans des applications standards ou des applications web et permet également d'exporter le graphique sous la forme d'une image. [JFree]
 - **Scene builder :** Scene Builder est un outil de création et de design d'interface graphique pour les applications JavaFX, sans les codes en glissant et déposant des composants vers une zone de travail, modifiant leurs propriétés et personnalisant les feuilles de style CSS.
- **Simbad :** Simbad est un simulateur de robot Java 3D à objectifs scientifiques et éducatifs. Étant open source et gratuit, il est principalement consacré aux chercheurs / programmeurs qui veulent une base simple pour étudier l'intelligence artificielle, l'apprentissage automatique et plus généralement les algorithmes AI, dans le cadre de la robotique autonome et des agents autonomes. Il fournit multiples fonctionnalités telles que la visualisation et détection 3D, capteurs sensoriels (sonars), de contact (bumpers) et visuels (caméra couleur monoscopique), ainsi que des Interfaces utilisateur pour le contrôle.
- **Java 3D :** Java 3D est une interface de programmation (API) pour la plateforme Java développée par Sun Microsystems. C'est une extension du langage Java visant la création et la manipulation des scènes 3D. [J3D] [J3D2]

Architecture générale de notre système :

Cette section propose un schéma explicatif qui décrit le fonctionnement du système développé.

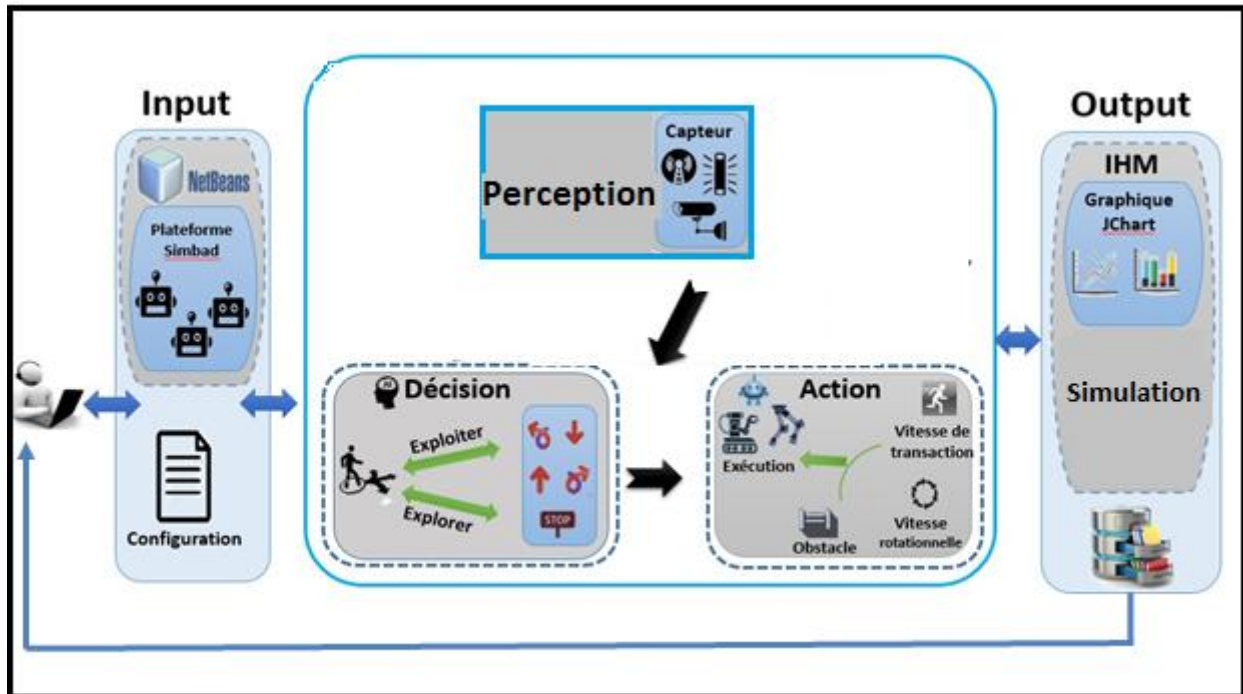


Figure 28 : Architecture de notre système

La figure. Montre que l'utilisateur commence d'abord par la configuration des paramètres de la simulation comme données d'entrée pour pouvoir commencer.

Présentation des interfaces

Les figures ci-dessous représentent notre système qui est composé de plusieurs interfaces qui seraient représentées dans ce qui suit sous forme de capture d'écran.

- **Interface principale:** La figure représente l'interface principale de l'application, c'est elle qui permet à l'utilisateur de saisir la configuration de la simulation (nombre de robots localisateurs, nombre de robot démineurs, nombre de mine ainsi que le choix de la stratégie), et lancer une simulation avec ces paramètres là; et c'est aussi l'interface depuis laquelle il peut visualiser les statistiques de simulation.

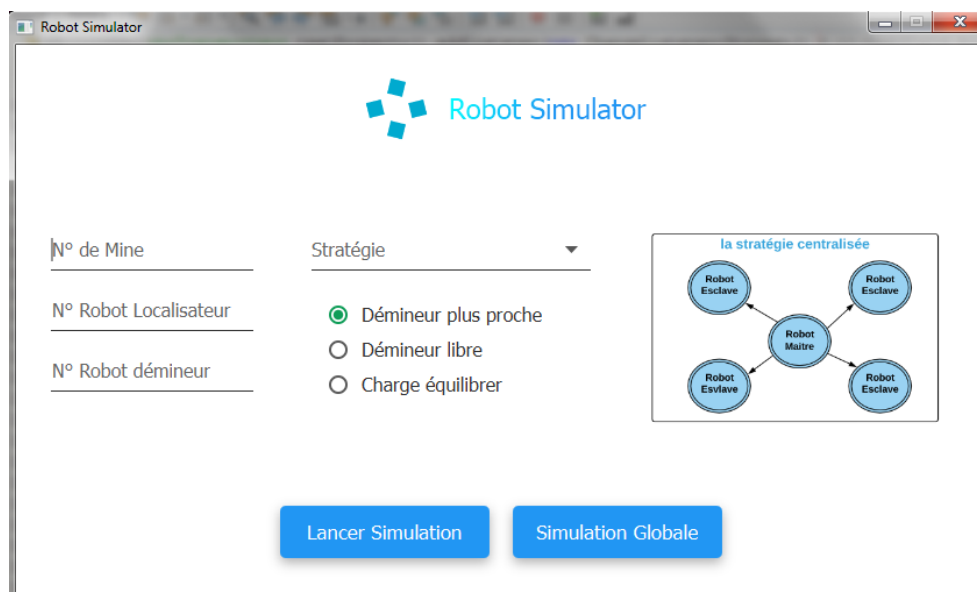


Figure 29 : Configuration des paramètres de lancement

- **Interface de simulation centralisée :** La figure représente l'interface de simulation centralisée

Orientation:

1 : le robot coordinateur.

2 : le robot localisateur.

3: le robot démineur.

4 : la mine.

5: la mure de l'environnement.

6 : panneau de configuration du robot (la vitesse , démarrer ,redémarrer, pause).

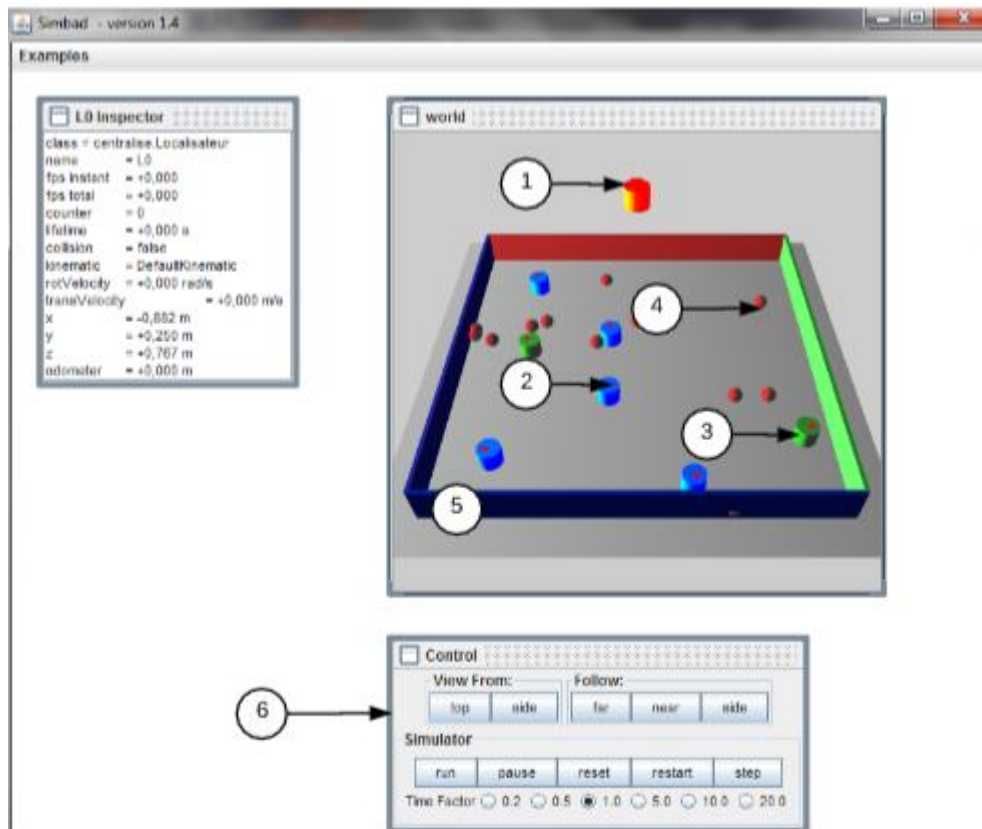


Figure 30 : Interface de stratégie centralisée

- **Interface de simulation décentralisée** : La figure représente l'interface de simulation décentralisée

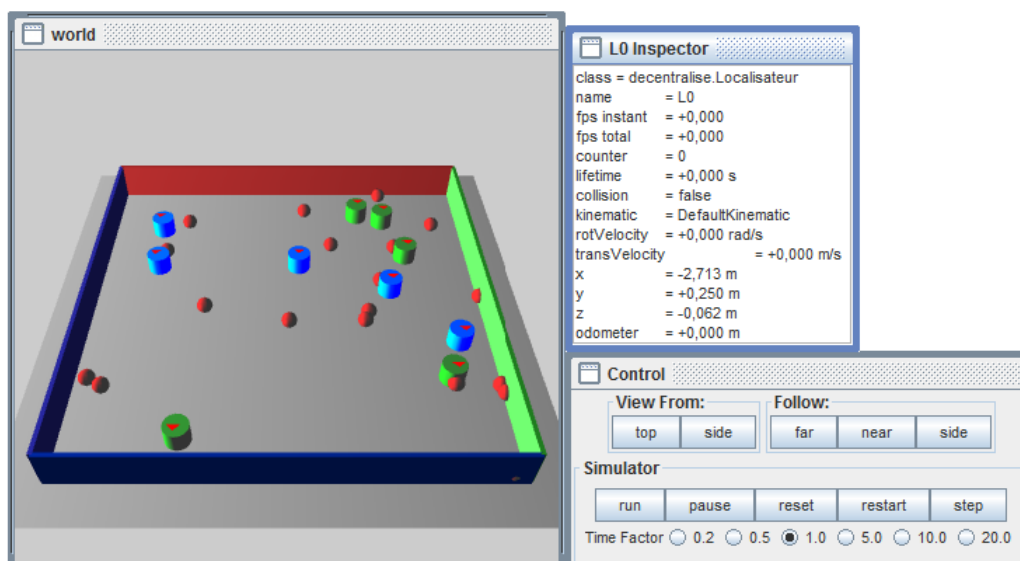


Figure 31 : l'interface de simulation décentralisée

- **Interface de simulation hybride:** La figure V.2 représente l'interface de simulation hybride

7: le coordinateur principal, 8 est le coordinateur de groupe.

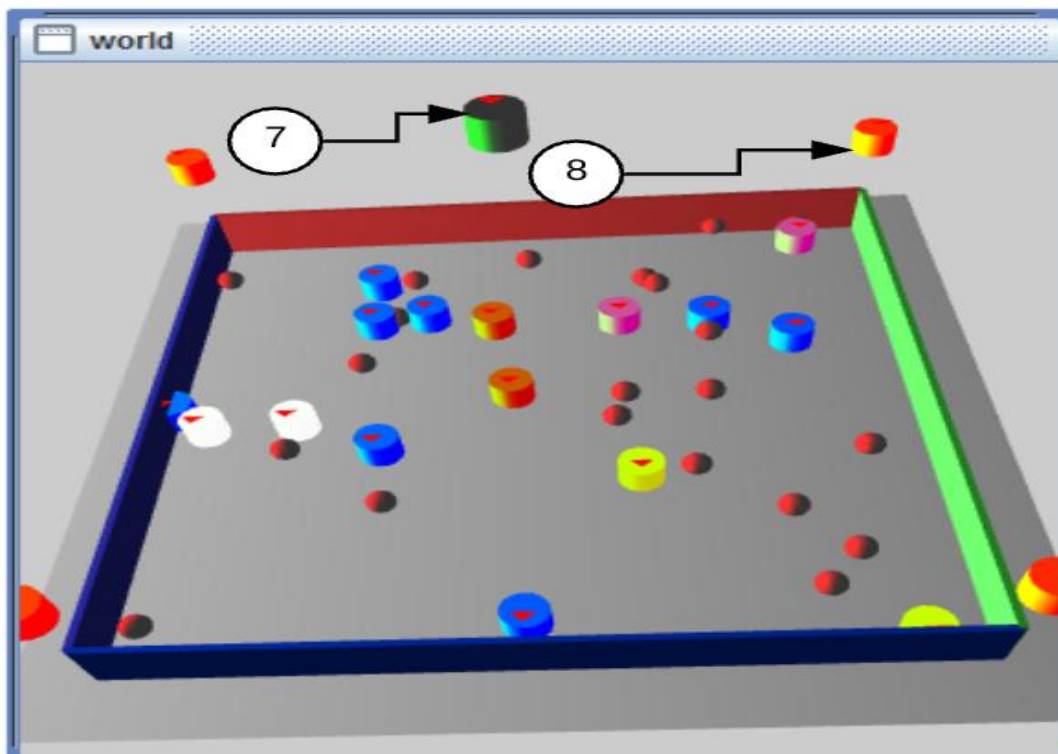


Figure 32 : l'interface de simulation hybride

Interface d'affichage des résultats:

Simulation 1 :

La figure suivante affiche les informations sur une simulation lancée précédemment pour une configuration de quatre robots localisateurs, trois robots démineurs et vingt cinq mines avec un temps de simulation fixé de deux minutes.

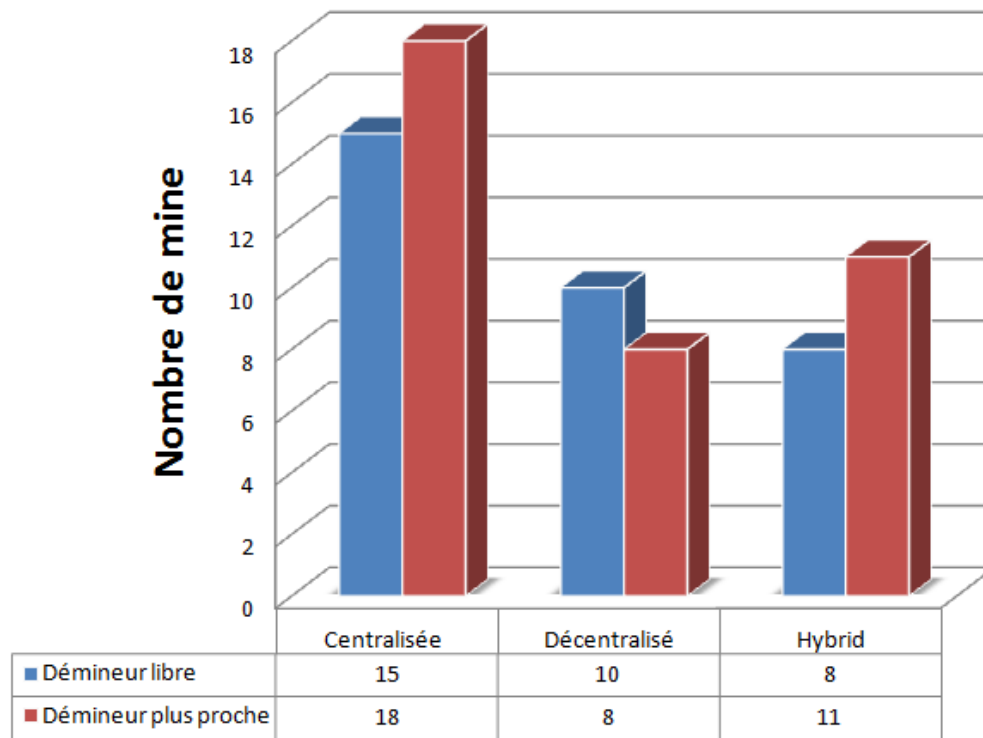


Figure 33 : L'histogramme des Résultats de simulation 1

Dans les l'histogramme, on remarque la l'approche centralisée est mieux que les deux autre approches pour cette configuration et ceci peut être expliqué par le cout de communication (échange de messages entre le localisateur et le démineur) dans la stratégie décentralisée et hybride.

Simulation 2 :

La figure suivante affiche les informations sur la simulation lancée précédemment pour une configuration de deux robots localisateurs, deux robots démineur et vingt cinq mines avec temps de simulation de deux minutes.

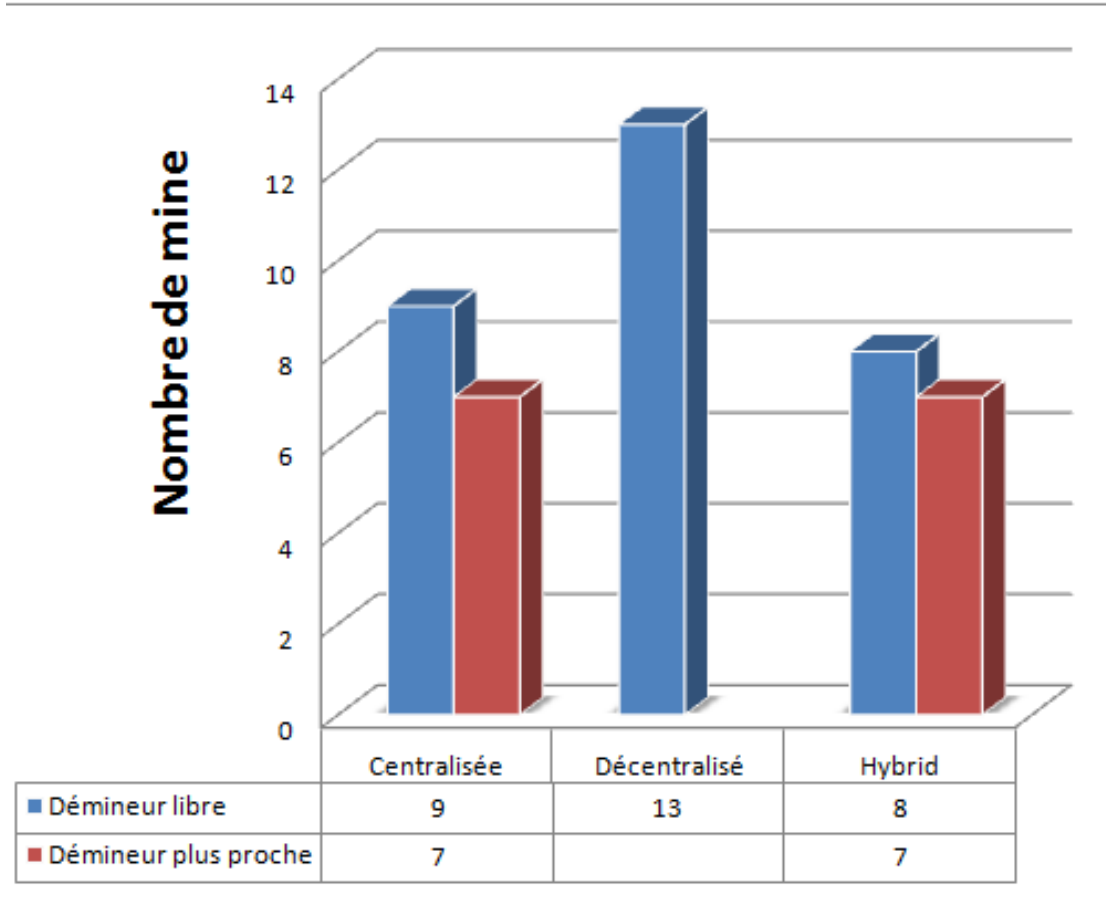


Figure 34 : L'histogramme des Résultats de simulation 2

On voit que la stratégie décentralisé est la meilleure et ceci peut être expliqué par le nombre limité de robots qui constitue le système et garantit un cout de communication plus bas que dans la simulation 1.

Simulation 3 :

La figure affiche les informations sur la simulation lancée précédemment pour une configuration de quatre robots localisateurs, quatre robots démineur et vingt cinq mines avec temps de simulation de deux minutes.

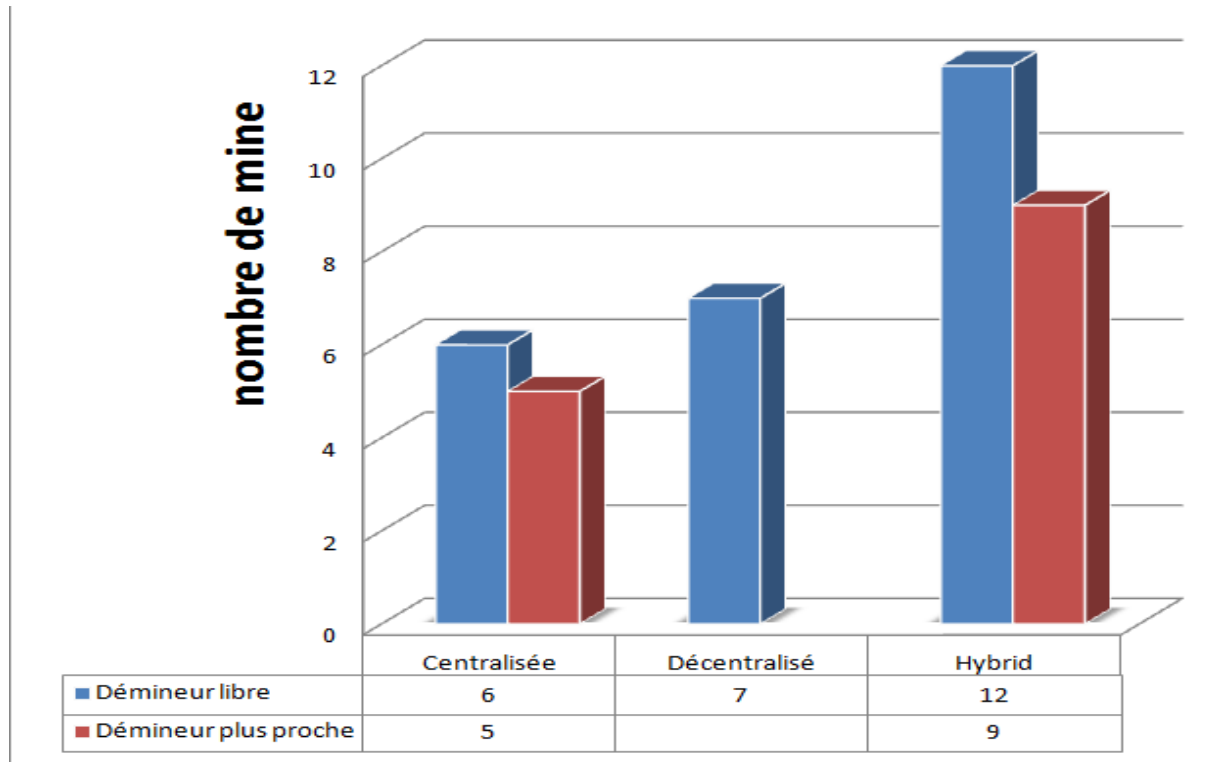


Figure 35 : L'histogramme des Résultats de simulation 3

Conclusion :

Nous avons présenté dans ce chapitre les technologies de développement qui nous ont servi durant le long du cycle de développement de notre application. Nous avons aussi présenté les interfaces de cette dernière, qui sont simples et faciles à utiliser.

Conclusion générale

L'objectif de ce mémoire est la simulation d'un système de navigation des robots mobiles dans un environnement dynamique et complexe. Il s'inscrit dans le contexte de la coopération du monde multi-robots. Cette dernière est une phase essentielle pour que le robot puisse réaliser sa tâche facilement.

La robotique, grâce à ses multiples avantages, est introduite de nos jours dans différents domaines. Les robots ont facilité la vie quotidienne des humains, en leur économisant les efforts et les risques vu qu'ils les remplacent dans la réalisation de tâches complexes dans des environnements dangereux et inaccessibles. C'est dans ce contexte que nous avons effectué une comparaison entre les stratégies de coopération.

L'utilisation du simulateur Simbad nous a permis de nous familiariser avec la technologie 3D, cependant, ce simulateur reste limité et très peu documenté, et aussi pas assez riche quand il s'agit de la simulation d'environnements complexes vu qu'il a un intérêt éducatif.

Le travail réalisé dans ce mémoire peut s'intégrer dans le secteur militaire.

Figure I	Cycle de vie d'un agent.....	05
Figure II	Un agent et son environnement.....	08
Figure III	Architecture BDI.....	09
Figure 4	Archetecture multi –niveau.....	09
Figure 5	Un SMA et son environnement.....	12
Figure 6	Composition d'un SMA.....	13
Figure 7	Représentation d'un agent en interaction avec son environnement.....	16
Figure 8	Applications des SMA.....	19
Figure 9	Les tortues de Bristol par William Grey Walter.....	22
Figure 10	Un bras manipulateur.....	26
Figure 11	Effecteur finale d'un robot.....	26
Figure 12	Différents actionneurs d'un robot.....	26
Figure 13	Différents capteurs d'un robot.....	27
Figure 14	Un contrôleur utilisé pour commander un robot.....	27
Figure 15	Le cerveau du robot.....	28
Figure 16	Architecture des robots.....	29
Figure 17	démineur humaine.....	33
Figure 18	Architectures centralisées et décentralisées.....	27
Figure 19	organigramme générale de robot de la navigation.....	46
Figure 20	Architecture Centralisée.....	47
Figure 21	Organigramme de robot localisateur.....	48
Figure 22	Organigramme de robot coordinateur.....	49
Figure 23	Diagramme de séquence de la stratégie centralisée.....	50
Figure 24	Robot localisateur et démineur.....	50
Figure 25	Organigramme de robot localisateur.....	51
Figure 26	Diagramme de séquence de la stratégie Décentralisée.....	51
Figure 27	Diagramme de séquence de la stratégie hybride.....	53
Figure 28	Architecture de notre système.....	56
Figure 29	Configuration des paramètres de lancement.....	57
Figure 30	Interface de stratégie centralisée.....	58
Figure 31	l'interface de simulation décentralisée.....	58
Figure 32	l'interface de simulation hybride.....	59
Figure 33	L'histogramme des Résultats de simulation 1.....	60
Figure 34	L'histogramme des Résultats de simulation 2.....	61
Figure 35	L'histogramme des Résultats de simulation 3.....	62

Bibliographie

- [FERB 95] : FERBER J. Les systemes Multi-Agent- Vers une Intelligence Collective, InterEditions, 1995.
- Gleizes.M.P., (2004). Vers la résolution de problèmes par émergence. Thèse d’habilitation de l’Université de Toulouse.
- Ferber.J. (2006). Concepts et méthodologies multi-agents. Dans Modélisation et simulation multi-agents, applications pour les sciences de l’homme et de la société. Edition Lavoisier.
- Rejeb.L. (2005). Simulation multi-agents de modèles économiques Vers des systèmes multi-agents adaptatifs. Thèse de doctorat de l’Université de Reims.
- Panait.L., Luke.S. (2005). Cooperative Multi-Agent Learning: The State of the Art. In Autonomous Agents and Multi-Agent Systems, 00, pp 1–48, Springer.
- Bongard Josh C. (2000). Thelegionsystem:A novel approach to evolving heterogeneity for collective problem solving».In: Genetic Programming. Springer, p.16–28(cf.p. 73).
- [Mr CHADLI -c1] : cours 2 SMA Mr CHADLI Abdelhafid . Disponible sur : <http://fmi.univ-tiaret.dz/index.php/13/cours/124/1245>

- [Noreils 93] F. R. Noreils. Toward a robot architecture integrating cooperation between mobile robots : application to indoor environment. International Journal of Robotics Research, vol. 12, pp. 79–98, 1993.
- [Adouane 05] :L. Adouane. Architectures de Contrôle Comportementales et Réactives pour la Coopération d’un Groupe de Robots Mobiles Architecture de contrôle hybride pour systèmes multi-robots mobiles. Thèse de doctorat, Laboratoire d’Automatique de Besançon (UMR CNRS 5696), 2005.
- [Khoshnevis 98] B. Khoshnevis & G. Bekey. Centralized sensing and control of multiple mobile robots. Computers & Industrial Engineering, vol. 35, pp. 503–506, 1998.
- [Webster 81] Webster. Webster’s third New International Dictionary. Encyclopaedia Britannica Inc., Chicago, 1981.
- [SIM] : <http://simbad.sourceforge.net/>
- [J3D] : Présentation Java 3D, Jean-Marc Farinone, Jean-Marc Le Gallic, Alexandre Topol. École Nationale des Sciences Géographiques, Marne la Vallée,France.
- [J3D2] : Cours Java 3D, concepts de base pour licence professionnelle ‘métiers de l’Informatique’. Patrick Reignier, Université Joseph Fourier 2011.France
- Weyns, Danny, Parunak, H, Van Dyke, Michel, Fabien, Holvoet, Tom et Ferber, Jacques(2005). « Environments for multiagents systems state-of-the-art and research challenges

- » In Environments for multi-agents systems, Springer, p.1-47(cf.p.69).
- Russell, Stuart Jonathan et Norvig, Peter(2010). Artificial intelligence: a modern approach, 3rd edition, Prentice Hall, Person Education Inc (cf.p.22, 70).
 - Demazeau, Yves (1995). «From interactions to collective behaviour in agent-based systems ». In : Proceedings of the 1st European Conference on Cognitive Science. Citeseer (cf.p.70).
 - **Wooldridge, Michael, et Jennings, Nicholas R. (1995).** « Intelligent agents : Theory and practice ».In : Knowledge engineering review10.2,p. 115-152 (cf.p.65).
 - **Ferber, Jacques (1999).***Multi agents systems : an introduction to distributed artificial Intelligence.*Addison Wesley Reading (cf.p.65).
 - Andre, David et Teller, Astro(1999). «Evolving team darwin united».In: RoboCup-98: Robot soccer world cup II. Springer, p.346–351(cf.p. 73).
 - <https://www.jmdoudoux.fr/java/dej/chap-bibliotheques-free.htm>
 - <https://hal.archives-ouvertes.fr/hal-01253056/document>
 - [SIM] : <http://simbad.sourceforge.net/>
 - [J3D] : Présentation Java 3D, Jean-Marc Farinone, Jean-Marc Le Gallic, Alexandre Topol. École Nationale des Sciences Géographiques, Marne la Vallée,France.
 - [J3D2] : Cours Java 3D, concepts de base pour licence professionnelle ‘métiers de l’Informatique’. Patrick Reignier, Université Joseph Fourier 2011.France
 - [DROCOURT, 2002] DROCOURT Cyril, “Localisation et modélisation de l’environnement d’un robot mobile par coopération de deux capteurs omnidirectionnels“, Thèse de doctorat 2002, Université de Technologie de Compiègne.
 - [MORETTE, 2009] MORETTE Nicolas, "Contribution à la Navigation de robots mobiles : approche par modèle direct et commande prédictive", Thèse de doctorat, 2009 , UNIVERSITÉ D’ORLÉANS.