

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abderrahmane Mira de Bejaia



Faculté de Technologie

Département de Génie Electrique

MEMOIRE DE MASTER

En vue de l'obtention du diplôme de Master en

Electronique

Option : Automatique

Présenté par :

ZIANE M^{ed} Lamine

Thème

Navigation flou d'un robot mobile

Soutenu le : 17/ 06 /2013, devant le Jury composé de :

M^{er}.H.LEHOUCHE

Univ. de Béjaia

Examineur

M^{me}.S.MEZZAH

Univ. de Béjaia

Examineur

M^{er}.B.MENDIL

Univ. de Béjaia

Rapporteur

Année Universitaire : 2012/2013

Remerciements

Ce présent travail est réalisé sous la direction du Pr. B. MENDIL, qu'il trouve ici ma profonde reconnaissance et mes sincères remerciements.

Je remercie les différents membres du jury qui m'ont fait l'honneur de juger ce travail.

Je voudrai adresser ma gratitude à ma famille, surtout mes très chers parents, mes sœurs et mon frère pour leurs soutien et encouragements.

Dédicaces

A mes parents.

A mes frères et sœurs et leurs enfants.

A mes tantes et oncles et leurs enfants.

A toute ma famille.

A tous mes amis.

M^{ed} lamine

Sommaire

Introduction générale	1
Chapitre I : Généralités sur les robots mobiles	4
I.1 Introduction.....	4
I.2 Présentation des robots mobiles.....	5
I.2.1 Définition d'un robot mobile.....	5
I.2.2 Aperçu historique.....	6
I.2.3 Domaines d'application de la robotique mobile.....	9
I.2.4 Types des robots.....	11
I.3 Architecture d'un robot mobile.....	12
I.3.1 La locomotion.....	12
I.3.1.1 La structure mécanique et motricité	13
I.3.1.2 La motricité et l'énergie.....	15
I.3.2 La perception.....	16
I.3.2.1 Les types de capteurs.....	16
I.3.2.2 Autonomie d'un robot mobile.....	19
I.3.3 La Décision.....	20
I.4 Conclusion.....	20
CHAPITRE II : Navigation d'un robot mobile	21
II.1 Introduction.....	21

II.2 Approches de navigation.....	21
II.2.1 Approche délibérative	21
II.2.2 Approche réactive	22
II.2.3 Approche hybride	22
II.3 Méthodes de Planification du Chemin.....	23
II.3.1 Espace des configurations.....	23
II.3.2 Méthode de décomposition en cellules de l'espace de recherche.....	23
II.3.3 Chemins pré-calculés (Roadmaps).....	25
II.3.4 Méthode du potentiel.....	26
II.4 Navigation flou du robot mobile logique	28
II.4.1 Aperçu sur la logique floue.....	28
II.4.2 Problématique de navigation floue d'un robot mobile.....	31
II.5 Conclusion.....	31
Chapitre III : Navigation dans un environnement sans obstacles	32
III.1 Introduction.....	32
III.2 Modélisation Cinématique du robot mobile.....	33
III.2.1 Posture d'un robot.....	33
III.2.2 Roulement sans glissement.....	34
III.2.3 Centre instantané de rotation.....	36
III.2.4. Robots mobiles de type unicycle.....	37
III.3 Navigation Avec retour d'état.....	39
III.3.1 Commande du robot par retour d'état.....	39
III.3.2 Résultats de simulation.....	41

III.4 Navigation en utilisant la logique floue.....	44
III.4.1 Commande du robot par la logique floue.....	44
III.4.2 Présentation schématique d'un robot mobile.....	45
III.4.3 Contrôleur flou.....	46
III.4.3.1 Définition des fonctions d'appartenance.....	46
III.4.3.2 Base de règles.....	47
III.4.4 Résultats de simulation.....	48
III.5 Conclusion.....	49
Chapitre IV : Navigation dans un environnement avec obstacles.....	50
IV.1 Introduction.....	50
IV.2 Le contrôleur d'évitement d'obstacles.....	51
IV.2.1 Perception et détection des obstacles.....	51
IV.2.2 Conception du navigateur réactif flou.....	52
IV.2.3 Définition des fonctions d'appartenance.....	52
IV.2.4 Base de règles.....	54
IV.3 Fusion de comportements.....	55
IV.4 Résultats de simulation.....	55
IV.5 Conclusion.....	59
Conclusion générale.....	60

Listes des figures

Figure I.1: Schéma des interactions d'un robot avec son environnement.....	6
Figure I.2: La tortue de Grey Walter	6
Figure I.3: A gauche : Robot "Beast" de l'université John Hopkins dans les Années 1960. A droite : Le robot Shakey de Stanford en 1969	7
Figure I.4: Genghis, développé par Rodney Brooks	8
Figure I.5: Exemples de robots utilisés dans différentes applications.....	10
Figure I.6: L'architecture fonctionnelle du robot mobile.....	12
Figure I.7 : Robot à roue.....	13
Figure I.8 : Robot à chenille.....	14
Figure I.9 : Robot marcheur.....	14
Figure I.10 : Principe d'un capteur ultrasonore.....	17
Figure I.11 : Principe d'un capteur infrarouge.....	18
Figure I.12 : Télémétrie optique par triangulation.....	19
Figure II.1 : Méthodes de navigation.....	21
Figure II.2 : Schéma général d'une architecture délibérative	22
Figure II.3 : Stratégie de fusion de comportements.....	22
Figure II.4 : Reconstruction d'un obstacle pour transformer le problème de planification...	23
Figure II.5 : Décomposition exacte de l'espace libre	24
Figure II.6: Décomposition exacte et méthodes de planification.....	24
Figure II.7 : Décomposition exacte et méthodes de décomposition.....	25

Figure II.8: Diagramme de Voronoï.....	25
Figure II.9: Construction d'un graphe de visibilité.....	26
Figure II.10 : Carte des potentiels attractif et répulsif.....	27
Figure II.11 : Configuration de base d'un contrôleur flou.....	29
Figure II.12 : Fonction d'appartenance pour les 3 cas.....	30
Figure III.1 : Situation du robot par rapport au repère de base.....	33
Figure III.2 : Caractérisation du roulement sans glissement.....	35
Figure III.3 : Centre instantané de rotation d'un robot de type unicycle.....	36
Figure III.4: Schéma du système de commande.....	39
Figure III.5: Distance et orientation du robot vers la cible avec retour d'état.....	44
Figure III.6 : navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 1 ^{er} exemple.....	41
Figure III.7: navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 2 ^{eme} exemple.....	42
Figure III.8 : navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 3 ^{eme} exemple.....	42
Figure III.9 : navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 4 ^{eme} exemple.....	43
Figure III.10 : Distance et orientation initiales du robot par rapport à la cible.....	44
Figure III.11: Schéma du système de commande.....	45
Figure III.12: Schéma représentatif d'un robot mobile.....	45
Figure III.13 : les fonctions d'appartenance des variables d'entrée du contrôleur « convergence vers la cible »	46
Figure III.14 : Les fonctions d'appartenance des variables de sortie du contrôleur.....	47

Figure III.15 : navigation du robot en utilisant la logique floue Dans un environnement sans obstacle: 1 ^{er} exemple.....	48
Figure III.16 : navigation du robot en utilisant la logique floue dans un environnement sans obstacle: 2 ^{eme} exemple.....	49
Figure IV.1 : Répartition des capteurs du robot.....	51
Figure IV.2 : Schéma de contrôle utilisé.....	52
Figure IV.3 : les fonctions d'appartenances des variables d'entrée du contrôleur « évitement d'obstacles ».....	53
Figure IV.4 : Les fonctions d'appartenance des variables de sortie du contrôleur d'évitement d'obstacles.....	53
Figure IV.5 : navigation du robot dans un environnement avec obstacle : 1 ^{er} exemple.....	56
Figure IV.6 : Navigation du robot dans un environnement avec obstacle : 2 ^{eme} exemple.....	57
Figure IV.7 : navigation du robot dans un environnement avec obstacle : 3 ^{eme} exemple.....	57
Figure IV.8 : navigation du robot dans un environnement avec obstacle : 4 ^{eme} exemple.....	58

Liste des tableaux

Tableau III.1 Base de règles floues de la vitesse d'avancement.....	47
Tableau III.2 Base de règles floues de la vitesse de braquage.....	48
Tableau IV.1 Base de règles floues de la vitesse d'avancement.....	54
Tableau IV.2 Base de règles floues de la vitesse de braquage.....	55

Introduction Générale

L'homme a toujours cherché à concevoir des entités à l'image de l'être humain sur lesquelles il pourrait exercer son autorité, exécutant ainsi ses moindres désirs. Les premières approches remontent à l'époque médiévale où des automates étaient placés dans les églises afin d'impressionner les paysans, fascinés par ces "êtres suprêmes". Les automates sont ensuite devenus des jouets populaires au 18ème siècle sous forme miniature. [MAL 11]

Les premiers travaux en robotique mobile en milieu industriel ne conféraient aux robots mobiles qu'une autonomie de déplacement très limitée et nécessitaient la réalisation d'importants et coûteux travaux d'infrastructure. C'était le cas, en particulier, des chariots filoguidés, limitant le déplacement du robot à des voies lui étant réservées. [SOU 09]

Depuis quelques années, un intérêt croissant est porté au sein de la communauté robotique au développement de systèmes intelligents et autonomes dans le cadre de la robotique mobile. Un tel intérêt peut être aperçu comme une conséquence à l'apparition d'applications potentielles (nettoyage, aide à la mobilité des personnes handicapées...) et au désir de mettre les robots sur des tâches nouvelles telles que l'exploration de sites inaccessibles à l'homme (planètes, fonds d'océans) ou opérer dans des milieux qui lui sont hostile (milieux radioactifs).

Dans de tels contextes, munir les systèmes robotiques d'une capacité de décision et, plus particulièrement, de planifier leurs trajectoires ou naviguer d'une manière autonome reste l'un des éléments clé dans la mise en œuvre de leur autonomie. En effet, le robot autonome doit être en mesure de prendre des décisions, quant aux mouvements à réaliser, en fonction des informations dont il dispose sur l'environnement ou lui parvenant de ses capteurs. Pour ce, il existe une variété de capteurs (caméras vidéo, télémètre, capteurs ultrasoniques ou infrarouges, etc.) et d'un système de navigation capable de gérer les différentes situations déjà citées.

A cet effet, le souci majeur est d'élaborer des techniques de navigation efficaces, telle que la sécurité soit prioritaire par rapport à l'optimalité. La stratégie de navigation peut intégrer les informations, a priori, dont on dispose sur l'environnement, sous forme d'un modèle de ce

dernier. Cependant, lorsque l'environnement devient plus complexe (partiellement connu, dynamique,...), il apparaît indispensable que le robot soit doté de capacités décisionnelles aptes à le faire réagir aux aléas qui peuvent contrarier ses mouvements.

Les architectures de contrôle en robotique mobile sont généralement classées en deux catégories principales: hiérarchiques et réactives. Les architectures hiérarchiques sont des architectures graphiques. Celles-ci génèrent une trajectoire à partir d'une carte de l'environnement de travail supposé parfaitement connu. Mais une connaissance parfaite d'un environnement réel est chose presque impossible ; sauf le cas d'un environnement dédié.

L'objectif principal de recherche sur les stratégies de la navigation réactive est la réalisation de systèmes autonomes équipés de capteurs et actionneurs de faibles coût, afin d'exécuter des tâches complexes dans des environnements incertains ou inconnus. Beaucoup d'études se sont concentrées sur l'approche comportementale dans laquelle la réactivité aux circonstances imprévisibles est accomplie par des algorithmes simples (implantant des comportements élémentaires) traitant les informations sensorielles en temps réel moyennant des inférences de haut niveau.

C'est dans cette perspective que s'inscrit notre travail. Il concerne principalement le développement d'une navigation réactive d'un robot mobile autonome dans un environnement inconnu. L'approche est basée sur la fusion de deux comportements ; à savoir, le comportement de convergence vers une cible et le comportement d'évitement d'obstacles. A cet effet, une technique de traitement de données et de prise de décisions a, tout particulièrement, retenue notre attention. Il s'agit du raisonnement approximatif à base de la logique floue.

La logique floue est souvent adoptée pour vaincre les difficultés de modélisation des environnements incertains et/ou dynamiques ; tâche difficile à exprimer par des équations mathématiques. Pour vaincre les incertitudes et rehausser la robustesse de navigation, beaucoup de recherches sont basées sur des techniques d'apprentissage automatiques qui permettent au contrôleur flou d'exploiter les données sensorielles de l'environnement exploré.

Pour une meilleure présentation de notre travail, le mémoire est organisé en quatre chapitres et structuré comme suit :

Dans le premier chapitre nous présentons les notions de base d'un robot mobile.

Dans le deuxième chapitre nous expliquons la stratégie de navigation d'un robot mobile et nous définissons quelques concepts de la logique floue.

Dans le troisième chapitre se trouve la première partie qui est réalisée avec retour d'état. La seconde partie est consacrée à la conception d'un contrôleur flou de convergence vers une cible. Celui-ci n'opère, bien sûr, que dans un environnement dénué d'obstacles.

Le quatrième chapitre comprend la simulation d'une architecture réactive comprenant deux comportements essentiels « convergence vers une cible » et « évitement d'obstacles, avec deux contrôleurs flous.

Chapitre I : Généralités sur les robots mobiles

I.1 Introduction :

L'objectif de ce chapitre, est de donner un aperçu sur le robot mobile. Après la présentation des différents types de robots mobiles et les contraintes du terrain sur lequel ils sont conçus pour évoluer, nous aborderons les contraintes principales liées à leur cinématique et les solutions développées pour y remédier. Enfin nous étudierons les outils permettant aux robots de percevoir leur environnement et de s'y repérer, étape primordiale nécessaire à l'autonomie totale des robots mobiles.

Cet état de l'art offre une vision non exhaustive des thématiques associées au domaine de la robotique mobile et présente l'ensemble des verrous scientifiques qu'il reste à lever pour aboutir au développement d'un robot autonome. Parmi ceux-ci, nous nous focaliserons alors sur celui de la navigation d'un robot mobile.

I.2 Présentation des robots mobiles :

Les robots mobiles sont en constante évolution principalement depuis les années 2000, dans les domaines militaires (drones volants, robots sous-marins, . . .), le médical, l'agriculture, etc. Ils sont particulièrement demandés pour réaliser des tâches considérées comme pénibles, dangereuses ou impossibles pour l'homme. C'est le cas, par exemple, des opérations de déminage, de recherche de boîtes noires d'avions de ligne au fond de l'océan ou d'exploration de planètes. Les satellites artificiels, les lanceurs (comme Ariane V), les métros sans conducteurs, les ascenseurs sont des exemples de robots mobiles qui nous entourent. Les avions de lignes, les trains et les voitures évoluent de façon continue vers des systèmes de plus en plus autonomes et deviendront très probablement des robots mobiles dans les prochaines décennies.

La robotique mobile est la discipline qui s'intéresse à la conception de robots mobiles. Elle s'appuie sur d'autres disciplines comme l'automatique, le traitement du signal, la mécanique, l'informatique et l'électronique. L'objectif de ce chapitre est de donner un aperçu des outils et des méthodes de la robotique qui permettront d'aider à la conception de robots mobiles.

I.2.1 Définition d'un robot mobile :

Il existe diverses définitions du terme robot, mais elles tournent en général autour de celle-ci : Un robot est une machine équipée de capacités de perception, de décision et d'action qui lui permettent d'agir de manière autonome dans son environnement en fonction de la perception qu'il en a. [FIL 09]

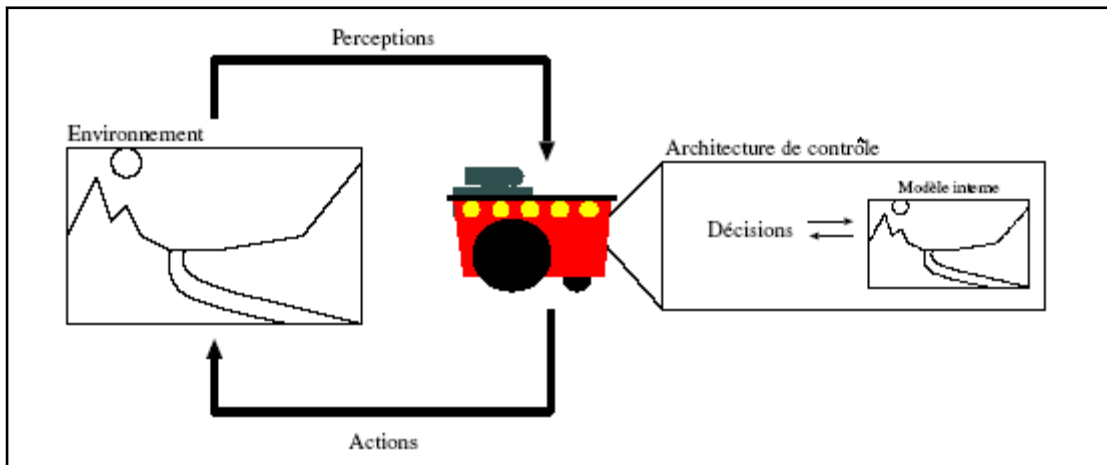


Figure I.1: Schéma des interactions d'un robot avec son environnement. Le modèle interne de l'environnement est optionnel. Un robot peut fonctionner sans en utiliser.

Cette définition s'illustre par un schéma classique des interactions d'un robot avec son environnement (Figure I.1). Les différentes notions que nous présenterons sont essentiellement issues de cette vision de la robotique, donnée par l'Intelligence Artificielle, qui place au centre des préoccupations l'enchaînement de ce cycle Perception/Décision/Action. La manière dont un robot gère ces différents éléments est définie par son architecture de contrôle qui peut éventuellement faire appel à un modèle interne de l'environnement pour lui permettre alors de planifier ses actions à long terme. [FIL 09]

I.2.2 Aperçu historique :

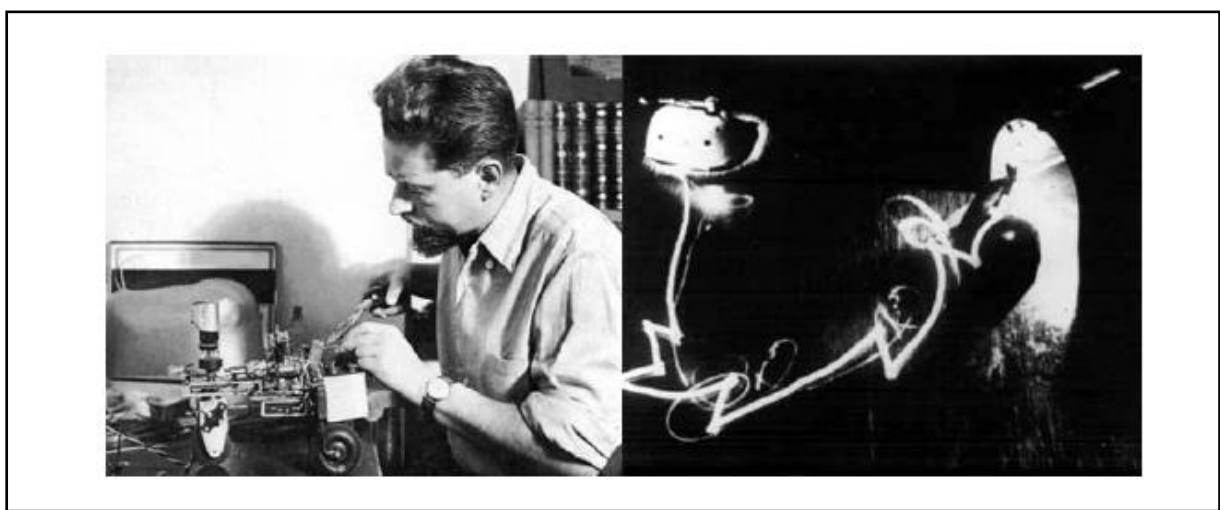


Figure I.2: La tortue de Grey Walter (nommée "machina speculatrix" et surnommée Elsie) et une illustration de sa trajectoire pour rejoindre sa niche.

Le terme de robot apparaît pour la première fois dans une pièce de Karel Capek en 1920 : Rossum's Universal Robots. Il vient du tchèque 'robot' (servitude) et présente une vision des robots comme serviteurs dociles et efficaces pour réaliser les tâches pénibles mais qui déjà vont se rebeller contre leurs créateurs.

La Tortue construite par Grey Walter dans les années 1950 (Figure I.2), est l'un des tout premiers robots mobiles autonomes. GreyWalter n'utilise que quelques composants analogiques, dont des tubes à vide, mais son robot est capable de se diriger vers une lumière qui marque un but, de s'arrêter face à des obstacles et de recharger ses batteries lorsqu'il arrive dans sa niche. [MEN 12]

Toutes ces fonctions sont réalisées dans un environnement entièrement préparé, mais restent des fonctions de base qui sont toujours sujets de recherche pour les rendre de plus en plus génériques. [MEN 12]

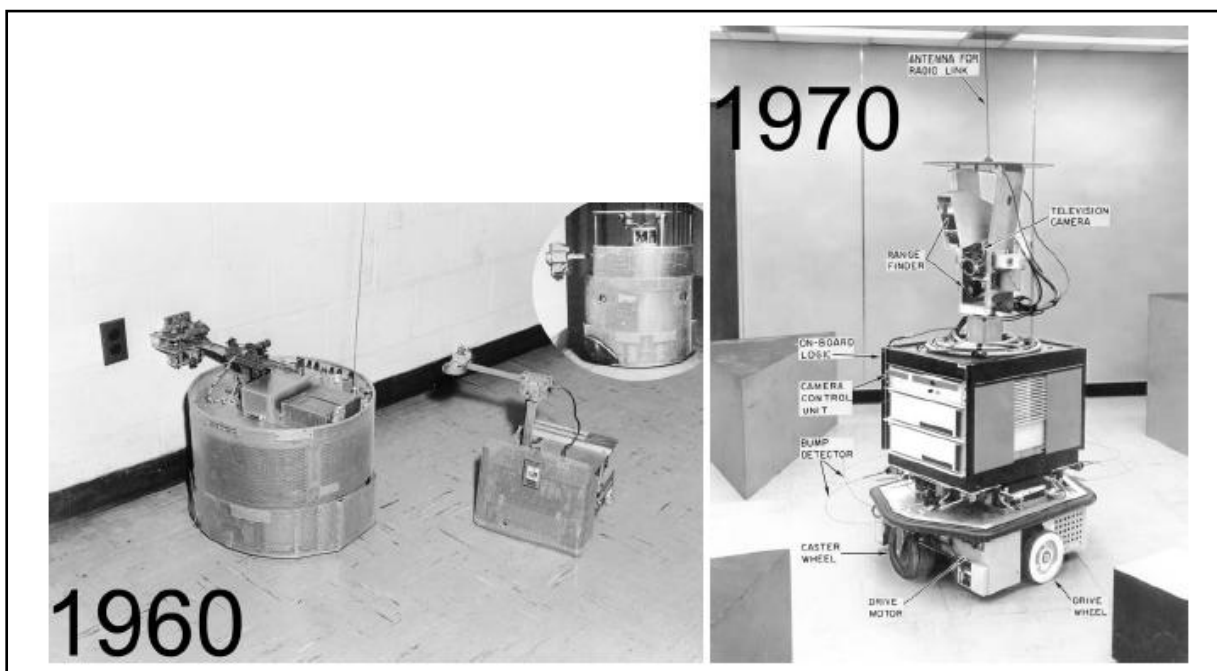


Figure I.3: A gauche : Robot "Beast" de l'université John Hopkins dans les Années 1960. A droite : Le robot Shakey de Stanford en 1969 a été une plate-Forme de démonstration des recherches en intelligence artificielle.

Dans les années 60, les recherches en électronique vont conduire, avec l'apparition du transistor, à des robots plus complexes mais qui vont réaliser des tâches similaires. Ainsi le

robot "Beast" (**Figure I.3**) de l'université John Hopkins est capable de se déplacer au centre des couloirs en utilisant des capteurs ultrason, de chercher des prises électriques (noires sur des murs blancs) en utilisant des photo-diodes et de s'y recharger. [MEN 12]

Les premiers liens entre la recherche en intelligence artificielle et la robotique apparaissent à Stanford en 1969 avec Shakey (**Figure I.3**). Ce robot utilise des télémètres à ultrason et une caméra et sert de plate-forme pour la recherche en intelligence artificielle, qui à l'époque travaille essentiellement sur des approches symboliques de la planification. La perception de l'environnement, qui à l'époque est considérée comme un problème séparé, voire secondaire, se révèle particulièrement complexe et conduit là aussi à de fortes contraintes sur l'environnement. Ces développements se poursuivent avec le Stanford Cart dans les années 1980, avec notamment les premières utilisations de la stéréovision pour la détection d'obstacles et la modélisation de l'environnement. [MEN 12]

Une étape importante est à signaler au début des années 1990 avec l'apparition de la robotique réactive, représentée notamment par Rodney Brooks. Cette nouvelle approche de la robotique, qui met la perception au centre de la problématique, a permis de passer de gros robots très lents à de petits robots (**Figure I.4**), beaucoup plus réactifs et adaptés à leur environnement. [MEN 12]



Figure I.4: Genghis, développé par Rodney Brooks au MIT au début des années 1990.

I.2.3 Domaines d'application de la robotique mobile :

Aujourd'hui, le marché commercial de la robotique mobile est toujours relativement restreint, mais il existe de nombreuses perspectives de développement qui en feront probablement un domaine important dans le futur. Les applications des robots peuvent se trouver dans de nombreuses activités "ennuyeuses, salissantes ou dangereuses" (3 D's en anglais pour Dull, Dirty, Dangerous), mais également pour des applications ludiques ou de service, comme l'assistance aux personnes âgées ou handicapées. [BOR 98]

Commençons par les trois domaines majeurs: le nucléaire, le spatial et le sous marin qui présentent des points communs: [BOR 98]

- La nécessité d'adaptation au milieu est primordiale
- La tâche à effectuer est non répétitive.
- En spatial, comme en nucléaire et en sous marin, la difficulté de communication nécessite l'autonomie du robot.

a) Le nucléaire : C'est le domaine le plus ancien, le nucléaire reste un grand utilisateur pour ses applications classiques: décontamination, inspection et éventuellement intervention en cas d'accidents. Comme on peut le constater au Service de Télé opération et Robotique du CEA (Commissariat à l'énergie atomique) Fontenay-aux-Roses (France). C'est certainement le domaine où les architectures utilisées sont les plus variées, car chaque problème trouve sa solution spécifique: robots longs et fins pour passer dans des canalisations, hexapodes, véhicules à chenilles, à roues, en trains...

b) Le spatial : La robotique spatiale a vraiment progressé dans les années 80. Après des fortunes diverses dues en partie aux pressions des astronautes pour favoriser les vols habités, la robotique mobile est revenue en force avec l'exploration martienne et le succès très médiatisé du petit robot Séjourner qui est un véhicule à six roues déposé le 4 juillet 1997 chargé d'explorer le site.

c) **Le sous-marin** : En sous-marin, la variété des applications s'étend de la recherche océanographique fondamentale à l'ingénierie. Nous citons: La cartographie des fonds sous marins, l'inspection des pavés comme l'exploration du TITANIC, l'observation de la flore, de la faune, de la géologie. Comme en nucléaire les systèmes utilisés diffèrent selon les applications et la profondeur : Engins télé opérés, véhicules autonomes de longue portée.

d) **Médecine** : Le monde de la santé est l'un des principaux utilisateurs potentiels de techniques de la robotique mobile; dans des hôpitaux ou des maisons de retraite, des prototypes de robots mobiles circulant dans les couloirs et les chambres pour livrer des médicaments et plateau-repas.

e) **Les véhicules automatiques** : Les recherches en assistance à la conduite des automobiles ou des camions, incluant même une automatisation partielle de celle-ci, utilisent largement les résultats des travaux sur les robots mobiles en localisation, évitement d'obstacles, planification des mouvements. Plus spécifique est le développement assez récent des études sur les 'drones', petits véhicules volants (dirigeables, avions hélicoptères), partiellement télé opérés ou autonomes. Outre que les applications militaires de surveillance ou de renseignement, il existe de nombreux besoins civils qui pourront être satisfait par de tels systèmes: cartographie automatique, inspection de lignes de haute tension, localisation d'accidentés en montagnes, sécurité, ...etc. On retrouve dans ce domaine les problèmes de localisation et de perception de l'environnement des robots mobiles terrestres, auxquels s'ajoutent des questions délicates de pilotage. A cela, s'ajoute à l'heure actuelle des nombreuses plates-formes conçues essentiellement pour les laboratoires de recherche. La figure I.5 montre quelques exemples de robot réels.



Figure I.5: Exemples de robots utilisés dans différentes applications.

I.2.4 Types des robots :

On distingue deux groupes de robots, les robots fixes et les robots mobiles. Les robots fixes sont utilisés dans l'industrie pour réaliser des tâches dangereuses (soudure du châssis ou peinture de la carrosserie dans une usine automobile), les robots mobiles pour transporter des tâches (depuis les chaînes de fabrication jusqu'aux zones de stockage) ou encore pour transporter le courrier dans les bureaux. [GIA 00]

La tâche essentielle qui incombe aux robots mobiles est le déplacement dans un environnement connu ou inconnu. Il est donc nécessaire de connaître la position du robot au sein de son univers de façon précise ou relative selon le cas. La solution pour un robot mobile doit être globale, c'est-à-dire qu'il ne faut pas simplement rendre autonome une plate-forme en oubliant provisoirement l'outil de travail qui l'équipe. [GIA 00]

La plupart des recherches sont concentrées sur les reconnaissances de l'environnement immédiat du robot. Des capteurs (infrarouges, ultrasons, laser caméra) associés à des logiciels adaptés permettent, par la reconnaissance de formes, d'optimiser les déplacements du mobile.

I.3 Architecture d'un robot mobile :

Un robot mobile se compose de trois grands modules fonctionnels : la locomotion, la perception et la décision. [GIA 00] (**Figure I.6**)

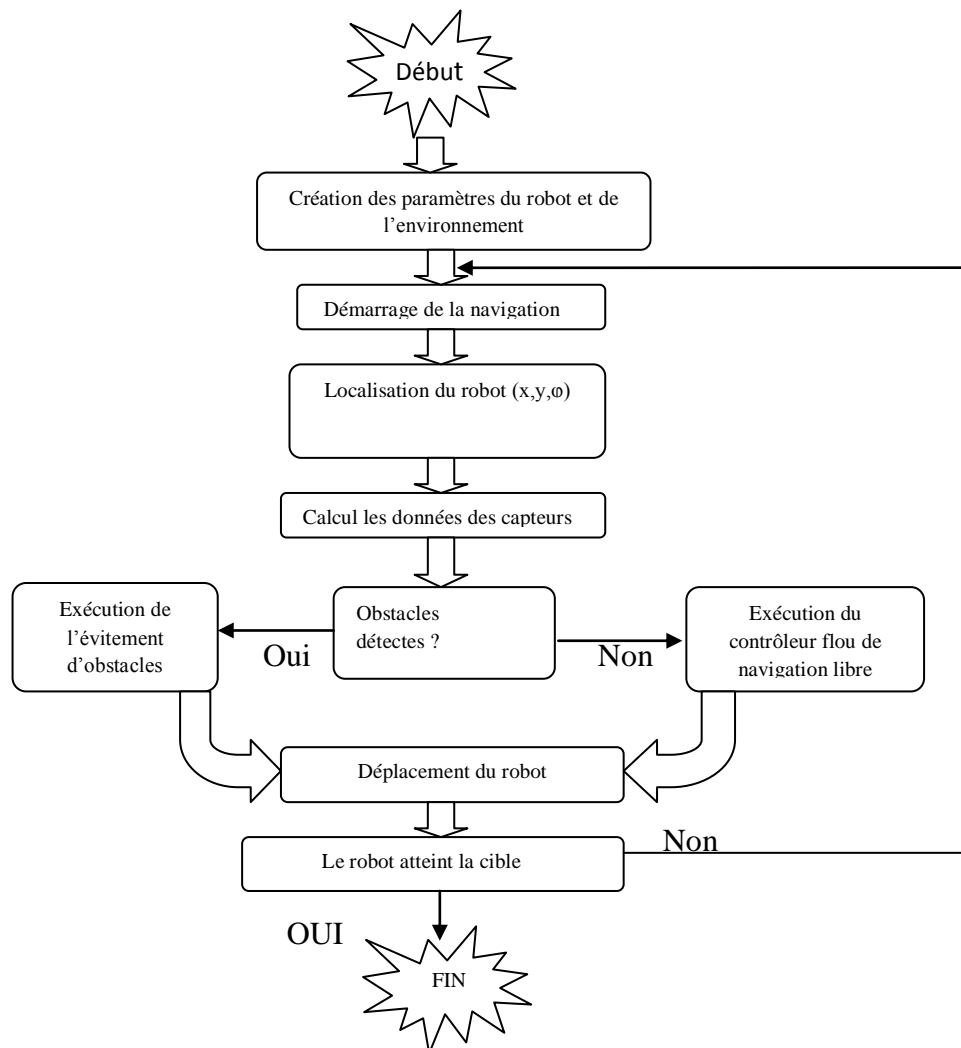


Figure I.6: L'architecture fonctionnelle du robot mobile

I.3.1 La locomotion :

La locomotion se décompose en deux parties : celle qui réalise l'appui vis-à-vis du milieu dans lequel on entend se déplacer et celle qui permet la propulsion. Celle-ci inclut les moteurs et les mécanismes conduisant au déplacement. Les moyens de déplacement sont

nombreux, ils dépendent du milieu où le mobile est censé se déplacer (mobile sur le sol, avion, sous-marin, ...). Dans notre étude nous traiterons seulement le cas des robots qui se déplacent sur une surface. [GIA 00]

1.3.1.1 La structure mécanique et motricité :

On peut dénombrer quatre types de structures mécaniques assurant la motricité. [PRU 88]

- ✓ **Les mobiles à roues :** La mobilité par roues est la structure mécanique la plus communément appliquée. Cette technique assure selon l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importante. Le franchissement d'obstacles ou l'escalade de marches d'escalier est possible dans une certaine mesure. Toutes les configurations (nombre, agencement, fonction) des roues sont appliquées.



Figure I.7 : Robot à roue

- ✓ **Les mobiles à chenilles**

L'utilisation des chenilles présente l'avantage d'une bonne adhérence au sol et d'une faculté de franchissement d'obstacles. L'utilisation est orientée vers l'emploi sur sol accidenté ou de mauvaise qualité au niveau de l'adhérence (présence de boue, herbe...).



Figure I.8 : Robot à chenille

✓ **Les mobiles marcheurs :**

Les robots mobiles marcheurs sont destinés à réaliser des tâches variées dont l'accès au site est difficile, dangereux ou impossible à l'homme. Leur autonomie à nombreux degrés de liberté permet un rapprochement avec les robots manipulateurs. La locomotion est commandée en termes de coordonnées articulaires. Les méthodes de commande des articulations définissent le concept d'allure qui assure le déplacement stable de l'ensemble. Les différentes techniques étudiées se rapprochent de la marche des animaux et notamment de celle des insectes. L'adaptation au support est un problème spécifique aux marcheurs. Il consiste à choisir le meilleur emplacement de contact en alliant l'avance et la stabilité avec l'aide de capteurs de proximité, de contact ou de vision.



Figure I.9 : Robot marcheur

✓ **Les robots rampants :**

La reptation est une solution de locomotion pour un environnement de type "tunnel" qui conduit à réaliser des structures filiformes. Le système est composé d'un ensemble de modules ayant chacun plusieurs mobilités. Les techniques utilisées découlent des méthodes de locomotion des animaux.

Le type scolopendre constitue une structure inextensible articulée selon deux axes orthogonaux.

Le type lombric comprend trois articulations, deux rotations orthogonales et une translation dans le sens du mouvement principal.

Le type péristaltique consiste à réaliser un déplacement relatif d'un module par rapport aux voisins.

1.3.1.2 La motricité et l'énergie :

Le déplacement des robots est réalisé par des moteurs de types: électrique, thermique et hydraulique. L'énergie électrique la plus fréquemment employée offre l'avantage d'une commande aisée. Par contre le transport et la généralisation présentent des difficultés. Plusieurs méthodes sont employées:

- par batteries qui sont soit rechargées périodiquement de manière automatique ou manuelle, soit par un échange avec d'autres lorsqu'elles sont déchargées.
- par groupe électrogène embarqué dont l'inconvénient constitue la masse élevée. L'énergie de base est alors thermique.
- par cordon ombilical qui réduit l'autonomie du robot. L'énergie thermique est essentiellement employée par des véhicules de forte puissance comme énergie de base pour la traction ou pour activer un compresseur hydraulique.

I.3.2 La perception :

Cette partie du robot est souvent la plus difficile à réaliser et les recherches dans ce domaine utilisent une plate-forme à trois ou quatre roues pour étudier les problèmes de comportement.

La perception passe par deux étapes successives : la lecture des capteurs, puis le traitement de l'information. L'interprétation, qui permet de délivrer un message clair à la fonction «locomotion», est dévolue à la fonction «décision» du robot. On peut imaginer la situation d'un robot mobile devant son environnement en comparant celui-ci avec un aveugle qui découvrirait un nouveau lieu. Le robot doit être capable d'analyser un objet localement, puis de positionner globalement tous les objets les uns par rapport aux autres. De cette manière, il construit un modèle de son environnement au fur et à mesure de son exploration. Les capteurs les plus simples sont utilisés pour analyse locale (les capteurs de contact, les capteurs de proximité avec ou sans mesure, les capteurs de lumière, les capteurs de pression mouvement), alors que la télémétrie laser ou ultrason autorise l'analyse d'une image simple ou d'un mouvement. [GIA 00]

I.3.2.1 Les types de capteurs :

Les capteurs jouent un rôle indispensable dans la modélisation de l'environnement. Car, ils fournissent des renseignements sur l'univers du robot. Grâce à ces informations, ce dernier peut connaître sa position et peut aussi évoluer librement en évitant les obstacles.

✓ Capteur ultrasonore :

La mesure de la distance objet-capteur se fait par une mesure temporelle entre l'instant d'émission et l'instant de réception (appelé temps de vol) de train d'ondes ultrasonores émis par un transducteur piézo-électrique. (**Figure I.10**) [BEN 97]

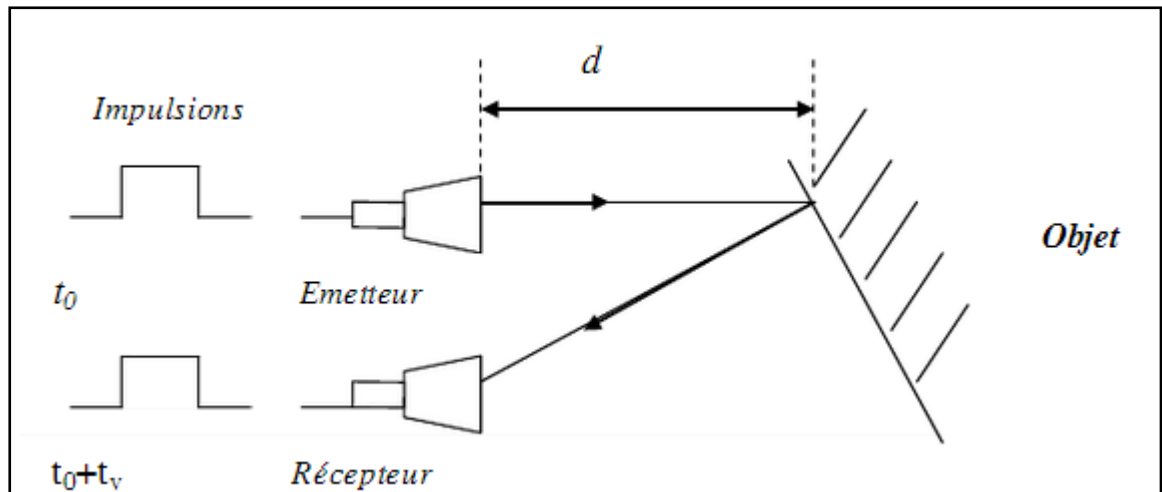


Figure I.10 : Principe d'un capteur ultrasonore

La distance d séparant l'émetteur et l'obstacle est donnée par:

$$d = \frac{c \cdot t_y}{2}$$

Avec : t_y : temps de vol . c : célérité du son.

Les avantages d'un système ultrasonore sont:

- Temps de propagation facilement mesurable.
- Une mise en œuvre simple.

Le principal inconvénient réside dans le mouvement d'air qui entraîne des fluctuations de la vitesse de propagation des ultrasons ce qui introduit des erreurs sur la mesure de la distance.

✓ **Les capteurs à infrarouge :**

Ce type de télémètre utilise les rayons infrarouges pour mesurer la distance entre l'objet et le robot. Son principe de fonctionnement est le suivant: Un émetteur (souvent une diode) est associé à un récepteur (photo transistor) qui est excité par la lumière réfléchiée par l'objet dont on souhaite évaluer la distance ou déterminer la présence. La figure I.11. [BEN 97]

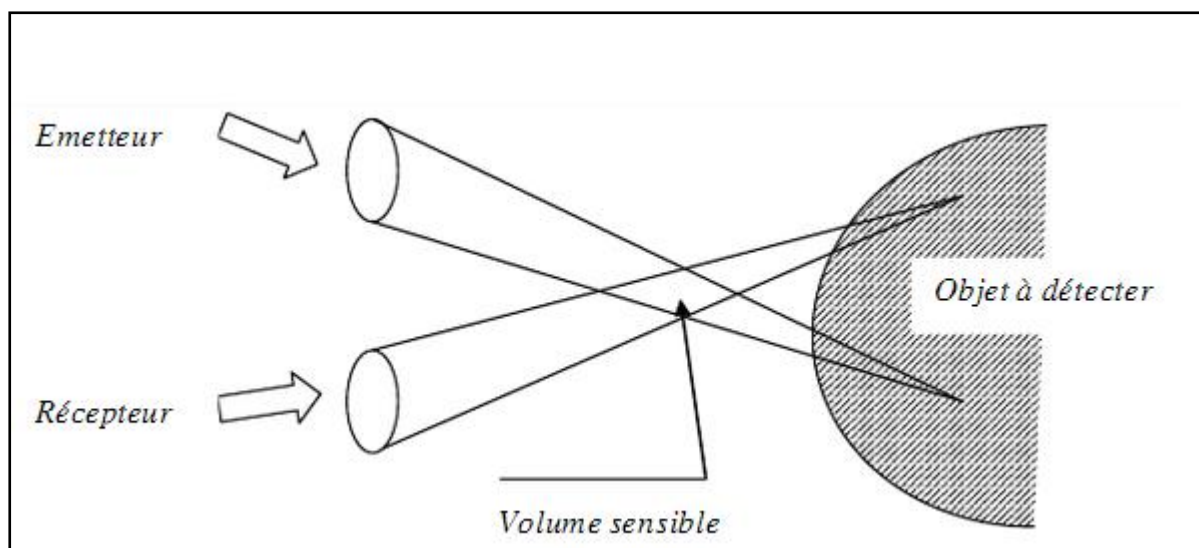


Figure I.11 : Principe d'un capteur infrarouge

Ces capteurs présentent un certain nombre d'avantages:

- Des mesures beaucoup plus précises que celles que l'on peut obtenir par les capteurs ultrasonores.
- Leurs distances minimales de détection descendent au dessous du centimètre, ils présentent la meilleure solution pour une distance minimum de la détection.
- Un coût faible et petit volume.

✓ **Les capteurs télémétriques laser :**

Avec ce type de capteurs, on peut calculer la distance à partir de différentes méthodes:

Mesure du temps du vol:

C'est-à-dire de l'intervalle de temps séparant l'émission d'un train d'impulsions et la réception de son écho. Une impulsion optique est générée sur une diode Laser recueillie en retour sur une photo détectrice. [BEN 97]

Triangulation optique:

La distance peut être aussi calculée en mesurant le déphasage entre le signal émis et le signal reçu. Ce qu'on appelle la triangulation optique. [BEN 97]

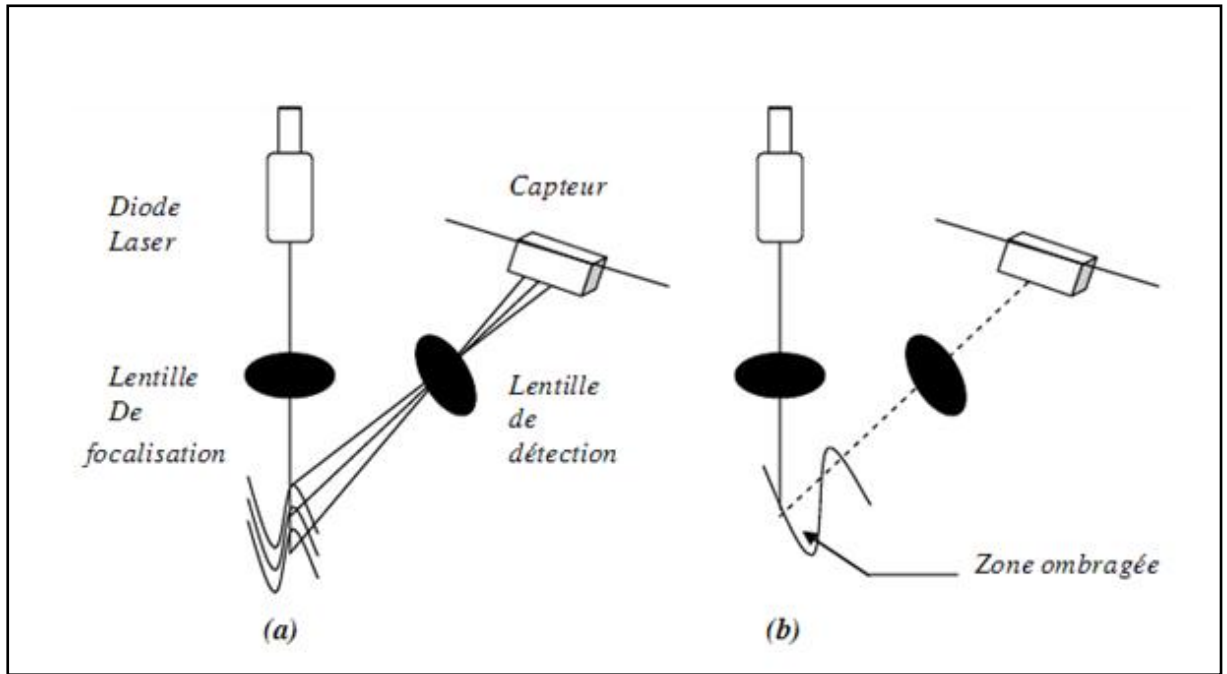


Figure I.12 : Télémétrie optique par triangulation

Une diode Laser émet un faisceau lumineux sur un point de surface, celle-ci diffuse une partie de la lumière reçue dans tout l'espace dont une partie est collectée par lentille de détection puis focalisée sur un capteur. Basée sur la mesure de l'angle entre la direction du faisceau émis et celle sous la quelle est vu le point d'impact. Cette technique ne marche pas s'il existe des zones d'ombre. [BEN 97]

I.3.2.2 Autonomie d'un robot mobile :

La principale caractéristique d'un robot mobile est l'autonomie qui est la capacité du robot mobile à prendre ses décisions et à déterminer ses actions lui-même, dans son environnement, par le biais des informations collectées par son système de perception et des connaissances incorporées à priori ou acquises par apprentissage. [CHO 98]

I.3.3 La Décision :

Les informations en provenance des différents capteurs doivent être interprétées comme autant d'éléments utiles à la prise de décision sur l'action à faire, le but étant de délivrer les ordres corrects aux actionneurs, bras, pinces ou moteurs des roues. Dans un robot mobile, il est nécessaire de donner des priorités en fonction des informations reçues. Par exemple, si un capteur de contact informe d'un choc sur l'avant, cette information est prioritaire sur un déplacement du robot vers l'avant et doit entraîner un arrêt ou un déplacement dans une autre direction.

On comprend ainsi toute la difficulté à réaliser cette fonction, car c'est elle qui donnera vie à notre robot. C'est lors de cette phase de la conception d'un robot qu'il est nécessaire de lui donner une forme d'intelligence en lui laissant le choix sur l'action à entreprendre. Cette prise de la décision est souvent arbitraire au début, mais elle permet de développer une forme d'apprentissage qui tient compte des résultats des décisions précédentes. Cette fonction est de loin la plus incomplète, car elle tente d'approcher la psychologie animale, voire humaine.

I.4 Conclusion :

Dans ce chapitre nous avons présenté : les différents types des robots, architecture d'un robot mobile (c'est-à-dire la locomotion, la perception et la décision). La robotique mobile connue une grande révolution. Elle est devenue indispensable et ces applications sont réparties dans presque dans tous les domaines de la vie. La robotique mobile va bien sûr continuer à intégrer les progrès de l'informatique et, en premier lieu, bénéficier d'algorithmes de plus en plus complexes, exécutables en temps réel.

CHAPITRE II : Navigation d'un robot mobile

II.1 Introduction:

La possibilité de naviguer avec une certaine autonomie dans un environnement contenant des obstacles est l'un des problèmes des robots mobiles. L'autonomie de navigation d'un robot mobile est la capacité des robots à prendre des décisions indépendamment de l'opérateur humain dans leurs déplacements. Il existe deux approches de navigation: délibérative et réactive.

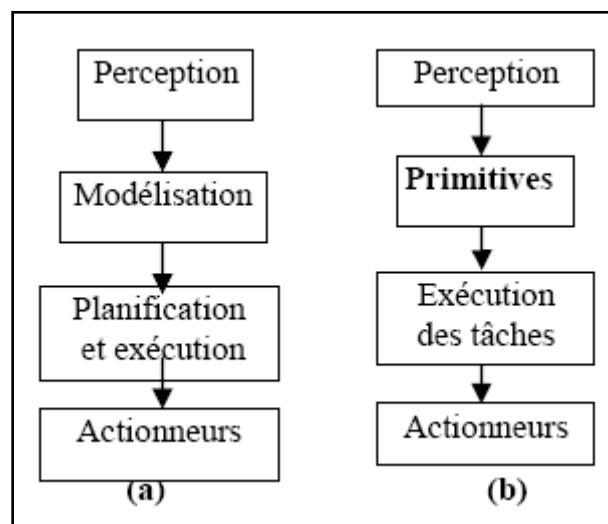


Figure II.1 : Méthodes de navigation :

(a) Délibérative (b) Réactive,

II.2 Approches de navigation :

II.2.1 Approche délibérative :

Les premiers robots mobiles utilisaient des contrôleurs dont le fonctionnement repose essentiellement sur la capacité de décision travaillant sur un modèle de l'environnement supposé parfait et connu a priori ou obtenu à partir des données capteurs (**Figure II.1-a**). Le fonctionnement est basé sur un cycle rigide de perception et modélisation de l'environnement, planification des actions au sein de cette représentation tout en évitant les obstacles, puis

exécution du plan. La **figure II.2** illustre, d'une manière plus détaillée, cette approche appelée souvent hiérarchique, cognitive ou délibérative. [MEN 12]

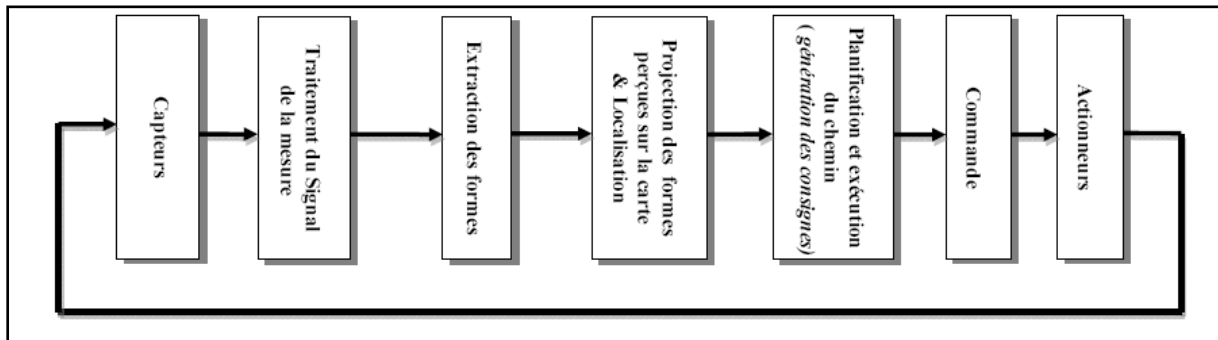


Figure II.2 : Schéma général d'une architecture délibérative.

II.2.2 Approche réactive :

Elle ne suppose aucun modèle a priori de l'environnement. Elle s'appuie sur le couplage étroit entre les capteurs et les actionneurs pour générer en continu les commandes (sans planification). Elle consiste à offrir un ensemble de primitives réactives. Elles correspondent à des sous-tâches (convergence vers une cible, suivre un mur, éviter un obstacle) dont l'enchaînement est du ressort d'un planificateur de tâches ayant décomposé la tâche globale (**Figure II.3**). [MEN 12]

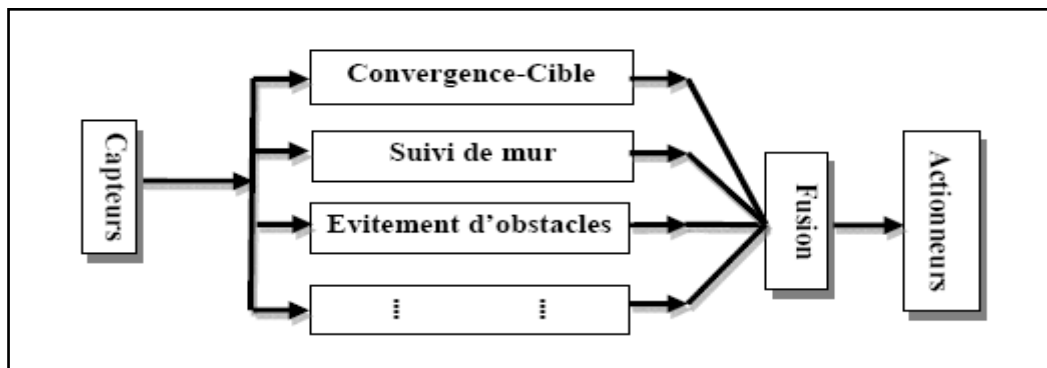


Figure II.3 : Stratégie de fusion de comportements.

II.2.3 Approche hybride :

Elle tente de combiner les deux approches précédentes afin de tirer partie des avantages respectifs de ces deux premières approches. [MEN 12]

II.3 Méthodes de Planification du Chemin :

On ne donnera ici qu'un aperçu des méthodes de planification dédiées à la robotique mobile. [MEN 12]

II.3.1 Espace des configurations:

L'espace des configurations d'un robot est constitué de l'ensemble des configurations qui lui sont accessibles. Cela s'illustre facilement dans le cas d'un robot représenté sous forme ponctuelle : *l'espace des configurations sans collision est l'espace dans lequel se déplace le robot, moins les obstacles.*

Dans le cas d'un robot de taille et de forme quelconque, on peut se ramener au problème de planification d'un robot ponctuel. Pour cela, on transforme l'environnement par un processus d'érosion de l'espace libre. Les points de l'espace libre qui ne peuvent être atteints par une configuration sans collision sont éliminés, selon le principe de **Figure II.4**. On peut bien considérer n'importe quel point du robot, et donc a fortiori le point remarquable OR, dont le mouvement est assimilé au mouvement du robot. Dans le cas pratique, on doit évidemment considérer, en plus, une marge de sécurité autour des obstacles. [MEN 12]

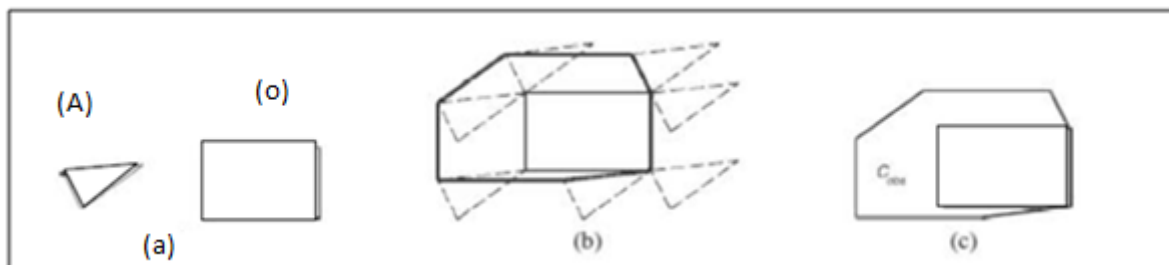


Figure II.4 : Reconstruction d'un obstacle pour transformer le problème de planification pour le robot en un problème de planification pour un point du robot : (a) robot triangulaire et point considéré A, obstacle O ;(b) robot au contact de l'obstacle ; (c) obstacle grossi C_{obs}.

II.3.2 Méthode de décomposition en cellules de l'espace de recherche :

Dans le cas où l'espace des configurations est représenté de manière continue, les techniques de planification ne sont utilisables qu'après discrétisation de l'espace libre, à l'exception de certaines techniques (telle que les champs de potentiel). La décomposition en cellules, de différents types, permettent de reproduire la topologie de l'espace libre. Les cellules obtenues sont alors utilisées de manière similaire aux nœuds des cartes topologiques dans le processus de planification. Les cellules adjacentes étant considérées comme reliées par une arête. [MEN 12]

A- Décomposition exacte en cellules :

Une décomposition **exacte** donne un ensemble de polygones subdivisant l'espace des configurations libres, **FigureII.5-(a)**. A partir de cette représentation, on crée un graphe reliant les différentes composantes adjacentes de l'environnement, **FigureII.5-(b)**. Ensuite, on identifie les cellules dans lesquelles se trouvent la configuration initiale et la configuration but. Le problème de planification consiste alors en un problème de recherche dans un graphe. Pour cela, on choisit un algorithme de parcours permettant d'optimiser la distance parcourue, ou tout autre critère. Pour finir, il faut relier par un chemin sans collisions les cellules du graphe solution. On peut, par exemple, utiliser le point milieu de la frontière de deux cellules adjacentes, comme cela est représenté à la **FigureII.5-(c)**. Une autre variante de cette méthode, avec quelques méthodes de planification associées, est illustrée par **Figure II.6**. Le chemin résultant peut être amélioré par des algorithmes d'optimisation. [MEN 12]

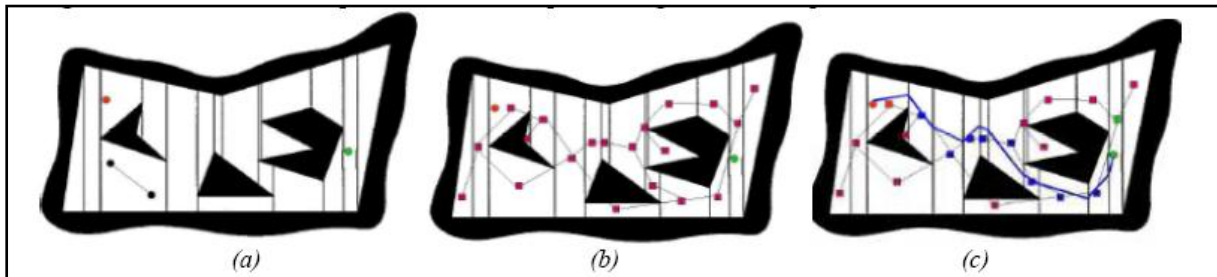


Figure II.5 : Décomposition exacte de l'espace libre en trapèzes et constitution d'un graphe.

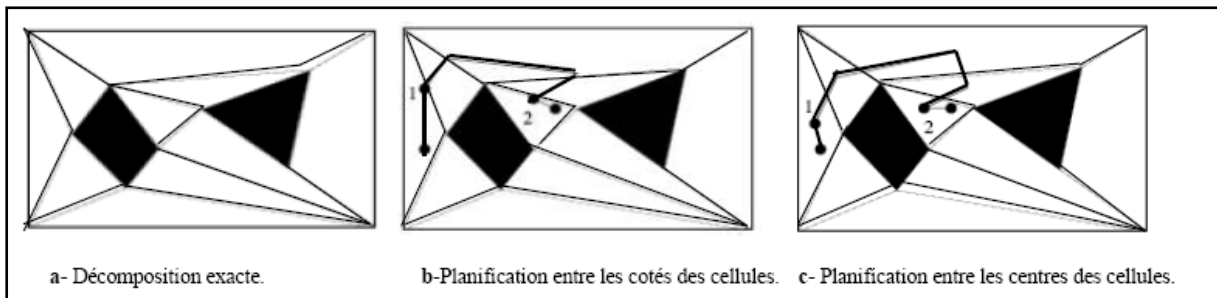


Figure II.6: Décomposition exacte et méthodes de planification.

B- Décomposition approchée en cellules:

La décomposition exacte permet de représenter l'ensemble de l'espace libre, à l'aide de cellules de formes irrégulières. La décomposition en cellules rectangulaires ne représente qu'un sous ensemble de l'espace libre suffisant pour la planification. Ces cellules peuvent être de taille régulière ou non. Enfin, une représentation hiérarchique telle que le «quadtree» permet d'utiliser des cellules de taille variable en fonction de la complexité locale de l'environnement. [MEN 12]

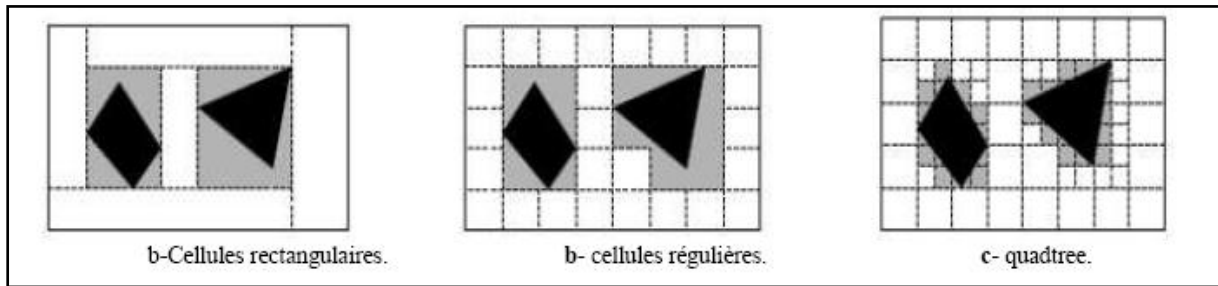


Figure II.7 : Décomposition exacte et méthodes de décomposition approchée de l'espace libre.

II.3.3 Chemins pré-calculés (Roadmaps):

Dans le cas où l'espace des configurations est représenté de manière continue, les techniques de planification ne sont utilisables qu'après discrétisation de l'espace libre, à l'exception de certaines techniques (telle que les champs de potentiel) qui calculent les chemins directement dans le domaine continu. [MEN 12]

Les méthodes de la seconde catégorie font appel au pré-calcul de chemins entre des points répartis dans l'environnement. Les points seront utilisés comme les nœuds d'une carte topologique. Tandis que les chemins pré-calculés reliant les nœuds seront utilisés comme les arêtes de cette carte.

Ces chemins relient des points particuliers utiles pour la navigation et répartis dans l'environnement. Différents choix de points sont possibles. [MEN 12]

A. Le diagramme de Voronoï :

Les premières roadmaps apparues étaient basées sur le diagramme de Voronoï qui utilise les points équidistants de plusieurs obstacles qui permettent de générer des chemins passant le plus loin possible des obstacles. Ces lignes permettent théoriquement de naviguer de manière sûre, en reliant la configuration initiale au point le plus proche du graphe et en navigant sur le graphe jusqu'au point le plus proche de la configuration finale, avant de rejoindre celle-ci. [MEN 12]

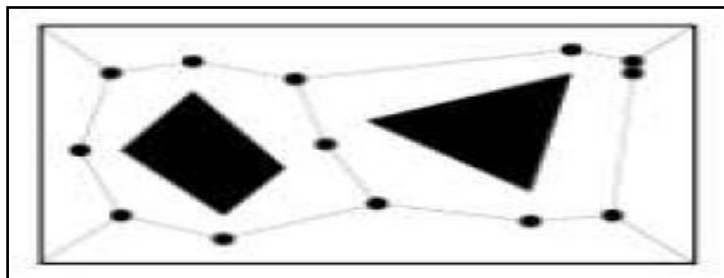


Figure II.8: Diagramme de Voronoï.

B. Le graphe de visibilité :

La construction d'un graphe de visibilité est une technique ancienne, utilisée sur Shakey (1970s). Elle utilise les sommets d'obstacles polygonaux qui sont les points que le robot devra contourner pour éviter ces obstacles. Le graphe est construit comme suit. On relie le point de départ aux sommets des obstacles qu'il peut voir. On fait de même avec le point final que l'on souhaite atteindre. On obtient un premier graphe (a). Ensuite, on reproduit l'algorithme en considérant tour à tour chacun des nouveaux nœuds du graphe ainsi crée (b). On obtient finalement le résultat représenté en (c). Une fois le graphe construit, on recherche un chemin dans celui-ci. D'une certaine manière, il passera ainsi plus près des obstacles. [MEN 12b]

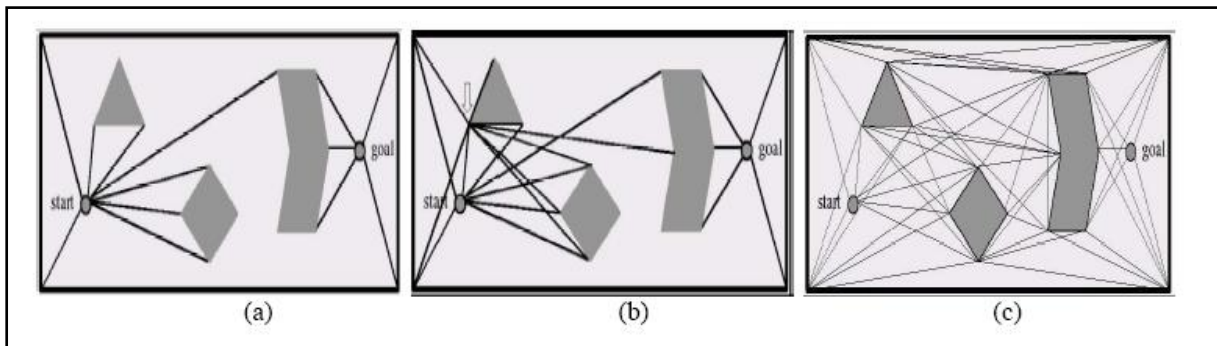


Figure II.9: Construction d'un graphe de visibilité.

II.3.4 Méthode du potentiel :

Cette méthode a été initialement introduite par O. Khatib pour des robots manipulateurs. Elle diffère sensiblement des méthodes déjà évoquées, du fait qu'elle ne résulte pas d'un raisonnement purement géométrique. Elle fournit une technique de navigation simple et compatible avec les exigences du temps réel.

Le principe de la méthode est le suivant. Le robot mobile est plongé dans un champ de potentiel, $U(q)$, qui résulte de la superposition d'un potentiel attractif, $U_{att}(q)$ lié à la configuration à atteindre, et d'une somme de potentiels répulsifs, U_{rep} , liés aux obstacles : [MEN 12b]

$$U(q) = U_{att}(q) + U_{rep}(q) \tag{01}$$

La force résultante est égale au gradient $F(q) = -\nabla U(q)$. Dans le cas simple d'un robot ponctuel, après reconstruction des obstacles, l'orientation peut être négligée et le champ est considéré dans l'espace $q = (x, y)^T$ et le gradient :

$$\nabla U = \left(\frac{\partial U}{\partial x}, \frac{\partial U}{\partial y} \right)^T$$

La force résultante est constituée de deux parties :

$$F(q) = F_{\text{att}}(q) + F_{\text{rep}}(q) \quad (02)$$

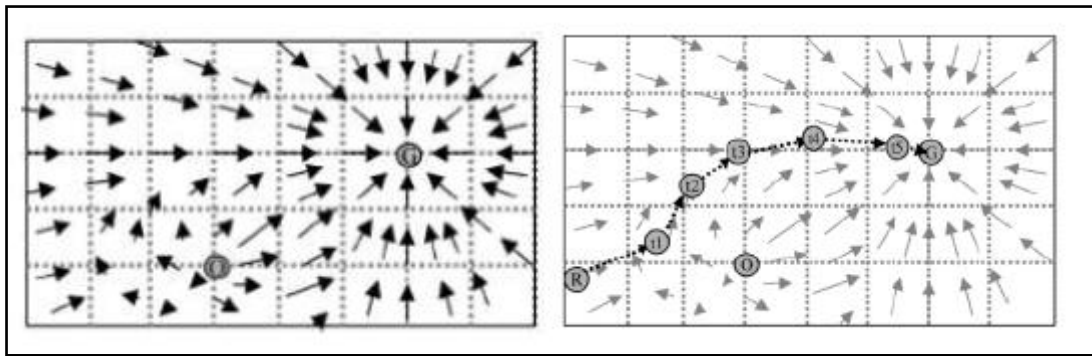


Figure II.10 : Carte des potentiels attractif et répulsif.

Cette évolution est généralement traduite en une vitesse de consigne, qui fait que cette méthode est tout autant une méthode réactive de commande qu'une méthode globale de planification. L'inconvénient majeur de cette méthode est la possibilité de blocage du robot à cause d'un minimum local de la fonction potentiel résultant de l'égalité : (potentiel attractif - potentiels répulsifs = 0). Pour surmonter cette difficulté, des versions modifiées ont été proposées dans la littérature. [MEN 12]

II.4 Navigation floue du robot mobile:

La problématique à traiter dans ce mémoire est la navigation réactive pour un robot mobile dans un environnement en utilisant le principe de la logique floue. La tâche confiée au robot est l'atteinte de cibles sans aucune intervention externe, et ce, dans un environnement sur lequel peu d'informations a priori lui sont fournies. Le robot est amené à exploiter les informations issues de ses capteurs pour pouvoir réaliser ces tâches et se guider de sorte à éviter les obstacles qui peuvent surgir sur son chemin.

Dans ce contexte plusieurs applications ont été développées pour assurer une navigation réactive dans un environnement même totalement inconnu tel que les méthodes de champ de potentiel. Dans a travail, on a opté pour la logique floue.

II.4.1 Aperçu sur la logique floue :

- **Historique**

Depuis une vingtaine d'années, l'utilisation de la logique floue dans la modélisation et la commande des systèmes complexes est devenue un outil à part entière. Le concept d'ensembles flous a été introduit par Zadeh en 1965[Zad 65]. Il constitue une interface de commande pour la modélisation du langage naturel, en particulier, des concepts linguistiques utilisés par l'expert d'un procédé. [Gue 03].

Le principe de la commande floue a été expérimenté pour la première fois, sur une turbine à vapeur par Mamdani et Assilian en 1974. La première application d'industrielle d'envergure, date de 1987 au Japon, dans le transport ferroviaire (métro) de Sendai (Tokyo), dont les performances rivalise celles d'un système de commande classique. [Sou 09]

De nos jours, les domaines d'application de la commande floue deviennent de plus en plus importants (industrie, automobile, robotique ...), et peuvent être classés en deux catégories :

- La conception de contrôleurs pour les procédés difficilement modélisables.
- La conception de contrôleurs pour les procédés modélisables non linéaires.

- **Contrôleur flou**

La logique floue a été conçue essentiellement, comme nous l'avons déjà dit, pour la commande des systèmes complexes ou mal définies et le traitement des données approximatives. Elle permet de prendre en considération des variables linguistiques dont les valeurs sont des expressions du langage naturel, telle que *négatif, positif, loin, proche, gauche droite,...*etc. A travers le nouveau concept d'ensembles flous, où un élément appartient de manière graduelle à un ensemble, elle permet, donc, de prendre en compte les états intermédiaires entre les deux états (tout ou rien) de la logique booléenne.

L'idée principale de la commande floue est d'imiter le processus du *raisonnement humain* en exprimant les connaissances humaines sous forme de règles de type « *si prémisses alors conclusions* ». La partie prémisses correspond à une description de l'état du système qui

déclenche les règles et les conclusions. Ces dernières représentent les actions qui doivent être appliquées au système. En résumé, les différents modules d'un contrôleur flou sont (**Figure II.11**) :

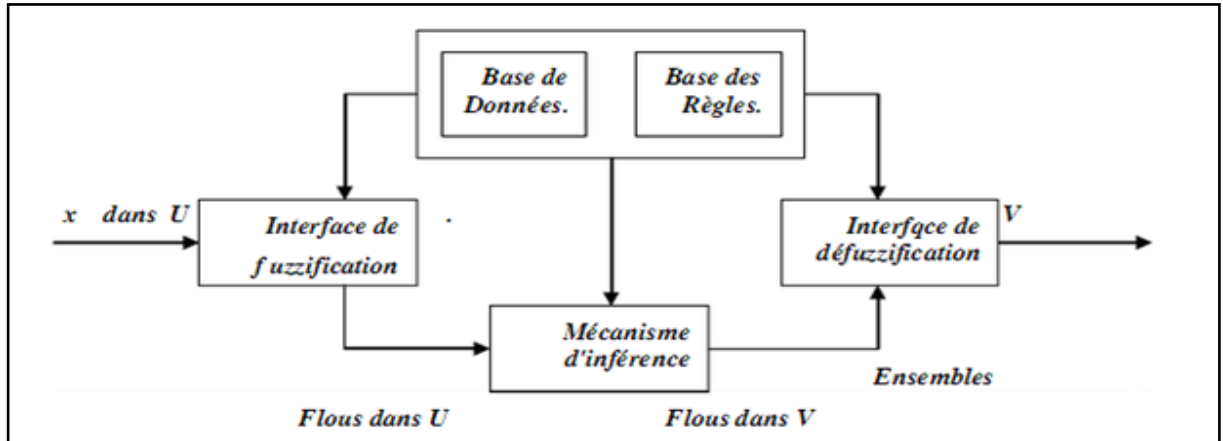


Figure II.11 : Configuration de base d'un contrôleur flou.

- **La fuzzification** : consiste à associer pour chaque valeur d'entrée, un ou plusieurs sous-ensembles flous. Cette étape fait passer les valeurs numériques en informations floues. Généralement la fuzzification singleton est utilisée.
- **Phase d'inférence**: consiste à calculer le degré de vérité de chaque règle et associer à chacune d'elle une valeur de sortie. Cette valeur est numérique dans le cas de contrôleur de type Sugeno et symbolique dans le cas de type Mandani.
- **La défuzzification** : dernière étape où les valeurs numériques des sorties sont obtenues à partir de la fonction d'appartenance résultante de la phase d'inférence floue. Dans la littérature, plusieurs méthodes sont proposées. Mais, les plus utilisées sont : méthode du centre de gravité (CG) et la méthode de la moyenne des maximas (MM).

La figure II.14 est une représentation graphique de ces étapes pour le calcul de la sortie d'un contrôleur flou de type Mamdani à deux variables d'entrée (X_1, X_2) et une variable de sortie (Y) intégrant la base de règles suivante :

- **Si X_1 est A_1 et X_2 est B_1 alors Y est C_1**
- **Ou, si X_1 est A_1 et X_2 est B_2 alors Y est C_2**

- Ou, si X_1 est A_2 et X_2 est B_2 alors Y est C_3

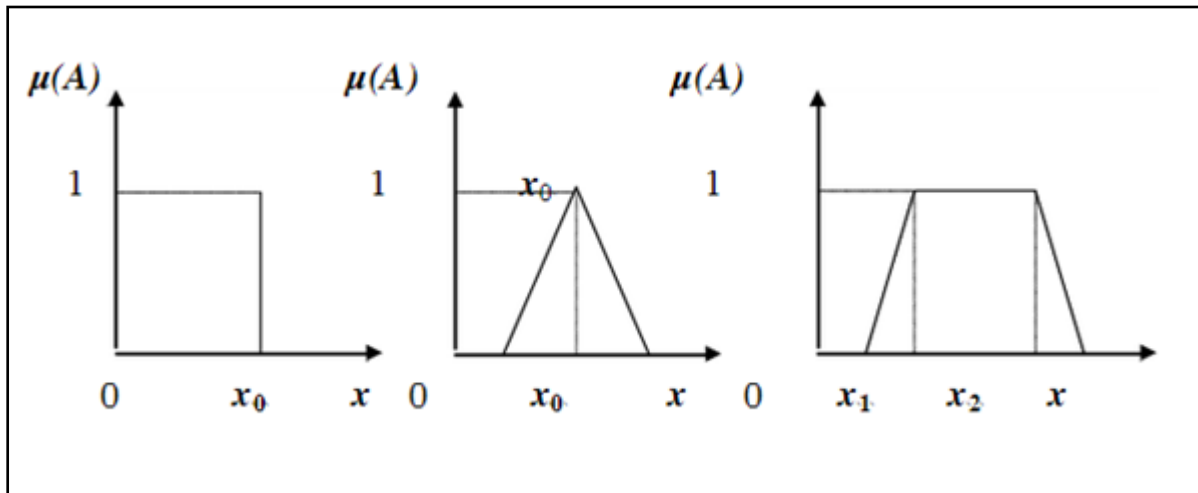


Figure II.12 : Fonction d'appartenance pour les 3 cas

II.4.2 Problématique de navigation floue d'un robot mobile :

La résolution du problème de navigation d'un robot mobile est sujette, en grande partie, aux recherches et aux résultats obtenus en intelligence artificielle ou en informatique. Dans cette optique, une des facultés premières que doit intégrer un robot mobile autonome est la capacité de mouvoir et de converger vers des lieux (cibles) qui lui sont indiqués au préalable sans aucune intervention externe. De plus, dans un environnement partiellement connu ou inconnu, des objets gênant peuvent surgir. Le robot doit être doté d'une seconde faculté qui est lui permet d'éviter ces obstacles. Dans notre cas, ces facultés sont mises en œuvre en utilisant le raisonnement flou. Dans un premier temps, nous avons implémenté ces deux facultés exclusivement.

II.5 Conclusion :

Dans ce chapitre, nous avons donné un aperçu général sur les stratégies de navigation des robots mobiles. La dernière partie est consacrée à la logique floue et l'exploitation de ses capacités de décisions pour la conception d'un système de navigation réactive. Les détails de cette conception seront donnés dans les deux chapitres suivants.

Chapitre III : Navigation dans un environnement sans obstacles

III.1 Introduction:

Dans ce chapitre, nous allons développer deux contrôleurs de navigation, le premier est basé sur le retour d'état et le deuxième avec la logique floue. Le contrôleur avec retour d'état est l'approche la plus appropriée pour contrôler les robots mobiles. Le contrôleur développé en utilisant le retour d'état réduit les tâches de planification et localisation des positions/orientations intermédiaires sur le chemin demandé. Notre contrôleur floue permet d'assurer la tâche de «*convergence vers la cible* » en considérant que ce comportement ne peut opérer que dans un environnement sans aucun obstacle gênant la progression du robot.

Pour que le robot puisse atteindre la cible, les coordonnées de cette dernière, lui sont introduites au début. Par la connaissance de ses position/orientation courantes, il calcule les valeurs des deux variables : distance du robot de la cible « D_{cible} » et l'erreur d'orientation (angle entre la direction du robot et le vecteur robot-cible) « φ_{cible} ».

III.2 Modélisation Cinématique du robot mobile :

Cette partie est consacrée à la modélisation des robots mobiles à roues. Dans le cas des robots manipulateurs, on s'intéresse seulement à la position de l'effecteur par rapport à la base. En plus, la mesure de celle-ci peut se faire en utilisant les mesures fournies par les capteurs liés aux articulations et le modèle cinématique. Dans le cas de la robotique mobile, c'est tout le robot qui se déplace par rapport au milieu et la modélisation doit tenir compte de la contribution de chaque roue et les contraintes associées. Pour la mesure, la difficulté vient du fait qu'on n'a aucun moyen direct pour mesurer la position instantanément et les méthodes alternatives utilisées conduisent souvent à des imprécisions. La dynamique du robot impose des contraintes supplémentaires en considérant les masses et les forces mises en jeu

III.2.1 Posture d'un robot :

Soit R un repère fixe quelconque, dont l'axe z est vertical et R_R un repère mobile lié au robot.

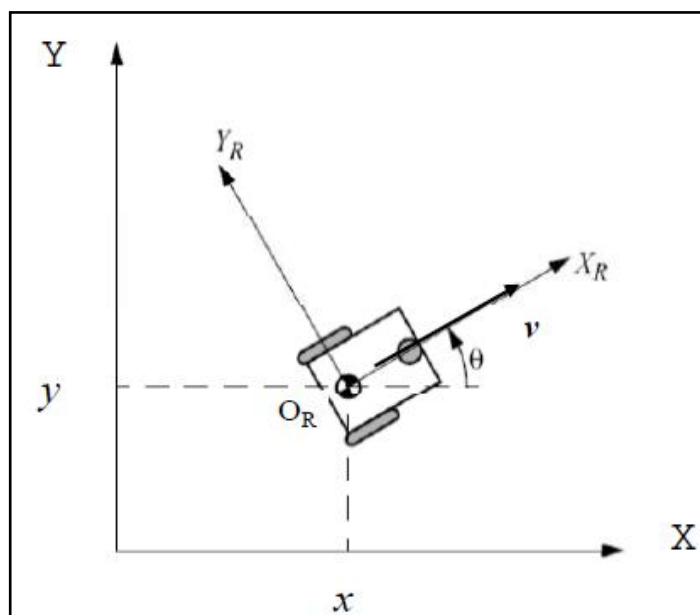


Figure III.1 : Situation du robot par rapport au repère de base.

On appelle *posture* du robot le vecteur $x = (x, y, \Theta)^T$, avec x , y et Θ sont respectivement l'abscisse, l'ordonnée et l'angle d'orientation du repère R_R , associé au robot,

par rapport au repère de base R. La posture du robot exprimée dans son propre repère peut être exprimée en fonction de ξ en utilisant la matrice d'orientation :

$$\xi_{R=R(\Theta)}.\xi = \begin{bmatrix} \cos\Theta & \sin\Theta & 0 \\ -\sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} . \xi \quad (1)$$

avec $R(\Theta)$ est la matrice de rotation ayant la propriété : $R(\Theta)^{-1}=R(\Theta)^T$. Cette matrice peut être utilisée pour la transformation des vitesses :

$$\dot{\xi}_{R=R(\Theta)}.\dot{\xi} \quad (2)$$

III.2.2 Roulement sans glissement:

La locomotion à l'aide de roues exploite la friction au contact entre roue et sol. Pour cela, la nature du contact (régularité, matériaux en contact) a une forte influence sur les propriétés du mouvement relatif de la roue par rapport au sol. Dans de bonnes conditions, il y a roulement sans glissement (**r.s.g.**) de la roue sur le sol, c'est-à-dire que la vitesse relative de la roue par rapport au sol au point de contact est nulle. Théoriquement, pour vérifier cette condition, il faut réunir les hypothèses suivantes :

- le contact entre la roue et le sol est ponctuel.
- les roues sont indéformables, de rayon **r**.

En pratique le contact se fait sur une surface, ce qui engendre bien évidemment de légers glissements. De même, alors qu'il est raisonnable de dire que des roues pleines sont indéformables, cette hypothèse est largement fautive avec des roues équipées de pneus. Malgré cela, on supposera toujours qu'il y a *r.s.g.* et, par ailleurs, que le sol est parfaitement plan.

Mathématiquement, on peut traduire la condition de *r.s.g.* sur une roue. Soit :

P : centre de la roue,

Q: point de contact de la roue avec le sol,

φ : angle de rotation propre de la roue (*dans le sens trigonométrique*),

Θ : angle entre le plan de la roue et le plan (o, x, z) (*orientation*),

On a : $P = (x, y, r)^T$, $Q = (x, y, 0)^T$, $v_p = (\dot{x}, \dot{y}, 0)^T$;

La nullité de la vitesse relative \vec{v}_Q roue/sol, au point de contact, permet d'obtenir une relation vectorielle entre la vitesse \vec{v}_P du centre P de la roue et le vecteur vitesse de rotation $\vec{\omega}$ de la roue :

$$\vec{v}_Q = \vec{v}_P + \vec{\omega} \wedge \overrightarrow{PQ} = \vec{0} \tag{4}$$

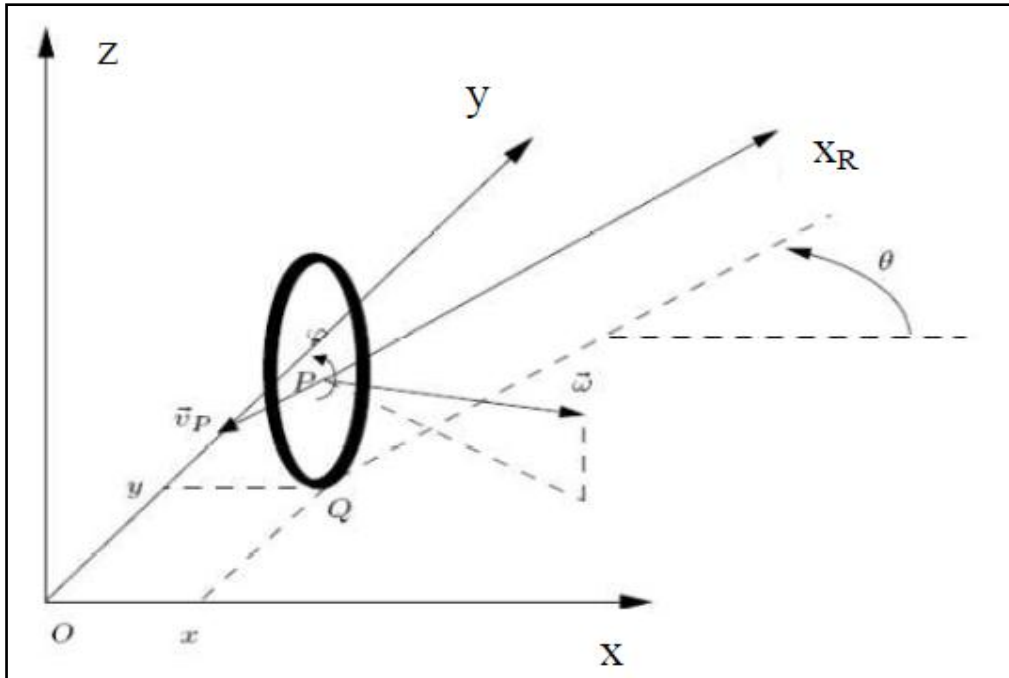


Figure III.2 : Caractérisation du roulement sans glissement

La partie $\vec{\omega} \wedge \overrightarrow{PQ}$ correspond au mouvement généré par la rotation de la roue. Il vient alors :

$$\vec{v}_Q = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ 0 \end{pmatrix} + \begin{pmatrix} \dot{\phi} \sin \Theta \\ \dot{\phi} \cos \Theta \\ 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ -r \end{pmatrix} \tag{5}$$

On obtient alors :

$$\vec{v}_Q = \begin{pmatrix} \dot{x} + r \cdot \dot{\phi} \cos \Theta \\ \dot{y} + r \cdot \dot{\phi} \sin \Theta \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{6}$$

Ceci qui se traduit par les deux contraintes scalaires :

$$\dot{x} + r \cdot \dot{\phi} \cos \Theta = 0$$

$$\dot{y} + r \cdot \dot{\phi} \sin \Theta = 0 \tag{7}$$

Que l'on peut transformer dans le repère mobile associé à la roue, pour faire apparaître les composantes de vitesse dans le plan de la roue et perpendiculairement à la roue :

Soit :

$$R(\theta) \cdot \begin{pmatrix} \dot{x} + r \cdot \dot{\phi} \cos\theta \\ \dot{y} + r \cdot \dot{\phi} \sin\theta \\ 0 \end{pmatrix} = \begin{pmatrix} \dot{x} \cdot \cos\theta + \dot{y} \cdot \sin\theta + r \cdot \dot{\phi} \\ -\dot{x} \cdot \sin\theta + \dot{y} \cdot \cos\theta \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (8)$$

$$\dot{x} \cdot \cos\theta + \dot{y} \cdot \sin\theta = -r \cdot \dot{\phi}$$

(Mouvement généré par la rotation dans le plan de la roue : pas de patinage)

$$-\dot{x} \cdot \sin\theta + \dot{y} \cdot \cos\theta = 0$$

(Pas de mouvement dans le plan \perp à la roue : pas de dérapage)

Ces contraintes traduisent le fait que le vecteur \vec{v}_p soit dans le plan de la roue et ait pour module $r\dot{\phi}$.

III.2.3 Centre instantané de rotation:

Bien évidemment, pour un ensemble de roues donné, toute disposition ne conduit pas à une solution viable. Par exemple, un robot équipé de deux roues fixes non parallèles ne pourrait pas aller en ligne droite. Pour qu'une disposition de roues soit viable et n'entraîne pas de glissement des roues sur le sol, il faut qu'il existe pour toutes ces roues un unique point de vitesse nulle autour duquel tourne le robot de façon instantanée. Ce point, lorsqu'il existe, est appelé *centre instantané de rotation* (**CIR**). Les points de vitesse nulle liés aux roues se trouvant sur leur axe de rotation, il est donc nécessaire que le point d'intersection des axes de rotation des différentes roues soit unique. **[MEN 02]**

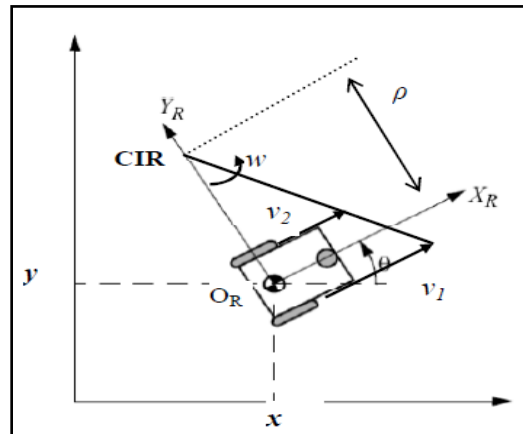


Figure III.3 : Centre instantané de rotation d'un robot de type unicycle.

III.2.4 Robots mobiles de type unicycle :

On désigne par *unicycle* un robot actionné par deux roues indépendantes et possédant éventuellement un certain nombre de roues folles qui assurent sa stabilité et qui n'interviennent pas dans la cinématique. Ce type de robot est très répandu en raison de sa simplicité de construction et de propriétés cinématiques intéressantes. Soit O_R est le point milieu entre les deux roues à une distance L de chacune. Etant données les vitesses de rotation des roues droite et gauche $(\dot{\theta}_1, \dot{\theta}_2)$ le modèle cinématique direct permet de prédire la vitesse globale du robot par rapport au repère de base :

$$\dot{\xi} = (\dot{x} \quad \dot{y} \quad \dot{\theta})^T = f(L, r, \theta, \dot{\theta}_1, \dot{\theta}_2) \quad (9)$$

La contribution de chaque roue peut être calculée séparément, par rapport au repère R_R , puis on calcule la somme. La vitesse linéaire de la roue droite est donnée par :

$$v_1 = r \cdot \dot{\theta}_1 \quad (10)$$

Sa rotation en avant conduit à une rotation du robot dans le sens trigonométrique. Quand la roue droite tourne seule, le centre du robot, O_R , pivote autour de la roue gauche avec une vitesse de rotation :

$$W_1 = \frac{r \cdot \dot{\theta}_1}{2L} \quad (11)$$

Soit celle de rotation de la roue droite sur un cercle de rayon $2L$.

Le même calcul peut se faire pour la roue gauche dont le déplacement en avant conduit à la rotation du robot dans le sens horaire, soit :

$$W_2 = -\frac{r \cdot \dot{\theta}_2}{2L} \quad (12)$$

Ainsi, la vitesse de rotation globale du robot dans le repère R_R est :

$$W = W_1 + W_2 = \frac{r \cdot \dot{\theta}_1}{2L} - \frac{r \cdot \dot{\theta}_2}{2L} \quad (13)$$

La vitesse linéaire est la moyenne des deux:

$$V = \frac{v_1 + v_2}{2} = \frac{r \cdot \dot{\theta}_1}{2} + \frac{r \cdot \dot{\theta}_2}{2} \quad (14)$$

Le rayon de courbure de la trajectoire du robot, ρ , c'est-à-dire la distance entre CIR et OR :

$$\rho = L \frac{\dot{\theta}_1 + \dot{\theta}_2}{\dot{\theta}_1 - \dot{\theta}_2} \quad (15)$$

Quand $\dot{\theta}_1 = \dot{\theta}_2$ le robot se déplace en ligne droite et ρ est infini. Si $\dot{\theta}_1 = -\dot{\theta}_2$ alors le robot effectue une rotation sur lui-même et $\rho = 0$.

Le modèle cinématique exprimant le mouvement du robot par rapport au repère de base :

$$\dot{\xi} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = R(\theta)^{-1} \begin{pmatrix} \frac{r \cdot \dot{\theta}_1}{2} + \frac{r \cdot \dot{\theta}_2}{2} \\ 0 \\ \frac{r \cdot \dot{\theta}_1}{2L} - \frac{r \cdot \dot{\theta}_2}{2L} \end{pmatrix} = R(\theta)^{-1} \begin{pmatrix} v \\ 0 \\ w \end{pmatrix} = \begin{pmatrix} v \cdot \cos(\theta) \\ v \cdot \sin(\theta) \\ w \end{pmatrix} = C(\theta) \cdot u \quad (16)$$

Cette équation décrit le modèle cinématique en posture qui relie la variation dans le temps de la posture, $\dot{\xi}$, à la commande $u = (v, w)^T$, avec :

$$R(\theta)^{-1} = R(\theta)^T = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad C(\theta) = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix}, \quad u = \begin{pmatrix} v \\ w \end{pmatrix}$$

Une commande par retour d'état de forme :

$$u = \begin{pmatrix} v \\ w \end{pmatrix} = K \cdot e \quad (17)$$

Permet de mener le robot vers la position désirée, avec e est l'écart entre le vecteur

$\xi = (x, y, \theta)^T$ (décrivant la situation actuelle du robot) et sa valeur finale désirée et K est une matrice de gain (2x3).

III.3 Navigation Avec retour d'état :

Considérant le robot dans une posture initiale quelconque. L'objectif est de calculer le retour d'état permettant déplacer le robot vers la posture finale désirée, $x_f = (x_f, y_f, q_f)^T$.

Pour ce faire, on préfère utiliser le modèle en coordonnées polaires (Figure III.4) :

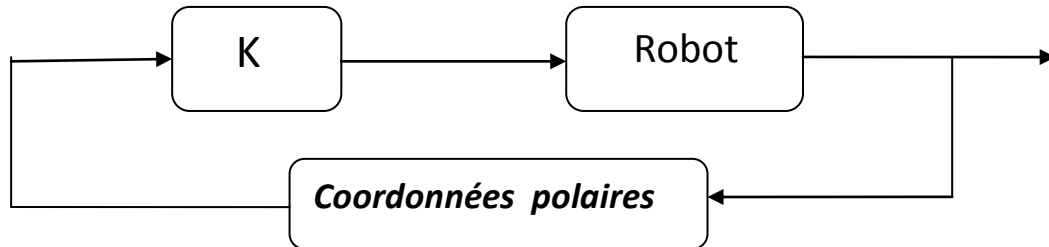


Figure III.4 : Schéma du système de commande.

III.3.1 Commande du robot par retour d'état :

Le modèle est donné par les équations suivantes:

- $d = \sqrt{\Delta x^2 + \Delta y^2}$

Avec : $\Delta x = x_f - y_0$ et $\Delta y = y_f - y_0$

- $\alpha = -\theta + \text{atan}(y/x)$
- $\beta = -\alpha - \theta$

Ce qui donne le modèle suivant:

$$\begin{pmatrix} \dot{d} \\ \dot{\alpha} \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} -\cos(\alpha) & 0 \\ \sin(\alpha)/d & -1 \\ -\sin(\alpha)/d & 0 \end{pmatrix} \cdot \begin{pmatrix} Va \\ w \end{pmatrix}$$

Avec :

d : distance entre le robot et le point cible.

$\alpha = [-\pi/2, \pi/2]$: angle entre le robot (axe X_R) et la direction du point cible (orientation robot-cible).

θ : angle entre robot (axe X_R) et l'axe X du repère de base (orientation robot espace de travail)

La figure sous dessous schématise le modèle de retour d'état :

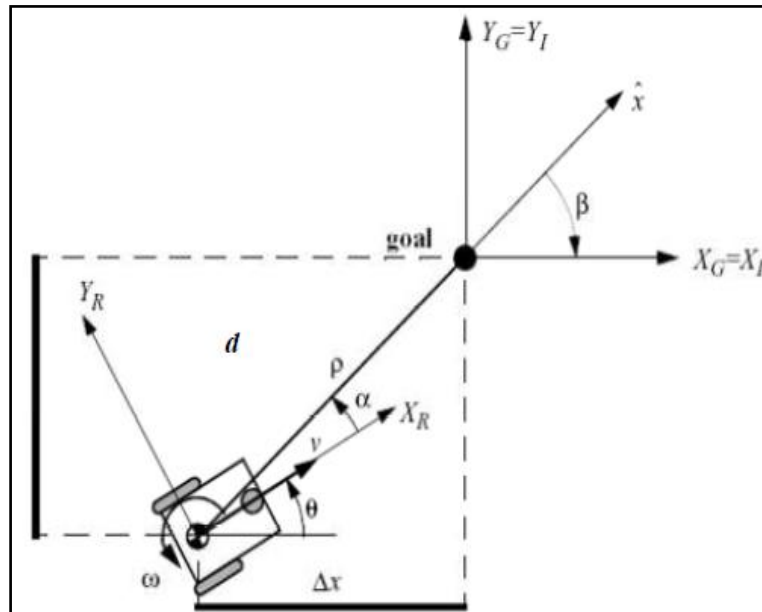


Figure III.5 : Distance et orientation du robot vers la cible avec retour d'état

Considérant maintenant la loi de commande linéaire :

$V_a = k_p \cdot d$: la vitesse d'avancement de robot.

$w = k_a \cdot \alpha + k_b \cdot \beta$: la vitesse de braquage du robot.

dans cet exemple en prenant : $(k_p, k_a, k_b) = (3, 8, -1.5)$

Qui donnera le modèle cinématique suivant:

$$\begin{cases} \dot{x} = v_a * \cos(\theta) \\ \dot{y} = v_a * \sin(\theta) \\ \dot{\theta} = w \end{cases} \quad (2.1)$$

Tel que (x,y) coordonnées du robot dans le repère absolu et θ son orientation. La position et orientation du robot sont obtenues par intégration de l'équation (2.1), d'où :

$$\begin{cases} \theta(i+1) = \theta(i) + \dot{\theta} * T_e \\ x(i+1) = x(i) + \dot{x} * T_e \\ y(i+1) = y(i) + \dot{y} * T_e \end{cases} \quad (2.2)$$

Tel que : T_e est la période d'échantillonnage.

III.3.2 Résultats de simulation :

Le premier résultat, montre bien que le robot adopte le comportement souhaité, à savoir : le robot tout en s'orientant vers la cible (avec une vitesse de braquage modérée mais permanente) tout en accélérant (avec une vitesse inversement proportionnelle à l'erreur d'orientation) durant cette manœuvre. Enfin, à partir d'une distance jugée proche de la cible, décélère jusqu'à l'arrêt total une fois cette dernière atteinte.

1^{er} cas: Point départ $\begin{cases} x_0 = 15 \\ y_0 = 10 \end{cases}$ Point arrivée $\begin{cases} x_f = 40 \\ y_f = 30 \end{cases}$

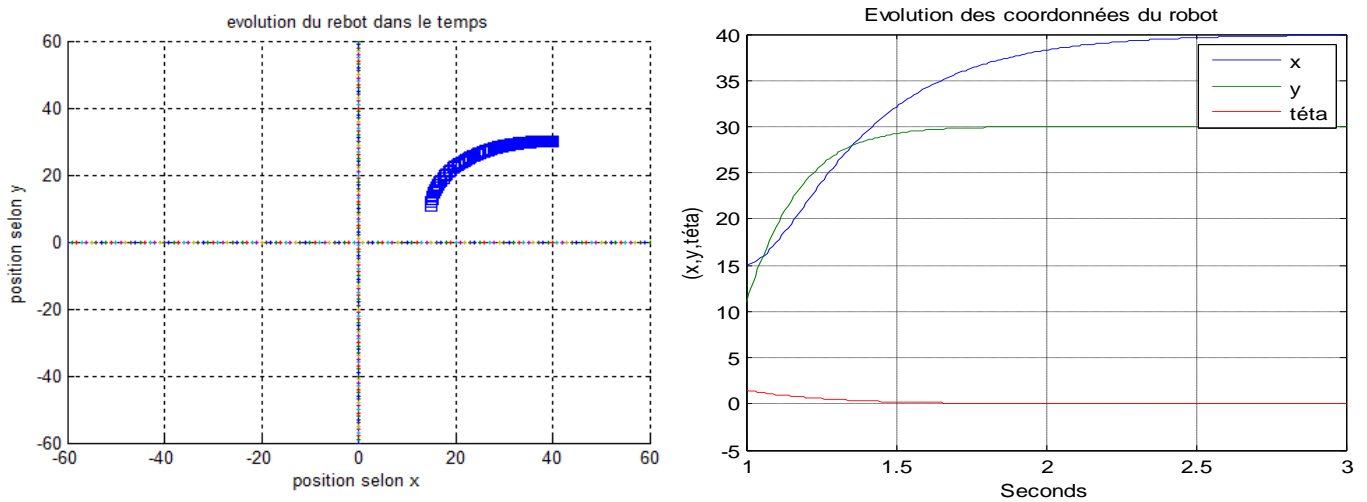


Figure III.6 : navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 1^{er} exemple

2^{eme} cas: Point départ $\begin{cases} x_0 = 30 \\ y_0 = 30 \end{cases}$ Point arrivée $\begin{cases} x_f = 120 \\ y_f = 70 \end{cases}$

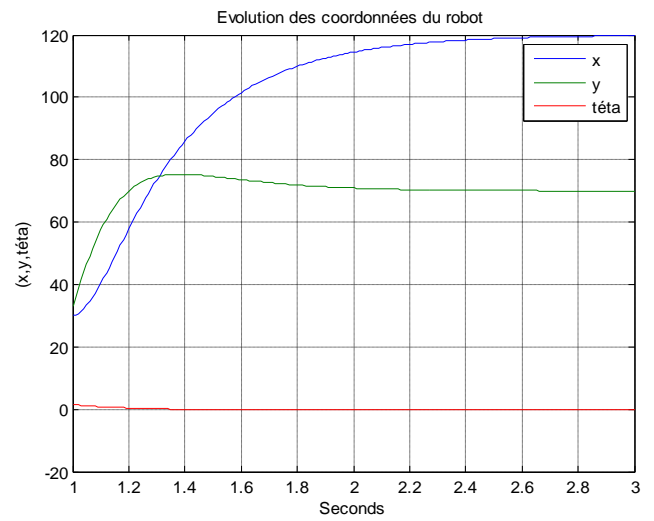
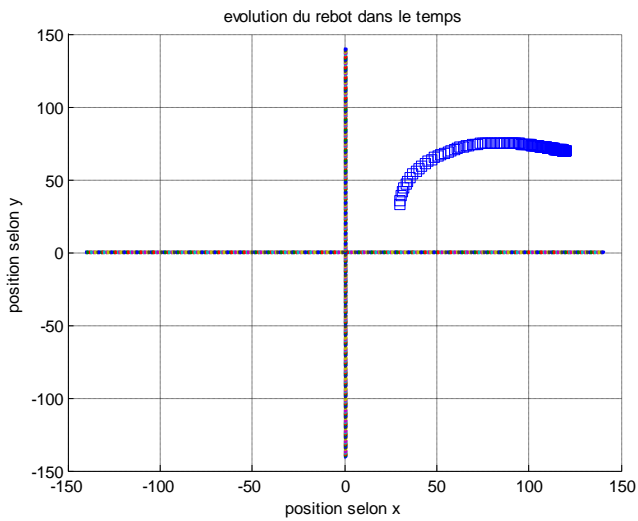


Figure III.7: navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 2^{eme} exemple

3^{eme} & 4^{eme} cas: Point départ $\begin{cases} x_0 = 90 \\ y_0 = 80 \end{cases}$ Point arrivée $\begin{cases} x_f = 10 \\ y_f = 10 \end{cases}$

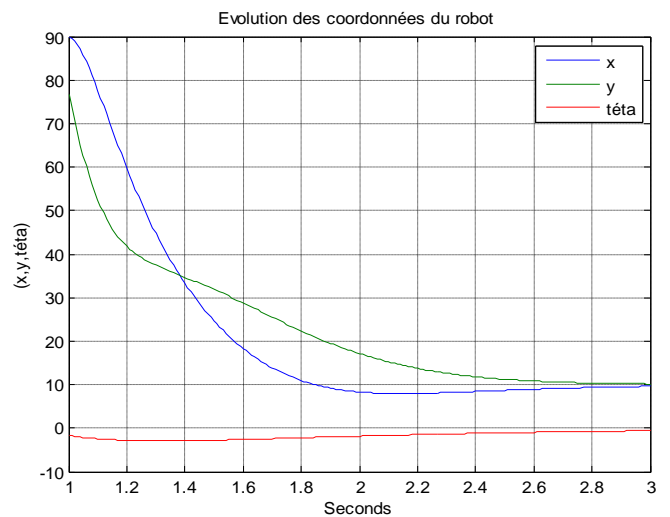
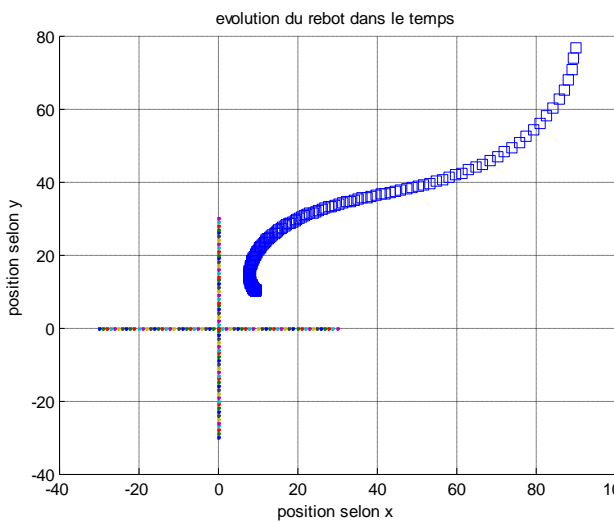


Figure III. 8 : navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 3^{eme} exemple

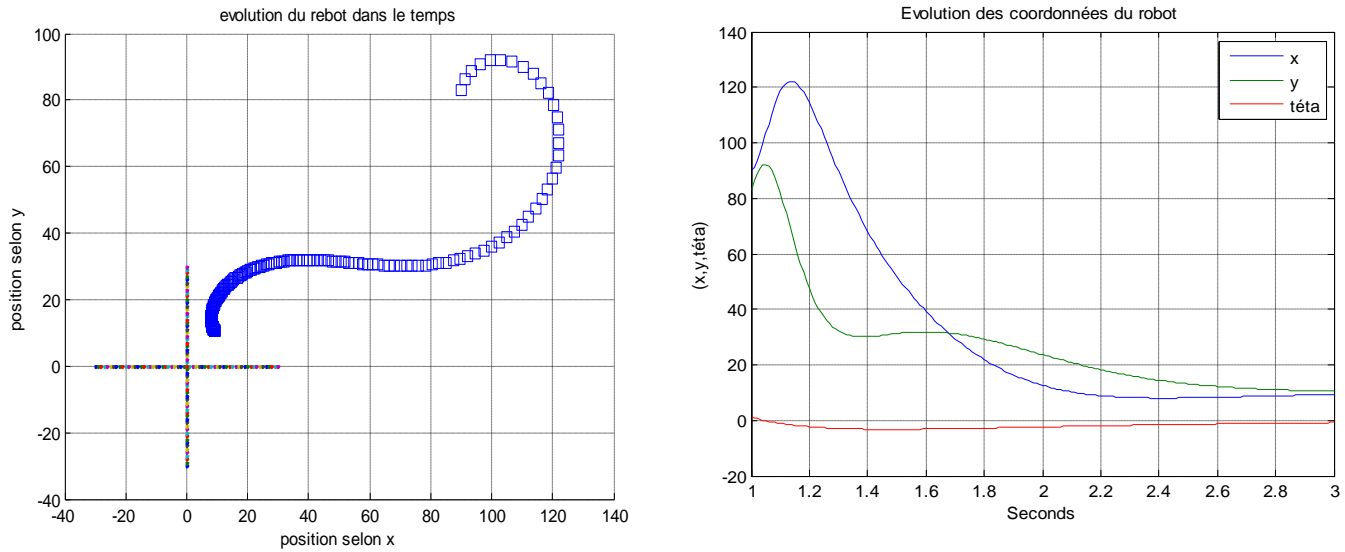


Figure III.9 : navigation du robot en utilisant le retour d'état dans un environnement sans obstacle: 4^{eme} exemple

Nous remarquons dans les figures III.8 et III.9 que la trajectoire effectuée par le robot est différente malgré les mêmes données de départ et d'arrivée, ceci s'explique par la position initiale du robot. En effet dans les deux cas, la position du robot diffère.

III.4 Navigation en utilisant la logique floue :

Le comportement de contrôleur avec la logique permet au robot de se mouvoir en utilisant seulement les informations sur ses positions/orientations courantes et la position cible à atteindre ; sans aucune prétention aux états des capteurs d'évitement d'obstacle et, par ce, ignore la présence des obstacles. L'unique objectif visé par ce contrôleur est de guider les déplacements du robot pour qu'il puisse atteindre le point d'arrivée. Il est bien entendu que ce comportement ne peut opérer que dans un environnement sans aucun obstacle gênant la progression du robot.

Figure III.10 représente la distance et l'orientation initiales du robot par rapport à la cible.

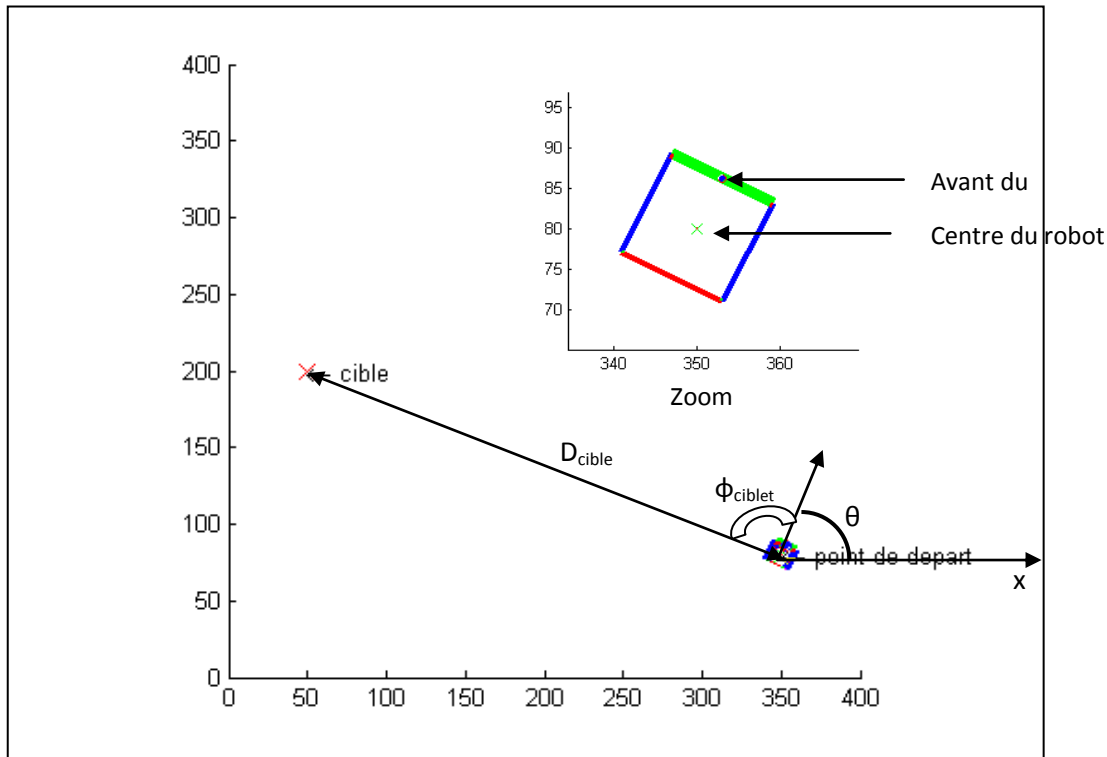


Figure III.10 : Distance et orientation initiales du robot par rapport à la cible.

III.4.1 Commande du robot par la logique floue:

Pour une commande cinématique du robot, en utilisant le modèle d'un robot mobile à roues différentielles, les variables de sorties (qui sont calculés avec la logique floue) du contrôleur sont : la vitesse de braquage (**W**) et la vitesse d'avancement (**V_a**) du robot (où $q=[X, Y, \theta]$ est la position dut robot). (Figure III.11)

Le modèle cinématique utilisé est donné par :

$$\begin{cases} \dot{x} = v_a * \cos(\theta) \\ \dot{y} = v_a * \sin(\theta) \\ \dot{\theta} = w \end{cases} \quad (2.1)$$

Tel que (x,y) coordonnées du robot dans le repère absolu et θ son orientation.

La position et orientation du robot sont obtenues par intégration de l'équation (2.1), d'où :

$$\begin{cases} \theta(i+1) = \theta(i) + \dot{\theta} * Te \\ x(i+1) = x(i) + \dot{x} * Te \\ y(i+1) = y(i) + \dot{y} * Te \end{cases} \quad (2.2)$$

Tel que : Te est la période d'échantillonnage

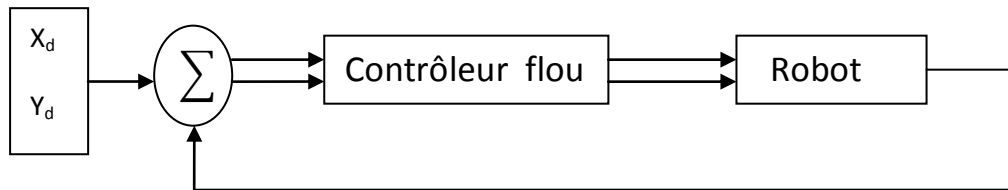


Figure III.11: Schéma du système de commande.

III.4.2 Présentation schématique d'un robot mobile :

Puisque nous avons fait appel au contrôle flou, nous avons d'abord procédé à une partition floue de l'espace opérationnel en trois orientations (gauche, avant, droite) où gauche pour $\varphi \in] 0 , \pi]$, droite pour $\varphi \in [- \pi , 0 [$ (selon le sens horaire)

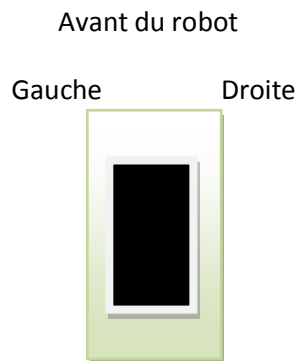


Figure III.12: Schéma représentatif d'un robot mobile

III.4.3 Contrôleur flou :

III.4.3.1 Définition des fonctions d'appartenance :

➤ *Fonctions d'appartenance d'entrée*

Les univers de discours des variables d'entrée (D_{cible} et ϕ_{cible}) du contrôleur de convergence vers la cible, retenus après plusieurs essais, sont respectivement décomposés en trois et cinq sous ensembles flous selon figure (III.13). Cette partition floue est assez simple et permet d'obtenir une base de règle concise et facile à interpréter (15 règles).

Les labels utilisés pour la distance D sont : Pr (proche), Moy (moyennement proche) et L (loin) et pour la variable angle d'orientation ϕ_{cible} : DrRap (droit rapide), DrLent (droit lent), Z (zéro) et Galent (gauche lent) et GaRap (gauche rapide).

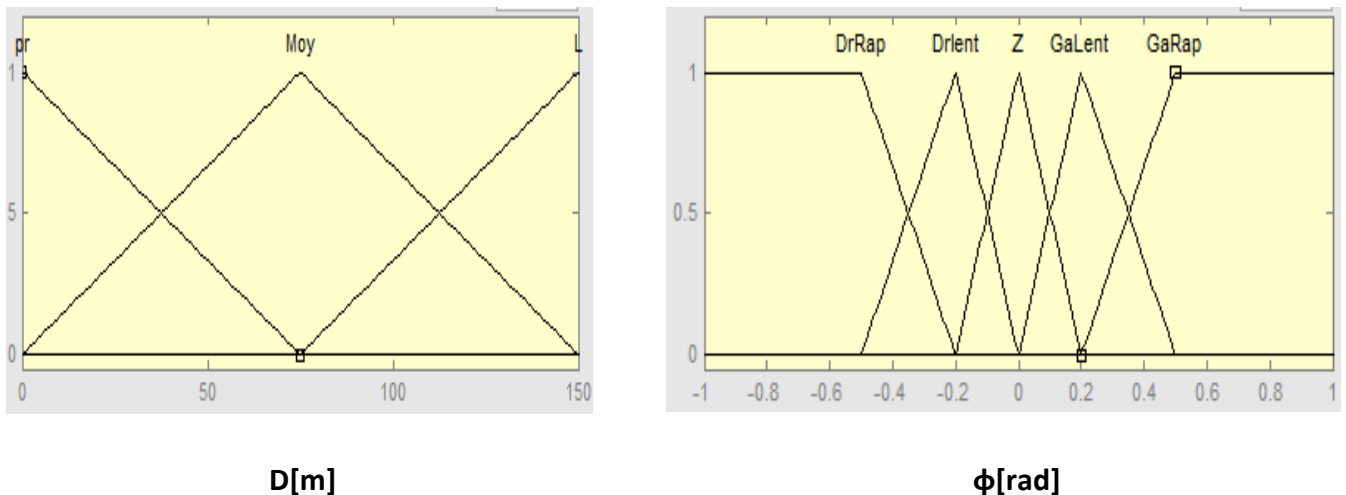


Figure III.13 : les fonctions d'appartenance des variables d'entrée du contrôleur « convergence vers la cible »

➤ *Fonctions d'appartenance de sortie :*

Quant aux variables de sortie du contrôleur (la vitesse d'avancement ' Va ' et la vitesse braquage W) sont partitionnées comme suit : la première variable est partitionnée en trois sous-ensembles flous : Lent(lent), Rap (rapidement) et TRap (très rapidement) et la deuxième variable est partitionnée en cinq sous-ensembles flous : DrRap (droit rapide), DrLent (droit lent), Z (zéro) et Galent (gauche lent) et GaRap (gauche rapide) figure (III.14)

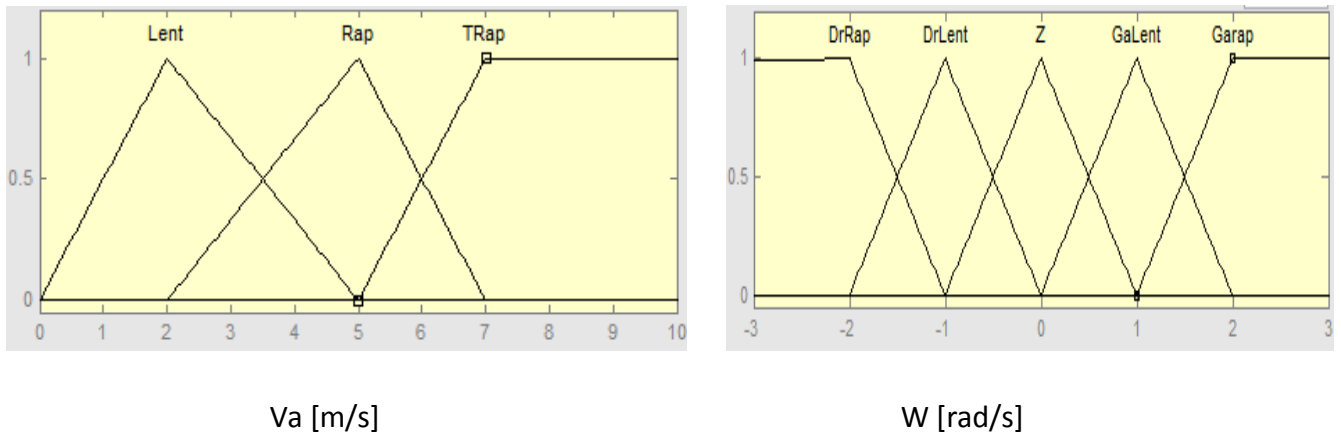


Figure III.14 : Les fonctions d'appartenance des variables de sortie du contrôleur.

III.4.3.2 Base de règles :

Le contrôleur flou utilisé est de type Mamdani et l'étape d'inférence floue fait appel à des règles floues faisant un lien entre les variables d'entrée D_{cible} et φ_{cible} et les variables de sortie Va et W . Les règles floues sont des expressions formées de prémisses (conditions) et de conclusions (actions) de la forme:

- **Si D_{cible} est L et φ_{cible} est C alors W est Z et Va est TRap.**

Les deux tableaux ci-dessous représentent la base de règles floues pour les deux variables de sortie **Va** Tableau (III.1) et **W** tableau (III.2), et ce, pour toutes les combinaisons possibles des deux variables d'entrée. Ces deux tables sont établit sur la base du raisonnement suivant : à partir de sa position initiale, le robot s'oriente vers la cible puis converge vers elle avec une vitesse inversement proportionnelle à la distance robot-cible pour s'arrêter une fois la cible atteinte avec une marge d'erreur admissible.

ϕ_{cible} \ D _{cible}	Pro	Moy	L
CD	Z	Z	Z
MD	Z	Rap	Rap
C	Z	TRap	TRap
MG	Z	Rap	Rap
CG	Z	Z	Z

Tableau III.1 Base de règles floues de la vitesse d'avancement.

ϕ_{cible} \ D _{cible}	Pro	Moy	L
CD	DrRap	DrRap	DrRap
MD	Dlent	DrRap	DrRap
C	Z	Z	Z
MG	GaLen	GaRap	GaRap
CG	GaRap	GaRap	GaRap

Tableau III.2 base de règles floues de la vitesse de braquage.

III.4.4 Résultats de simulation :

Ce résultat second, montre bien que le robot adopte le comportement souhaité (avec la logique floue), à savoir : le robot s'oriente d'abord vers la cible (vitesse de braquage maximale) tout en accélérant (avec une vitesse inversement proportionnelle à l'erreur

d'orientation) durant cette manœuvre. Ensuite, le robot avance avec une vitesse maximale, et enfin, à partir d'une distance jugée proche de la cible, décélère jusqu'à l'arrêt total une fois cette dernière est atteinte.

1^{er} cas: Point départ $\begin{cases} x_0 = 0 \\ y_0 = 0 \end{cases}$ Point arrivée $\begin{cases} x_f = 20 \\ y_f = 10 \end{cases}$

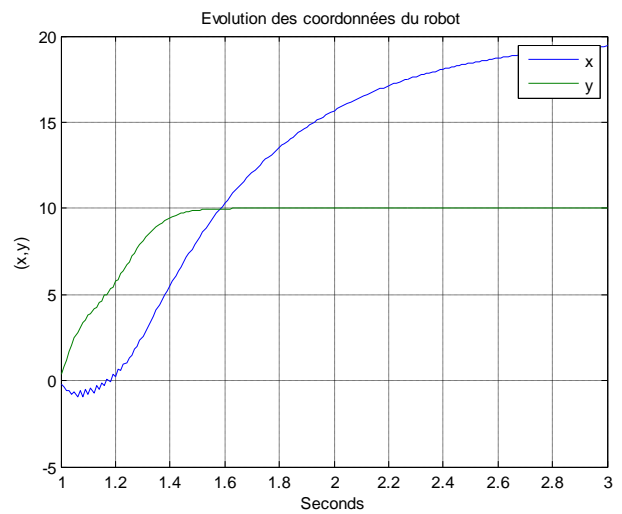
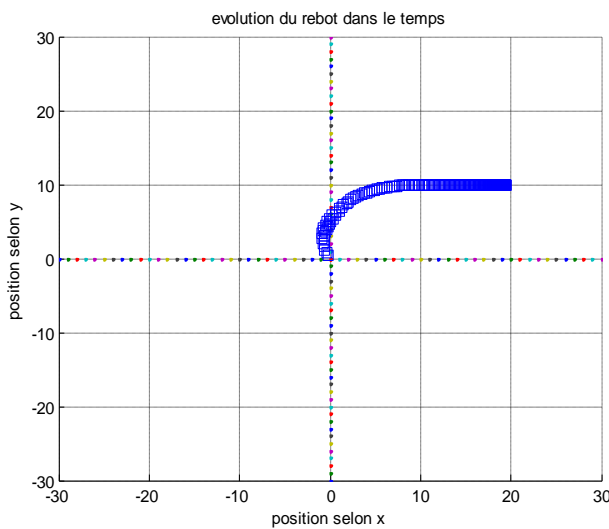


Figure III.15 : navigation du robot en utilisant la logique floue Dans un environnement sans obstacle: 1^{er} exemple

2^{eme} cas: Point départ $\begin{cases} x_0 = 30 \\ y_0 = 30 \end{cases}$ Point arrivée $\begin{cases} x_f = 120 \\ y_f = 70 \end{cases}$

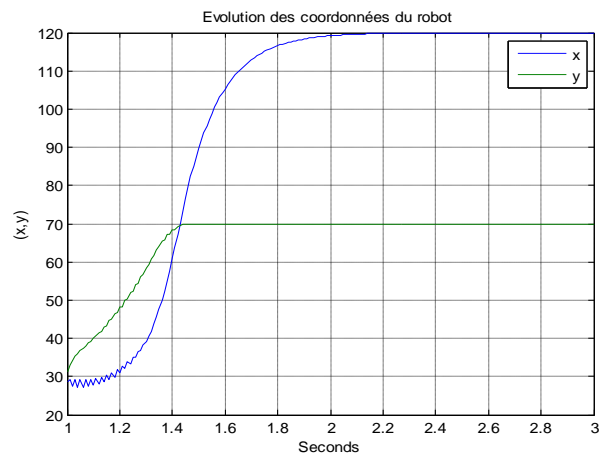
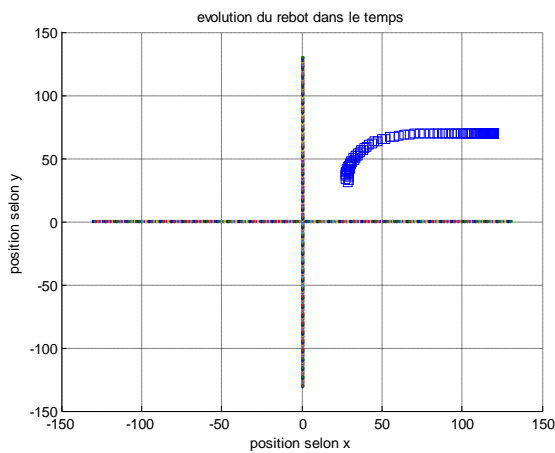


Figure III.16 : navigation d u robot en utilisant la logique floue dans un environnement sans obstacle: 2^{eme} exemple

III.5 Conclusion:

- Dans ce 1^{er} cas, nous avons exploré le concept d'une commande de retour d'état d'un robot mobile dans un environnement connu, et ce, par une approche mathématique à savoir la navigation et la convergence du robot mobile vers une cible, avec une vitesse de braquage modérée mais permanente mais sans la prise en compte des obstacles
- Dans ce 2^{ème} cas, nous avons créé un système de contrôle basé sur la logique floue qui par ses vitesses de braquage et d'avancement nous permet d'atteindre la cible.

Chapitre IV : Navigation dans un environnement avec obstacles

IV.1 Introduction:

La problématique à traiter dans ce chapitre est la planification de trajectoires pour un robot mobile. La tâche confiée au robot est l'atteinte de cibles sans aucune intervention externe, et ce, dans un environnement sur lequel peu d'informations a priori lui sont fournies. Le robot est amené à exploiter les informations issues de ses capteurs pour pouvoir réaliser ces tâches et se guider de sorte à éviter les obstacles qui peuvent surgir sur son chemin.

La possibilité de naviguer avec une certaine autonomie dans un environnement contenant des obstacles est l'un des problèmes des robots mobiles. L'autonomie de navigation d'un robot mobile est la capacité des robots à prendre des décisions indépendamment de l'opérateur humain dans leurs déplacements.

IV.2 Le contrôleur d'évitement d'obstacles :

Dans le chapitre précédant, les résultats obtenus pour un environnement sans obstacles sont assez satisfaisants. Néanmoins, pour une approche non loin de la réalité, où des objets (obstacles) gênant la progression du robot peuvent être présents, le contrôleur de convergence vers la cible, employé seul, ne peut garantir au robot l'accomplissement de la tâche qui lui est assignée. Donc, le robot doit être muni d'un deuxième contrôleur lui permettant d'éviter la collision avec ces obstacles.

Le contrôleur dont on parle, *contrôleur d'évitement d'obstacles*, exploite les données (positions des obstacles par rapport au robot) issues des capteurs, répartis comme le montre la figure (IV.1), pour guider le déplacement du robot.

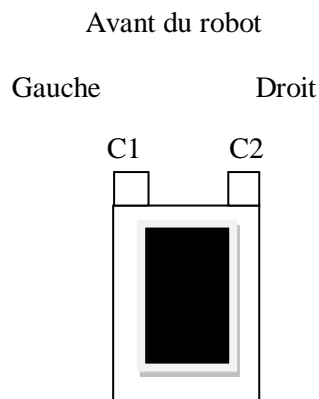


Figure IV.1: Répartition des capteurs du robot

IV.2.1 Perception et détection des obstacles :

Dans notre cas, un module de perception est intégré via différentes procédures. La démarche suivie pour la simulation des capteurs de détection des obstacles est la suivante :

Calcul, à chaque instant, des distances entre le robot et les obstacles ainsi que leurs positions. Si des obstacles se trouvent dans un périmètre de rayon « D » (simulant la portée des capteurs), on calcule l'état de chaque capteur, et ce, en fonction de la distance robot-obstacle et sa position (angle entre la direction du robot et les obstacles).

IV.2.2 Conception du navigateur réactif flou :

Afin d'assurer la non collision du robot avec des obstacles, le contrôleur d'évitement d'obstacles est sollicité à chaque fois qu'un capteur (ou plus) détecte un obstacle situé à une distance inférieure à la distance de sécurité (D_s). Ce dernier génère une commande permettant d'orienter le robot dans le sens opposé à ce dernier, c'est-à-dire, le robot s'oriente toujours dans la direction libre d'obstacles. Par exemple, si le capteur C1 détecte un obstacle proche, le robot tourne à droite et dans le cas où le capteur C2 détecte un obstacle le robot tourne à gauche. La figure (IV.2) donne la représentation du schéma de contrôle utilisé. On a utilisé, comme entrées du contrôleur, les rapports des distances mesurées par les capteurs C1-C2 à la distance (D_s), par contre, les sorties du contrôleur sont la vitesse de braquage (W) et la vitesse d'avancement V_a .

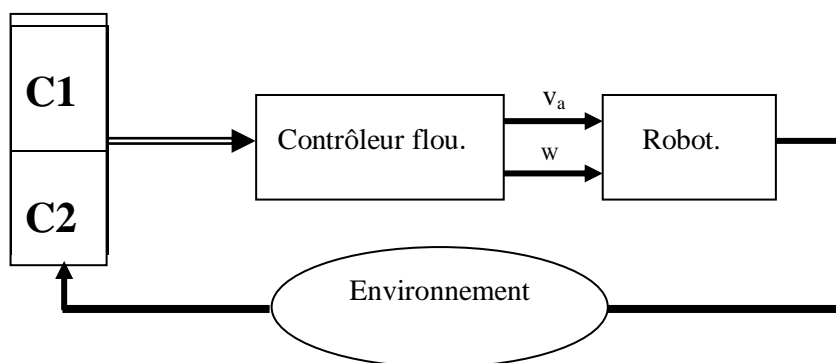


Figure IV.2: Schéma de contrôle utilisé.

IV.2.3 Définition des fonctions d'appartenance:

➤ *Fonctions d'appartenance d'entrées*

L'univers de discours des variables d'entrée du contrôleur d'évitement d'obstacles est décomposé en trois sous ensembles flous. Comme dans le premier contrôleur, ces derniers sont caractérisés par des fonctions d'appartenance triangulaires. Les nuances linguistiques utilisées pour les distances sont : Pro (proche), Moy (moyennement proche) et loin (loin) figure (IV.3).

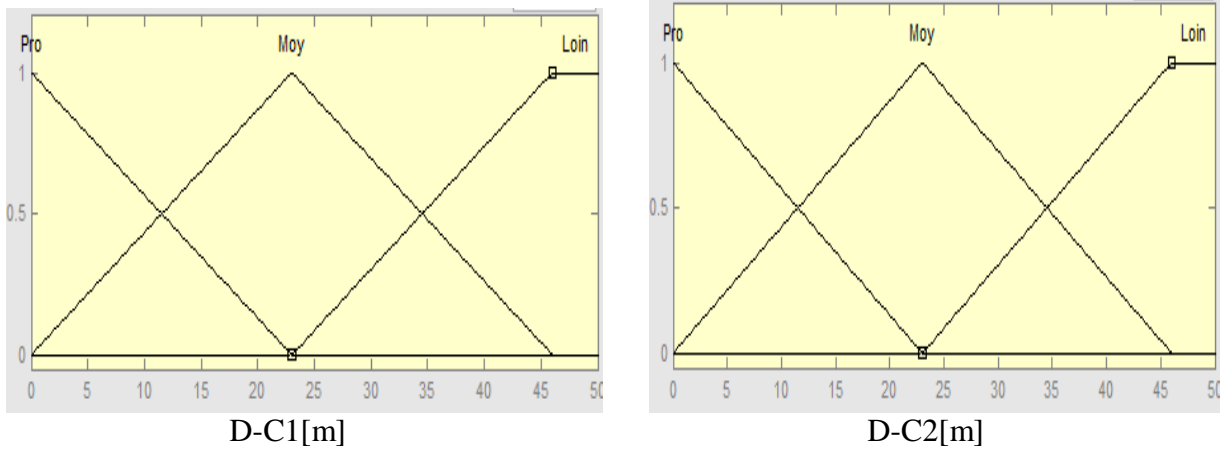


Figure IV.3 : les fonctions d'appartenances des variables d'entrée du contrôleur « évitement d'obstacles »

➤ *Fonctions d'appartenance de sorties :*

Les univers de discours des variables de sortie du contrôleur (vitesse d'avancement V_a et vitesse de braquage W), sont partitionnées comme suit : la première variable est partitionnée en trois sous-ensembles flous : Lent(zéro), Rap (rapidement) et TRap (très rapidement) et la deuxième variable est partitionnée en cinq sous-ensembles flous : DrRap (droit rapide), DrLent (droit lent), Z (zéro) et GaLent (gauche lent) et GaRap (gauche rapide) figure (IV.4)

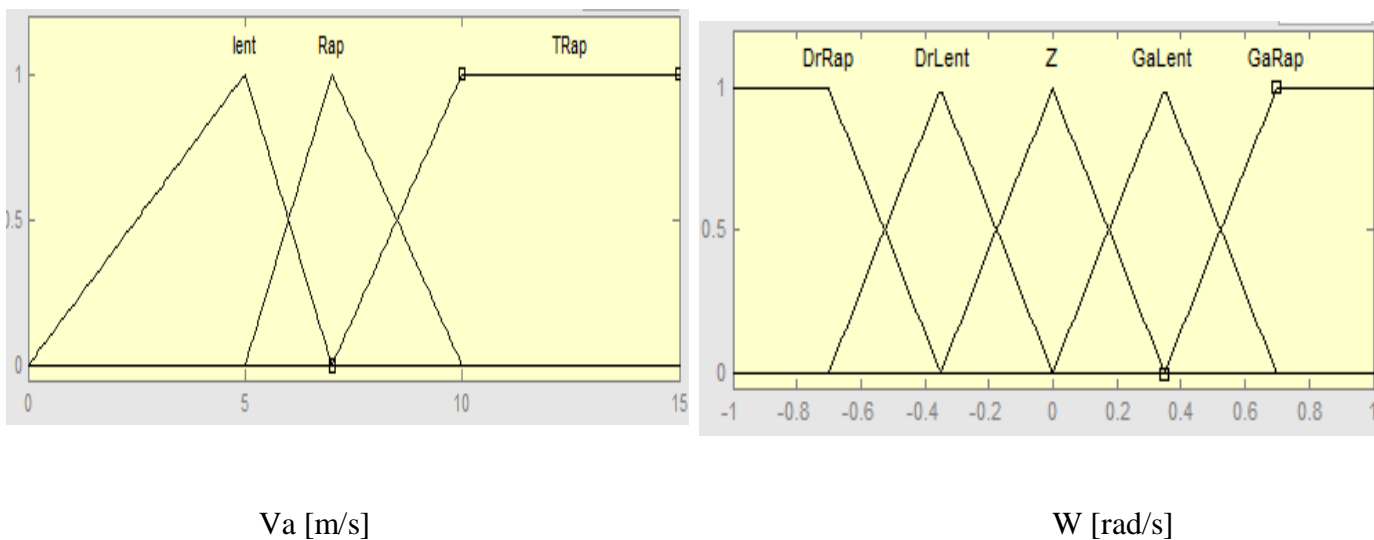


Figure IV.4 : Les fonctions d'appartenance des variables de sortie du contrôleur d'évitement d'obstacles.

IV.2.4 Base de règles :

Le contrôleur flou utilisé est de type Mamdani et l'étape d'inférence floue fait appel à des règles floues faisant un lien entre les variables d'entrée D-C1 et D-C2 et les variables de sortie Va et W. Les règles floues sont des expressions formées de prémisses (conditions) et de conclusions (actions) de la forme:

➤ **Si D-C1 est Moy et D-C2 est L alors W est GaLent et Va est Rap.**

Les deux tableaux ci-dessous représentent la base de règles floues pour les deux variables de sortie **Va** Tableau (IV.1) et **W** tableau (IV.2), et ce, pour toutes les combinaisons possibles des deux variables d'entrée. Ces deux tables sont établies sur la base du raisonnement suivant : à partir de sa position initiale, le robot s'oriente vers la cible en évitant les obstacles se trouvant sur son trajet et converge vers elle avec une vitesse proportionnelle à la distance robot-cible pour s'arrêter une fois la cible atteinte avec une marge d'erreur admissible.

D-C2 D-C1	Pro	Moy	L
Pro	Lent	Lent	Lent
Moy	Lent	Rap	Rap
L	Lent	Rap	TRap

Tableau IV.1 Base de règles floues de la vitesse d'avancement.

D-C1 D-C2	Pro	Moy	L
Pro	DrRap/GaRap	GaRap	GaRap
Moy	DrRap	Z	GaLent
L	DrRap	DrLent	Z

Tableau IV.2 base de règles floues de la vitesse de braquage.

III.3 Fusion de comportements:

Le comportement obtenu avec le contrôleur de convergence vers la cible et celui obtenu avec le contrôleur d'évitement d'obstacles fonctionnent indépendamment. Si les deux contrôleurs sont employés ensemble, il arrive qu'à l'approche d'un obstacle, ils génèrent des consignes contradictoires. De telles situations ne peuvent être gérées qu'en fusionnant les deux comportements dictés par ces derniers afin de diriger le mouvement du robot de telle manière que l'action de convergence vers une cible n'amène pas le robot à une collision avec un obstacle, et l'action d'évitement d'obstacle ne le fait pas dévier de sa trajectoire de convergence vers la cible.

III.4 Résultats de simulation :

Les tests de simulation effectués, ont montré que le robot évolue dans l'environnement où se trouvent deux obstacles éparpillés. A chaque fois qu'un obstacle est détecté par l'un des capteurs, le contrôleur d'évitement d'obstacles réagit en générant une consigne qui permet au robot d'éviter ce dernier. La trajectoire réalisée par le robot est donnée par les figures ci-dessous.

Les résultats obtenus montrent bien que le contrôleur d'évitement d'obstacles garantit une navigation saine et l'arrivée à la cible, dans les figures qui suivent, on montrera comment le robot munit d'un contrôleur conçu auparavant peut atteindre la cible sans collision avec les obstacles.

On donne quelques exemples qui simulent et testent notre contrôleur flou dans les différents cas en vérifiant les résultats obtenus :

1^{er} cas: trajectoire du robot entre les deux obstacles, évitement du 1^{er} obstacle par sa gauche

Point départ $\begin{cases} x_0 = 0 \\ y_0 = 0 \end{cases}$ Point arrivée $\begin{cases} x_f = 80 \\ y_f = 80 \end{cases}$ obstacle1 $\begin{cases} x = 30 \\ y = 20 \end{cases}$ obstacle2 $\begin{cases} x = 60 \\ y = 60 \end{cases}$

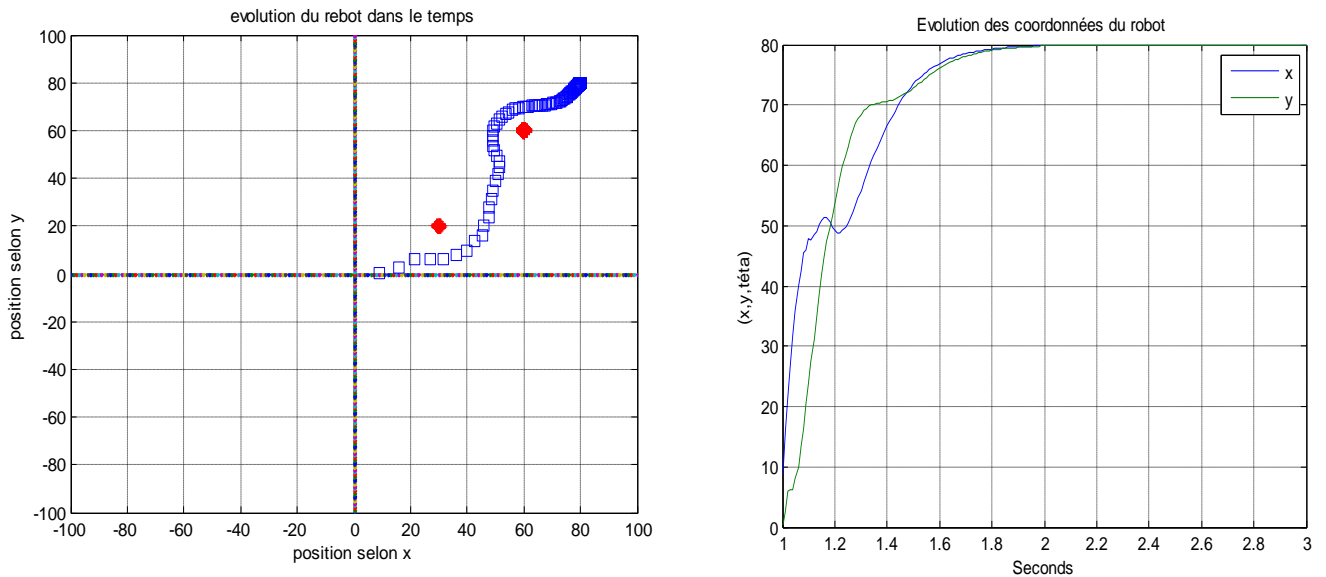


Figure IV.5 : navigation du robot dans un environnement avec obstacle : 1^{er} exemple

2^{eme} cas: trajectoire du robot entre les deux obstacles, évitement du 1^{er} obstacle par sa droite

Point départ $\begin{cases} x_0 = 0 \\ y_0 = 0 \end{cases}$ Point arrivée $\begin{cases} x_f = 60 \\ y_f = 80 \end{cases}$ obstacle1 $\begin{cases} x = 20 \\ y = 25 \end{cases}$ obstacle2 $\begin{cases} x = 40 \\ y = 60 \end{cases}$

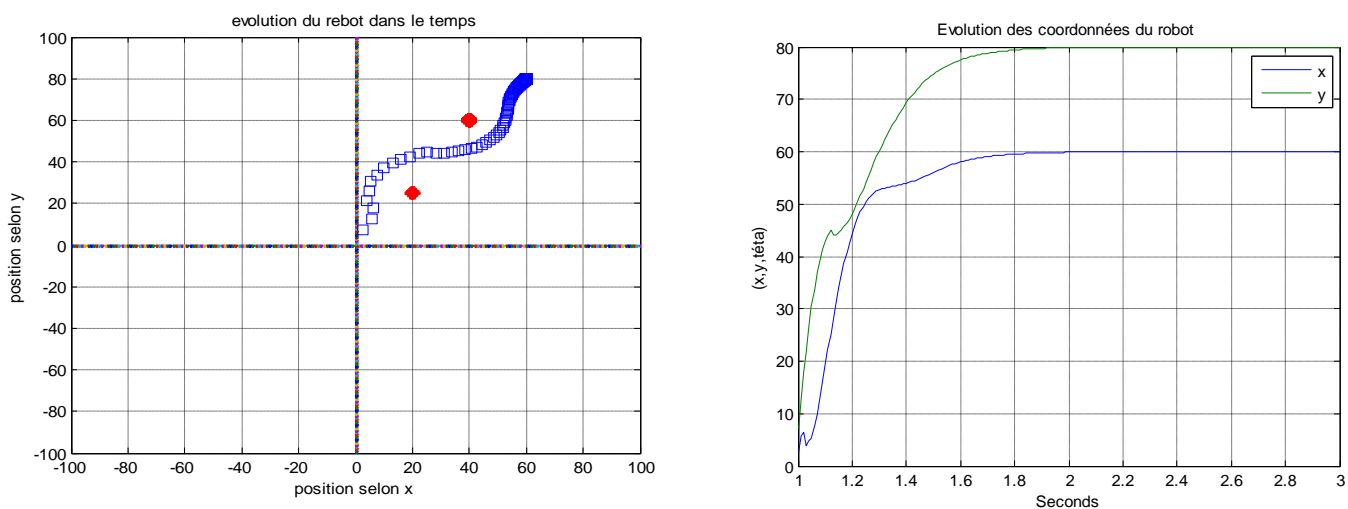


Figure IV.6 : Navigation du robot dans un environnement avec obstacle : 2^{eme} exemple

3^{ème} cas: évitement des obstacles par sa droite

$$\text{point départ } \begin{cases} x_0 = 20 \\ y_0 = 40 \end{cases} \text{ point arrivée } \begin{cases} x_f = 40 \\ y_f = 70 \end{cases} \text{ obstacle1 } \begin{cases} x = 30 \\ y = 60 \end{cases} \text{ obstacle2 } \begin{cases} x = 35 \\ y = 60 \end{cases}$$

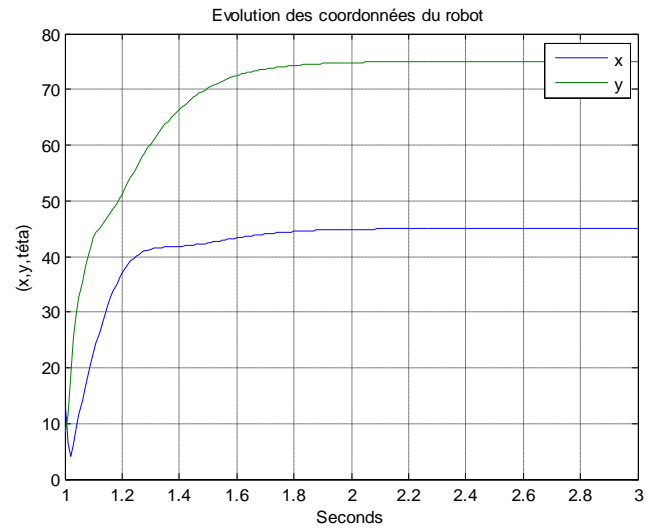
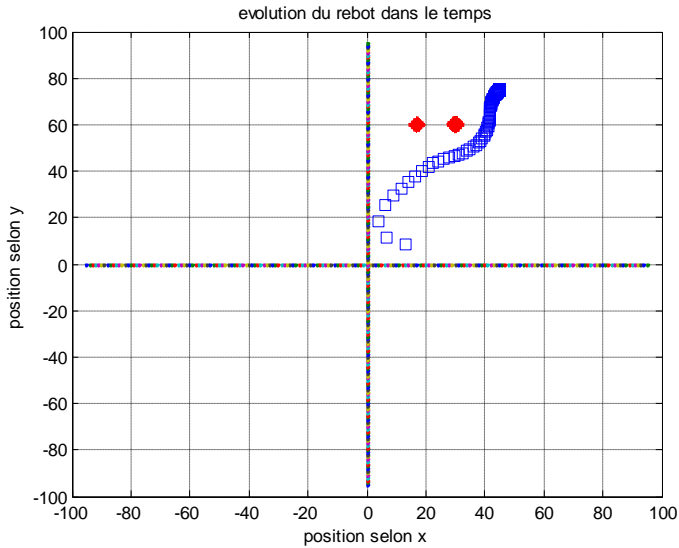


Figure IV.7 : navigation du robot dans un environnement avec obstacle : 3^{ème} exemple

4^{ème} cas: évitement des obstacles par sa gauche

$$\text{Point départ } \begin{cases} x_0 = 0 \\ y_0 = 0 \end{cases} \text{ point arrivée } \begin{cases} x_f = 45 \\ y_f = 75 \end{cases} \text{ obstacle1 } \begin{cases} x = 17 \\ y = 25 \end{cases} \text{ obstacle2 } \begin{cases} x = 30 \\ y = 60 \end{cases}$$

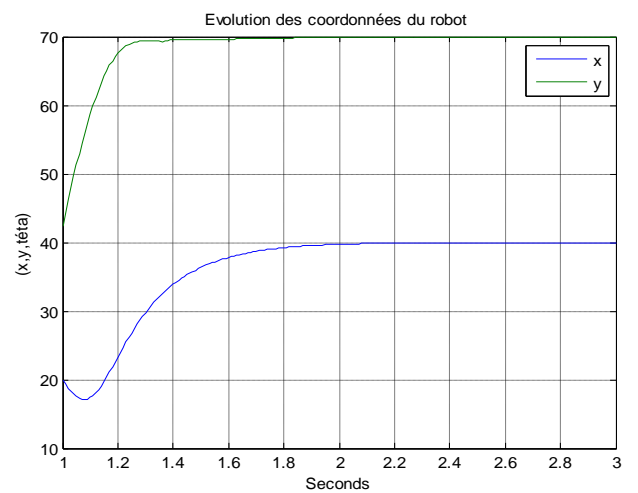
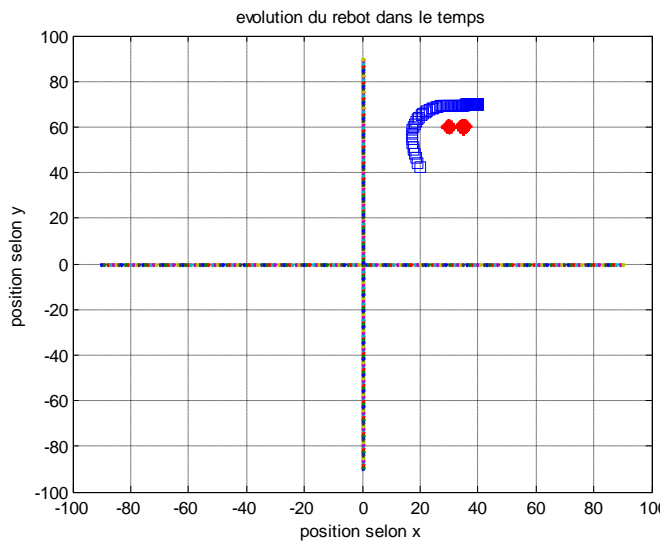


Figure IV.8 : navigation du robot dans un environnement avec obstacle : 4^{ème} exemple

IV.5 Conclusion :

Les systèmes à base de la logique floue ont un grand degré d'autonomie et une capacité de déterminer la conclusion (action) en temps réel plus proche du mode de raisonnement humain. Les résultats obtenus après l'implémentation de ce contrôleur flou ont été très satisfaisants, car le robot peut se déplacer d'une situation initiale vers une situation finale, tout en évitant les obstacles fixes à n'importe quel environnement.

Nous avons exploré le concept de navigation floue d'un robot mobile dans un environnement inconnu, et ce, par une approche comportementale à savoir : le comportement de convergence du robot mobile vers une cible avec la prise en compte des obstacles grâce à la fusion des deux comportements afin d'assurer une navigation ayant comme objectif l'atteinte de cibles, toute en garantissant la non collision du robot avec les obstacles rencontrés sur son trajet.

Conclusion Générale

Le travail présenté dans ce mémoire porte sur la navigation réactive floue, basée sur l'approche comportementale, d'un robot mobile dans un environnement inconnu.

La première partie concerne la conception d'un contrôleur floue et retour d'état permettant au robot mobile de converger vers une cible. Dans les deux tests, le comportement de convergence vers la cible a été simulé en considérant un environnement sans obstacles. Les résultats obtenus sont assez satisfaisants. Sauf que la nature de l'environnement considéré est loin d'être réelle.

Dans le souci de simuler le comportement du robot dans un environnement proche du réel, un deuxième test a été effectué. Dans ce dernier, l'environnement considéré comporte des obstacles de formes circulaires. La détection de ces obstacles est assurée par un module de perception. Ce dernier transmet les états des capteurs au contrôleur flou dit d'évitement d'obstacles. Ce dernier génère les deux sorties (vitesse de braquage et vitesse d'avancement) permettant de dévier le robot des obstacles rencontrés. Ce second comportement, utilisé seul, garantit la non collision de robot avec les obstacles rencontrés sur sa trajectoire, mais sans pouvoir autant garantir l'atteinte d'une cible. Un module, exploitant les données issues des capteurs et la position du robot par rapport à la cible est alors implanté assurant la fusion des ces deux comportements. Les résultats obtenus montrent que le robot atteint les cibles sans collision avec les obstacles.

Comme perspectives, l'étude proposée peut être complétée pour améliorer les performances de l'architecture adoptée en lui intégrant d'autres modules assurant l'intégration de comportements de plus haut niveau tel que :

- Perception visuelle
- Evaluation de l'environnement
- Elaboration de cartes, pouvant servir pour planifier des missions futures ou caractérisant la dynamique de l'environnement.

Référence Bibliographique

- [BEN 97] BENHALIMA Noreia et BELLAHCENE Latéfa, «*Maquette logicielle pour la simulation de la télé opération d'un robot mobile à l'aide d'un joystick*», Mémoire d'ingénieur USTO, 1997.
- [BOR 98] P.Borne, «*Introduction à la commande floue*», Edition Technip, 1998.
- [CHO 98] A.Chohra, «*Planification et contrôle de la navigation des Véhicules Autonomes Intelligents (VIA) en environnements dynamiques*», Thèse de doctorat d'état en électronique université polytechnique d'Alger, 1998.
- [GIA 00] Frédéric GIAMARCHI, «*Petits robots mobiles*», Edition Technique et Scientifiques Françaises (ETSF), 2000.
- [LIM 91] Komol LIMSAKOUNE, «*Étude de la modélisation des robots mobiles et de leur environnement; Application du développement d'un logiciel CAO une mission d'optimisation*», Thèse de Doctorat de l'université de Paris 6, Mai 1991.
- [MAL 11] Mallem Ali «*Algorithmes d'aide à la navigation d'un robot mobile*», Mémoire magister de l'université de Batna ,2011.
- [MEN 12] Mendil Boubekour, «*Robotique II Robotique Mobile*» cour de Robotique partie II chapitre 02,2012
- [MEN 12] Mendil Boubekour, «*Robotique II Navigation* » cours de Robotique partie II chapitre 04,2012
- [PRU 88] Alain BRUSKI, «*Robotique générale*», Edition Ellipses Paris, 1988.
- [GUE 03] Guenounou Ouahib, «*Optimisation des contrôleurs flous par algorithmes génétiques hiérarchisés. Application sur un actionneur asynchrone.* », Mémoire magister de l'université de Bejaia ,2003.
- [SOU 09] Soualmi Boussaad, «*Planification de trajectoires dans un environnement partiellement connu a basse des techniques d'intelligence artificielle* » Mémoire magister de l'université de Bejaia ,2009.
- [ZAD 65] L.A.Zadeh, «*Fuzzy sets*», Information and Control, 1965.
- [FIL 09] David Filliat, «*Cours de Robotique Mobile* », Ecole Nationale Supérieure de Techniques Avancées de Paris, 2009.

RESUME

Le travail présenté dans ce mémoire porte sur la navigation réactive floue, basée sur l'approche comportementale, d'un robot mobile dans un environnement inconnu.

La première partie concerne la conception d'un contrôleur flou et retour d'état permettant au robot mobile de converger vers une cible. Dans les deux tests, le comportement de convergence vers la cible a été simulé en considérant un environnement sans obstacles. Les résultats obtenus sont assez satisfaisants. Sauf que la nature de l'environnement considéré est loin d'être réelle. Dans le souci de simuler le comportement du robot dans un environnement proche du réel, un deuxième test a été effectué. Dans ce dernier, l'environnement considéré comporte des obstacles de formes circulaires. La détection de ces obstacles est assurée par un module de perception. Ce dernier transmet les états des capteurs au contrôleur flou dit d'évitement d'obstacles. Ce dernier génère les deux sorties (vitesse de braquage et vitesse d'avancement) permettant de dévier le robot des obstacles rencontrés