



République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université Abderrahmane Mira-Bejaia

Faculté de Technologie

Département de génie électrique

Mémoire de Fin de Cycle

En vue de l'obtention du diplôme **MASTER** en **Electronique**
option : Automatique

Thème

Etude comparative de quelques algorithmes
d'évitement d'obstacles en robotique mobile

Par : Ishak MOUMENE

Encadré et dirigé par :

• *MR. Boubkeur MENDIL*

Devant le jury composé de :

• *Mr. K. BOUDELLABA*

• *Mme. S. MEZZAH*

Promotion 2013-2014

Avant propos

Honneur à ceux qui sont à l'origine de ce travail, je parle bien évidemment de Mr. Boubkeur MENDIL professeur à l'université de Bejaia sans son aide, sans sa confiance qu'il m'a accordée, ma reprise à l'université n'aurait pas eu lieu, car il m'a encouragé à mainte reprise de reprendre les études après une lourde déception au courant de cette année.

Je lui exprime ma sincère reconnaissance pour l'honneur qu'il m'a fait en acceptant ce travail et de le valider.

Mes remerciements vont également à tous ceux qui ont permis de loin ou près, de contribuer à l'aboutissement de ce travail et à tous les chargés de cours de cette promotion qui m'ont encouragés à reprendre tous, sans exception.

Un remerciement particulier pour mes parents, mes frères, mes camarades, ainsi que tout ma famille pour m'avoir donné la force d'aller au bout de ce travail, pour leur patience et leur soutien.

Liste des tableaux

<i>Tableau.II.1. Base de règles floues de la vitesse d'avancement « V_{ac} »</i>	36
<i>Tableau.II.2. Base de règles floues de la vitesse de braquage « W_c »</i>	36
<i>Tableau.III.1. Base de règles floues de la vitesse d'avancement « V_{aob} »</i>	43
<i>Tableau.III.2. Base de règles floues de la vitesse de braquage « W_{ob} »</i>	43

Introduction générale

L'un des défis de l'homme aujourd'hui est de copier la nature et de reproduire des modes de raisonnement et de comportement qui lui sont propre. Cette envie a fait naître le concept d'intelligence artificielle. Ce concept est étroitement lié à la robotique.

Les premiers robots ont été caractérisés par leur aptitude de manipulation d'objets ce qui a permis de les intégrer dans des unités de production pour réaliser des tâches répétitives. Le premier objectif de leurs conceptions a été atteint : remplacer l'homme dans des tâches pénibles. [1]. L'évolution des robots de type manipulateur a permis d'accroître leur vitesse, d'augmenter les charge transportées, de fiabiliser les structures mécaniques. et de les rendre de plus en plus adaptables aux tâches demandées. Ces premiers robots ont tous été reliés à un référentiel fixe et planés dans un environnement parfaitement structuré. Les progrès technologiques ont conduit vers la réalisation d'un robot tout à fait différent dans sa conception et dans le type de tâche à accomplir : le robot mobile. Ce dernier se caractérise par l'absence de lien mécanique avec un objet de référence ce qui ouvre magistralement la porte aux applications potentielles. La liberté de mouvement lui confère une autonomie qui lui permet de trouver d'autres utilisations que manufacturière (industrielle) [1].

L'une des problématiques dans le domaine de la robotique mobile est la navigation du robot tout en évitant les obstacles. Notre travail s'intéresse à la navigation d'un robot mobile à roues non-holonyme soumis à des roulements sans glissement.

Dans cette optique, l'une des facultés premières que doit intégrer un robot mobile autonome est la capacité de se mouvoir et de converger vers des lieux (cibles) qui lui sont indiqués au préalable sans aucune intervention externe. Ceci fait l'objet du chapitre deux.

De plus, dans un environnement partiellement connu ou inconnu, des objets peuvent surgir. Le robot doit être doté d'une seconde faculté qui lui permet d'éviter ces obstacles. [2]. Dans ce contexte, plusieurs applications ont été développées pour assurer une navigation réactive. Le chapitre trois présente notre étude de deux méthodes, à savoir : les méthodes de champs de potentiel et la logique floue.

Chapitre I

Introduction à la robotique mobile

I.1. Introduction

Avant de présenter notre travail, il est important de rappeler quelques définitions et notions de base de la robotique mobile, telles que : la définition d'un robot mobile et quelques exemples d'applications, les différents types de locomotion, l'architecture de système de commande ainsi que les notions de perception, localisation, planification du chemin, etc.

I.2. Concepts de base

I.2.1. Définition d'un robot mobile

Il existe diverses définitions du terme robot. Fondamentalement, on peut retenir 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 [3].

On distingue deux types de robots. Les *robots fixes* sont utilisés généralement dans l'industrie pour réaliser les tâches dangereuses, répétitives, de précision, etc. Les *robots mobiles* sont utilisés par exemple pour transporter des charges ou pour réaliser des tâches diverses : ménagères, militaires, etc.

I.2.2. Exemples d'application

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. Parmi les domaines concernés, citons :

- La robotique de service (hôpital, bureaux)
- La robotique de loisir (aibo, robot 'compagnon')
- La robotique industrielle ou agricole (entrepôts, récolte de productions agricoles, mines)
- La robotique en environnement dangereux (spatial, industriel, militaire)



Figure.1.1. Exemples de robots utilisés dans différentes applications.

I.2.3 Notions d'holonomie/non-holonomie

Les systèmes mobiles dit *non-holonomes* sont ceux que l'on rencontre le plus dans la vie courante (voiture particulière, bus, camion, ...etc.). Ces systèmes ont une structure mécanique relativement simple (des roues motrices, des roues directrices et des roues libres). Une roue peut avoir une, deux ou trois fonctions. Mais tous ces systèmes ont une caractéristique commune : la direction de la vitesse d'entraînement (vitesse linéaire) est imposée par la direction des roues directrices.

Les systèmes *holonomes* sont beaucoup plus rares dans notre vie quotidienne. Ils ont une structure mécanique complexe qui leur permet de se déplacer dans toutes les directions sans manœuvre.

La société Nomadic disparue en l'an 2000 ; a conçu un système mobile holonome : le XR4000 (Figure.I.2). Il dispose de 4 roues motrices et directrices montées comme des roues de chariot. La synchronisation des 8 axes (2 par roue, rotation et orientation) est assurée par une carte dédiée basée sur le microcontrôleur Motorola 68332 et des circuits FPGAs et la structure mécanique est composée d'engrenages coniques [4].



Figure.I.2. Le robot XR4000 de la société Nomadic.

I.3. Systèmes de locomotion

Un système de locomotion assure deux fonctions :

- La propulsion.
- L'appui sur le milieu.

Les modes de propulsions sont variées. Cela s'étend du moteur thermique ou électrique. Le milieu dans lequel évolue le mobile oriente très fortement la mécanique mise en œuvre pour le mouvement.

I.3.1 Les mobiles à roues

Les robots mobiles à roues sont les plus répandus actuellement. La raison est essentiellement la simplicité de conception du mécanisme. La plupart des robots mobiles opérationnels jusqu'à présent, évoluent sur sites aménagés : environnement intérieur ou sites industriels. Ces robots comportent, généralement, trois ou quatre roues. Dans les cas marginaux, ils peuvent prendre six roues voire plus. On peut distinguer plusieurs types de roues :

- Roues motrices.

- Roues motrices directrices.
- Roues libres.
- Roues libres directrices.
- Roues folles.

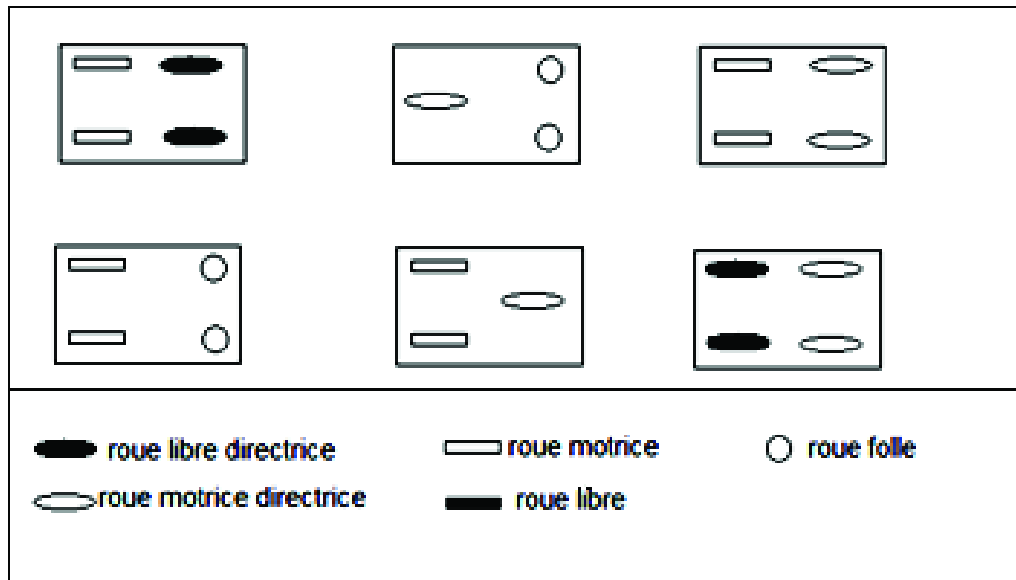


Figure.I.3. Les différentes configurations des roues possibles.

I.3.2 les mobiles à chenilles

Lorsque la configuration du terrain est plus chaotique, les engins à roues ont du mal à évoluer. Les chenilles sont alors plus performantes (Figure.I.4). Elles permettent d’augmenter l’adhérence au sol. La direction de conduite est définie en fonction de la différence entre les vitesses imposées aux chenilles.



Figure.I.4. Les robots à chenilles.

I.3.3 Les mobiles à pattes

Lorsque le terrain devient encore plus incertain, c'est-à-dire avec de grandes différences d’amplitudes comme un sol jonché de rochers, des engins à roues ou à chenilles

ne sont plus efficaces. Dans ce contexte, les mobiles ayant des points d'appuis discrets (Figure.I.5) sont la solution au problème du mouvement. Cette solution est universelle, puisque la très grande majorité des animaux terrestres se meuvent de cette manière. Mais la réalisation d'un tel système et l'établissement de sa commande est complexe.



Figure.I.5. Les robots à pattes.

Plusieurs autres types de robots mobiles existent, tels que les robots volants, les robots sous-marins, etc.

I.4. Architecture du système de commande

L'objectif principal d'un robot mobile consiste à réaliser le mouvement d'une manière autonome. L'exécution de cette tâche nécessite l'utilisation d'un ensemble de ressources et d'une structure assurent une coopération efficace entre elles. Les performances et les capacités du robot dépendent à la fois de la qualité des ressources et à leur gestion [5].

L'architecture du robot décrit l'organisation des divers modules qui le compose ainsi que leurs interactions. Les modules peuvent être retirés à volonté, selon la tâche envisagée. En général, l'informatique à bord est distribuée. Ce qui permet un fonctionnement indépendant et parallèle des modules.

Nous pouvons décomposer un robot mobile en quatre niveaux :

- ✓ Niveau interface homme-machine.
- ✓ Niveau planification de tâches.
- ✓ Niveau contrôle.
- ✓ Niveau fonctionnel.

I.4.1. Niveau interface homme-machine

Ce niveau d'architecture gère les communications avec le programmeur ou l'ordonnateur de tâche. L'opérateur décrit selon le mode de communication (écran/clavier, écran/souris, communication sans fil) la tâche que doit exécuter le robot. Celle-ci sera soumise, selon une forme adéquate, aux générateurs de tâches. Des accusés de réception des résultats de traitement ou éventuellement des précisions sur la tâche demandée sont transférés vers l'homme, selon toute forme de support visuel ou sonore.

I.4.2 Planificateur (Générateur) de tâches

A ce niveau de l'architecture, se construit le plan de la tâche à réaliser. Le générateur de plan va déterminer une suite séquentielle d'actions, en utilisant les ressources disponibles et en définissant les modes d'activation. La génération de plan repose sur plusieurs notions:

- ✓ La connaissance de l'état initial.
- ✓ La stabilité des connaissances en cours d'exécution.
- ✓ La connaissance parfaite de l'impact des actions.

Si ces trois notions sont valides, alors le plan d'action est généralement simple. La planification est nettement plus délicate lorsque une ou plusieurs des notions ci-dessus n'est plus vérifiée. Dans ce cas, le déroulement du plan, ou encore l'élaboration du plan, doit subir des modifications par rapport d'informations supplémentaires en cours d'exécution par reconnaissance de la situation qui est établie selon deux types d'applications :

- Si le robot est télé- commandé, alors l'opérateur intervient pour interpréter les informations en sa possession et pour prendre les décisions concernant la modification du plan.
- Si le robot possède une forte autonomie, alors la reconnaissance de situation consiste à identifier des propriétés courantes de l'environnement utiles à la mission.

L'autonomie d'un robot peut s'étendre aux trois notions suivantes :

l'action réflexe, la récupération d'anomalies et la re-planification.

L'action réflexe consiste à faire appel à une ressource sans qu'elle soit prévue par le plan original. Cet appel est réalisé au vue d'une situation particulière. L'exécution de cette

action a pour objet de se dégager des situations locale en espérant se rapprocher au plan initial. L'exemple le plus typique est le contournement des obstacles. La conséquence de l'introduction de cette action est double : soit la réaction permet de se dégager de problème, soit elle conduit vers une situation non récupérable. Le plan d'origine devient alors caduc.

La non-récupération du plan est souvent due à un manque de connaissance de la situation. Il est nécessaire d'utiliser toutes les ressources disponibles afin d'analyser la situation courante. Une bonne connaissance de la situation accroît fortement les chances de *récupération de l'anomalie*. On parle de récupération ou de planification locale.

La troisième notion de réaction concerne la re-planification. Le robot se trouve dans le cas extrême où le plan initial a échoué. Les actions réflexes et la récupération d'anomalies sont inopérantes. Dans ce cas, si la situation courante est connue, il est possible de re-planifier la mission.

I.4.3 Le contrôleur d'exécution

Le contrôle d'exécution orchestre la mise en œuvre du plan établi au niveau du planificateur. Lors du déplacement, il adapte le comportement du robot aux situations instantanées. Ainsi, si les situations successives concordent avec le plan établi, celui-ci se déroule comme prévu. Dans le cas contraire, le contrôleur applique la méthode convenue. Les informations nécessaires à la réactivité, c'est-à-dire l'application des comportements face à une situation, sont issues d'un module de surveillance de l'environnement. Ce dernier se compose de plusieurs sous-modules travaillant en parallèle.

I.4.4 Le niveau fonctionnel

Le niveau fonctionnel est le plus bas dans la hiérarchie de l'architecture. Il se compose de modules capteurs, effecteurs et actions de base. Les modules capteurs sont organisés en plusieurs niveaux : les éléments de détection (c'est-à-dire, les capteurs physiques) et l'élément de traitement de signal élémentaire (ex : extraction de primitives).

L'utilisation d'information s'effectue selon les besoins. Pour l'évitement d'obstacles, l'information télémétrique est suffisante.

Un module effecteur assure la réalisation du mouvement du robot. Comme pour le module précédent, celui-ci se décompose en plusieurs niveaux selon le degré de complexité de la consigne (ex : garder une orientation constante, suivre la droite défini par deux points...).

Le module des actions de base se compose d'un ensemble de primitives dans l'usage est fréquent. Nous pouvons citer notamment l'évitement d'obstacles ou le suivi de murs.

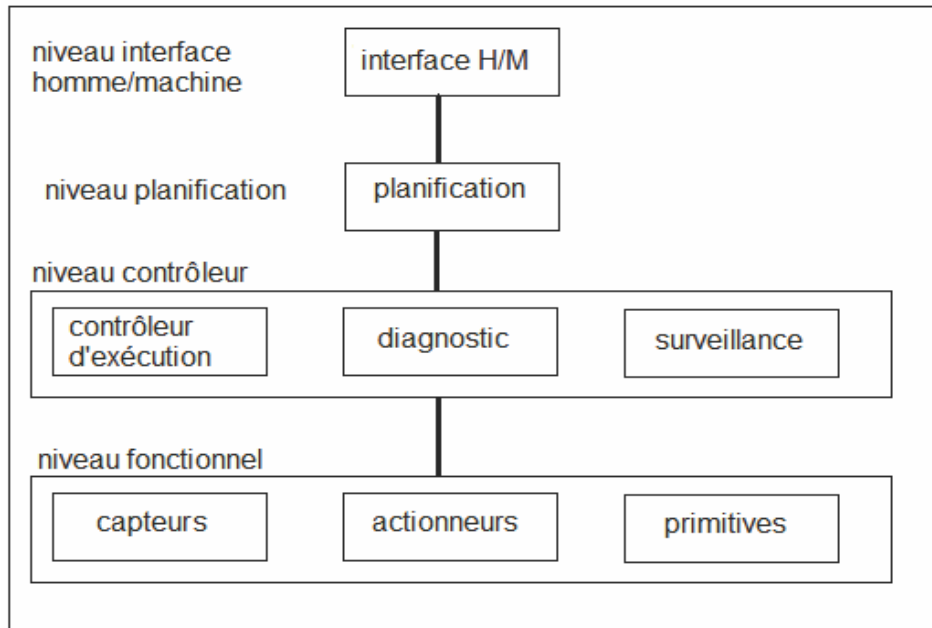


Figure.I.6. Architecture d'un robot mobile.

I.5. Navigation

La navigation autonome d'un robot est la capacité à évoluer sans aide dans leur environnement de travail (espace de configuration). La complexité de la méthode de navigation mise en œuvre sur un robot mobile dépend donc de l'environnement dans lequel doit évoluer (milieu intérieur ou environnement naturel, sol plan ou irrégulier,...). Elle dépend également de la connaissance de cet environnement qui peut être figé ou évolutif et du mode de définition de la trajectoire (apprentissage préalable, planification en ligne, ...).

Les performances du système de navigation sont étroitement liées à la précision, à la fiabilité et au temps de réponse des capteurs et des méthodes mises en œuvre pour localiser le véhicule [6]. La navigation des robots mobiles s'articule autour de trois niveaux principaux :

I.6. La localisation

Le caractère principal d'un robot mobile est la faculté de se mouvoir d'un point vers un autre. Pour ce faire, il est nécessaire d'avoir la connaissance de sa localisation par rapport à un espace de référence dans lequel sont définis les point source du but. De même, la poursuite d'une trajectoire prédéterminée suppose la connaissance instantanée de sa position. Nous pouvons parler de localisation statique lorsque le calcul de la position s'effectue à l'arrêt. La localisation dynamique est évaluée durant le mouvement. Nous

pouvons considérer deux grands systèmes de localisation : la localisation relative et la localisation absolue.

I.6.1 La localisation à l'estime (relative)

La localisation relative permet de déterminer la configuration courante du robot à partir de sa situation antérieure. Elle consiste à déterminer la position par l'intégration de mesure de vitesse ou d'accélération. Les capteurs fournissant ces données sont les odomètres ou gyromètres pour les mesures d'accélération.

Les odomètres sont présents sur l'ensemble des robots mobiles à roues. Cette technique peu coûteuse consiste à fixer sur les roues des codeurs délivrant une impulsion toutes les fractions de tour de roue. L'intégration de ces valeurs permet de déduire la position et l'orientation du mobile par rapport à l'initialisation des compteurs d'impulsion. Cette technique délivre des mesures peu fiables dans le temps.

Les techniques inertielles (accéléromètre, gyromètre) plus coûteuses constituent le haut de la gamme des systèmes de localisation à l'estime. La connaissance de la position référencée à l'origine du déplacement nécessite une double intégration de l'accélération. Ces calculs entraînent une inévitable accumulation d'erreurs qui constitue une dérive d'estimation dans le temps. Un recalage périodique est alors indispensable.

La technique de localisation à l'estime présente l'immense avantage d'être indépendante de l'environnement. Les seules erreurs qu'elle peut générer sont celles dues à son mode de fonctionnement interne. Par contre, l'inconvénient majeur est l'accumulation d'erreurs due aux différentes intégrations.

I.6.2 La localisation absolue

La localisation absolue est une technique qui permet à un robot de se repérer directement dans son milieu d'évolution, que ce soit en environnement extérieur (mer, espace, terre), ou en environnement intérieur (ateliers, immeubles, centrales nucléaires...).

Ces méthodes de localisation sont basées sur l'utilisation de capteurs extéroceptifs.

Pour répondre à la problématique qui est la localisation d'un robot dans son environnement, deux types de stratégies sont utilisables :

- La première consiste à utiliser des points de repère naturels.
- La deuxième à utiliser des points de repère artificiels.

Quelque soit le cas de figure, la localisation absolue nécessite toujours une représentation de l'environnement. Le robot possède donc «une banque de données» regroupant les caractéristiques des références externes qui est appelée carte de l'environnement.

I.7. La perception

La capacité d'autonomie d'un robot est liée directement à la puissance d'appréhension de l'environnement. Que l'espace d'évolution soit structuré, non structuré ou semi-structuré, il est toujours possible que survienne aléatoirement un événement demandant de la part du robot une réaction qui se compose de plusieurs modules : la perception d'événement, la prise de la décision. [1]

La perception de l'événement constitue le rôle de porteur. Il va traduire un état physique en signaux électriques. La modélisation consiste à associer des représentations informatiques et/ou mathématiques aux informations délivrées par le capteur, ceux-ci dans le but de pouvoir les manipuler avec des outils disponibles.

La détermination du type d'événement consiste à trouver, à partir de modèles ou éventuellement de plusieurs modèles, les événements qui viennent d'être perçu.

Cette chaîne de traitement d'informations a pour objet de délivrer une commande vers la machine en réaction à des événements. Les réactions sont soit de type « ordre générique » (arrêt immédiat, prise de l'objet...) soit de type adaptatif (s'approcher, demande de complément d'information...) soit de type sans réaction (poursuit de tâche en cours...).

L'acquisition d'un état de l'environnement s'effectue selon deux types de principes : soit l'environnement émet une énergie propre détectable par un système physique (cas d'utilisation d'un capteur passif), soit il renvoie une énergie émise par une source artificielle (cas d'utilisation d'un capteur actif). L'information utile est alors contenue dans la forme de l'énergie réfléchi.

Pour réaliser un système de robotique mobile intelligent, il est nécessaire d'utiliser des capteurs qui fourniront la perception requise de l'environnement pour une prise de décision intelligente.

L'importance des capteurs peut être passablement influencée par l'environnement dans lequel le système est amené à évoluer. Le défi est donc de réaliser des systèmes qui utilisent le mieux possible des capteurs ou une combinaison de capteurs en tenant compte de leurs conditions d'utilisation, et ceci requiert une bonne connaissance de leurs caractéristiques.

I.7.1 capteurs proprioceptifs

Les capteurs proprioceptifs fournissent par intégration des informations élémentaires sur les paramètres cinématiques du système mobile. Les informations sensorielles gérées dans ce cadre sont généralement des vitesses, des accélérations, des angles de giration, des angles d'altitude. Cependant, ils ne peuvent pas procurer de renseignements lors de l'arrêt du système mobile.

On peut regrouper les capteurs proprioceptifs en deux familles: [7]

- Les capteurs de déplacement qui comprennent les odomètres, les accéléromètres et les radars Doppler. Cette catégorie permet de mesurer des déplacements élémentaires, des variations de vitesse ou d'accélération sur des trajectoires rectilignes ou curvilignes.
- Les capteurs d'attitude, qui mesurent deux types de données : les angles de cap et les angles de roulis et de tangage. Ils sont principalement constitués par les gyroscopes, les gyromètres, les gyrocompas, les capteurs inertiels composites, les inclinomètres et les magnétomètres. Ces capteurs sont en majorité de type inertiel.

I.7.1.1 Les odomètres

Ces capteurs fournissent une estimation en temps réel de la position (x, y) et de l'angle d'un véhicule navigant sur un sol plan, par rapport au repère de référence qui était celui du véhicule dans sa configuration précédente.

I.7.1.2 Les accéléromètres

Un accéléromètre est un capteur qui, fixé à un mobile ou tout autre objet, permet de mesurer l'accélération linéaire de ce dernier. On parle encore d'accéléromètre même s'il s'agit en fait de trois accéléromètres qui calculent les trois accélérations linéaires selon trois axes orthogonaux.

Par contre, lorsqu'on cherche à détecter une rotation ou vitesse angulaire, on parle de gyromètre. Plus généralement on parle de centrale inertielle lorsqu'on cherche à mesurer l'ensemble des six accélérations.

I.7.1.3 Radars Doppler

Un radar Doppler est un radar qui utilise l'effet Doppler-Fizeau de l'écho réfléchi par une cible pour mesurer sa vitesse radiale. Le signal micro-onde émis par l'antenne directionnelle du radar est réfléchi par la cible et comparé en fréquence avec le signal original aller et retour. Il permet ainsi une mesure directe et extrêmement précise de la composante vitesse de la cible dans l'axe du faisceau.

Les radars Doppler sont utilisés pour la défense aérienne, pour le contrôle du trafic aérien, pour la surveillance des satellites, pour les contrôles de vitesse sur route, en radiologie et dans les réseaux d'assainissement.

I.7.1.4 Le gyroscope

Un gyroscope est un capteur de position angulaire et un gyromètre un capteur de vitesse angulaire. Le gyroscope donne la position angulaire (selon un, deux ou les trois axes) de son référentiel par rapport à un référentiel inertiel (ou galiléen).

1.7.2 Capteurs extéroceptifs

Les capteurs extéroceptifs sont employés en robotique mobile pour collecter des informations sur l'environnement d'évolution du système mobile. Ils sont le complément indispensable aux capteurs proprioceptifs présentés précédemment.

Des méthodes de fusion de données sont alors utilisées pour conditionner et traiter les informations sensorielles de natures différentes. Ils sont notamment utilisés dans les domaines d'application tels que l'évitement d'obstacle, la localisation, la navigation et la modélisation d'environnements.

Les principaux capteurs utilisés en robotique mobile sont : les capteurs télémétriques (les ultrasons, les lasers et les infrarouges), le GPS et les caméras.

1.7.2.1 Les télémètres

Un télémètre est un appareil ou dispositif permettant par télémétrie de mesurer une distance. La télémétrie est un procédé (technique) permettant de calculer ou de mesurer la distance d'un objet lointain par utilisation d'éléments optiques, acoustiques ou radioélectriques (un télémétrique laser, par exemple).

1.7.3 Notion d'incertitude

Une information est incertaine, si elle est bruitée (mauvaises conditions de mesures), incomplète (obstruction d'un capteur ou portée limitée, absence d'informations sur l'évolution d'un objet ou d'un phénomène) ou imprécise (les glissements des roues par rapport au sol sont observables mais rarement mesurables avec précision) [1].

I.8. Méthodes de planification du chemin

Les méthodes d'évitement d'obstacles sont des méthodes locales dont l'objectif est d'assurer la sécurité du robot dans un temps très court. On ne donnera ici qu'un aperçu des méthodes de planifications utilisées en robotique mobile.

I.8.1 Méthodes délibératifs

Ce sont des méthodes qui calculent un ensemble de solutions potentielles compte-tenu des informations sur l'environnement, puis sélectionnent une solution particulière afin de satisfaire des contraintes de tâche. Les solutions peuvent prendre une forme de direction privilégiée du robot ou d'une consigne en vitesse.

I.8.1.1 Graphe de visibilité

Le principe de ces méthodes est de tenter de capturer la topologie de l'espace de recherche (espace de configuration ou espace d'état du système robotique) dans le but de simplifier le problème à une recherche dans un graphe. Elles sont donc constituées de deux étapes :

- Construction du graphe dans l'espace de recherche approprié.

- Parcours du graphe dans le but de déterminer un chemin ou une trajectoire entre les configurations initiale et finale.

Lorsque la complexité du problème est assez faible (faible dimensionnalité de l'espace de recherche), des approches combinatoires garantissant ainsi la complétude de l'algorithme de planification. La construction d'un graphe de visibilité [8], qui se trouve peut-être d'ailleurs être la toute première approche de planification de chemin connue, fait partie de celles-ci. Un graphe de visibilité (Figure.I.7) consiste à considérer chaque sommet des enveloppes convexes d'obstacles polygonaux, et à relier chacun de ces sommets à tout autre sommet visible de cet ensemble. On obtient ainsi un graphe dans lequel peut être effectuée une planification après avoir relié les positions initiale et finale aux sommets de cet ensemble les plus proches.

On note que cette technique autorise les configurations de contact entre le système mobile et les obstacles, c'est l'une des raisons pour lesquelles elle est relativement peu utilisée.

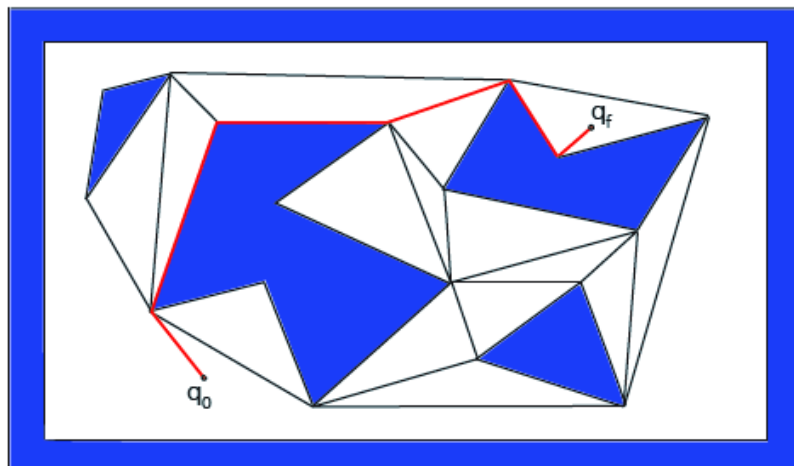


Figure.I.7. Chemin déterminé entre deux configurations q_0 et q_f à partir d'un graphe de visibilité.

I.8.1.2 Décomposition cellulaire

Une autre approche consiste en une décomposition cellulaire de l'espace de configuration. A partir d'une représentation simple de l'espace de configuration, cette approche consiste à diviser cet espace en un nombre fini de sous-espace convexes. Par exemple, dans un espace de configuration de dimension 2 dont la représentation des obstacles est polygonale (Figure.I.8), une décomposition cellulaire verticale [9] ou par triangulation [10] peut être facilement calculée. Un graphe est alors construit comme suit : les nœuds sont définis aux barycentres de sa cellules obtenues et au milieu de leurs côtés.

Une arête relie ensuite chaque nœud défini sur un côté au barycentre des deux cellules adjacentes. Comme précédemment, en reliant la configuration initiale q_0 avec la configuration finale q_f aux nœuds du graphe les plus proches, une recherche heuristique dans le graphe résultant permet de trouver un chemin liant ces deux configurations.

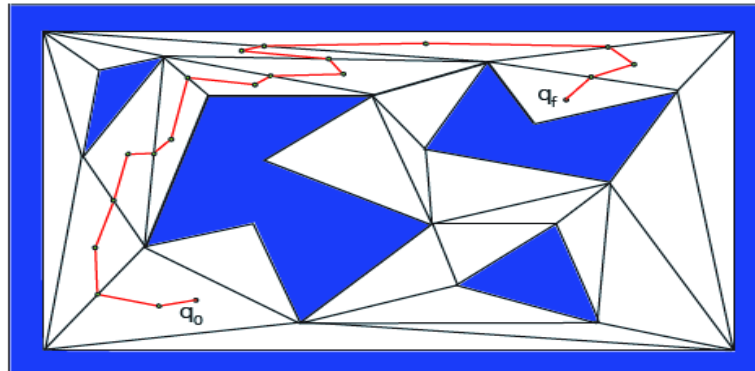


Figure.I.8. Chemin déterminé entre deux configurations q_0 et q_f à partir d'une décomposition cellulaire.

I.8.2 Méthodes replanification

En parallèle de la planification classique par exploration d'un graphe de recherche sont apparues les méthodes par arbres. Celle-ci consiste, à partir de la configuration initiale du système, à construire un arbre se développant dans toutes les directions autour du robot dans l'espace de recherche. Elles sont donc bien adaptées dans le cas d'espaces de recherche à forte dimensionnalité.

I.8.2.1 Le fil d'Ariane

Le fil d'Ariane présenté dans [11] explore l'espace de configuration du système robotique à partir de sa configuration initiale en construisant un arbre de recherche par l'alternance de deux étapes (Figure.I.9) :

- Explore : Cette étape a pour but d'explorer l'espace de configuration libre en y plaçant des balises aussi loin que possible des balises existantes. A l'initialisation, la seule balise disponible est la configuration initiale du système. A chaque nouvel appel de la méthode "Explore", l'arbre est étendu à partir d'une des balises existantes choisie aléatoirement.
- Recherche : Cette étape recherche autour d'une balise posée s'il est possible d'accéder directement à la configuration finale par un mouvement simple (chemin de Manhattan).

En répétant successivement ces deux étapes, l'arbre de recherche va s'étendre rapidement sur tout l'espace de configuration accessible à partir de la configuration initiale, jusqu'à converger vers le but, ou s'arrêter s'il n'est plus possible de placer une balise à moins d'une distance minimale de celles déjà posées.

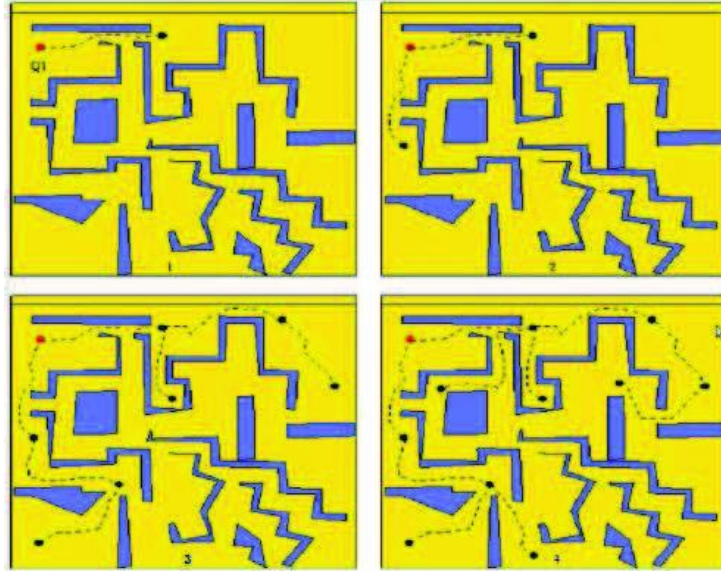


Figure.I.9. Le fil d'Arian, l'évolution de l'algorithme de planification alternant les deux étapes pour la planification d'un chemin.

I.8.3 Approche réactives

Les approches réactives consistent à calculer à chaque pas de temps (après récupération des informations sur l'environnement fournies par les capteurs du système) le contrôle instantané à appliquer sur les actionneurs du système.

I.8.3.1 Approche par champs de potentiel

Les approches dites par champs de potentiel initialement proposées par Khatib [12] consistent à considérer le robot mobile comme une particule soumise à divers champs électromagnétiques régissant son mouvement. Cette méthode s'appuie sur le calcul de deux champs de potentiel (Figure.I.10) :

- Un champ de potentiel attractif provenant de la position finale q_f du système à atteindre.
- Un champ de potentiel répulsif provenant des obstacles statiques et mobiles de l'environnement.

Initialement conçus pour le calcul du mouvement de bras manipulateurs, cette méthode dispose de l'avantage de calculer ces champs de potentiels dans l'espace de travail (espace euclidien \mathbb{R}^2 ou \mathbb{R}^3 dans lequel une représentation des obstacles est

disponible), définissant ainsi une direction privilégiée à suivre par l'élément terminal du bras dans cet espace. Une modification de la configuration du robot dans son espace articulaire est alors déduite de ce champ dans un second temps.

Malgré que cette méthode est simple et élégante mais elle possède de nombreux inconvénients mis en évidence par Koren et Borenstein [13]. D'une part, cette approche est sujette à des minima locaux. Par conséquent, la convergence vers le but n'est pas assurée. D'autre part, ces potentiels peuvent donner lieu à de fortes oscillations du mouvement du robot en présence d'obstacles et principalement lorsque celui-ci navigue dans des passages étroits (couloirs, portes). Enfin, le vecteur de déplacement désigné par le champ de potentiel ne prend en aucun cas en compte la cinématique ou la dynamique du système robotique considéré. Un robot disposant de contraintes non-holonomes aura de sérieuses difficultés à suivre une telle direction.

Malgré ces limitations, de nombreuses techniques de navigation ont découlé de ces champs de potentiels. Ils ont par exemple été adaptés à la navigation au milieu d'obstacles mobiles dans [14] en prenant en compte non seulement la distance aux obstacles mais également la vitesse de ces derniers pour calculer les champs répulsifs.

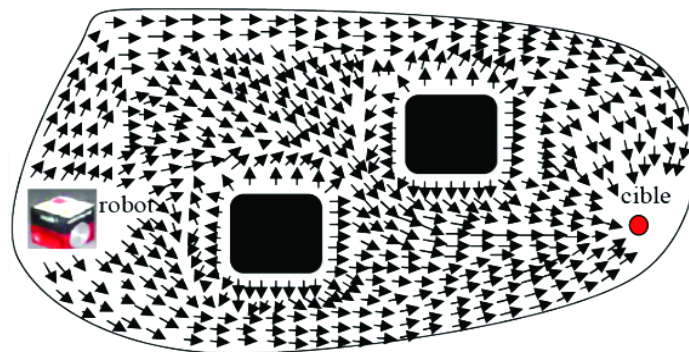


Figure.I.10. Calcul d'un chemin vers la cible par la méthode de champs de potentiels.

I.8.3.2 Déformation de la trajectoire

Les déformations de mouvement consistent, comme leur nom l'indique, à adapter le mouvement par rapport aux éventuelles modifications de l'environnement.

En absence des obstacles, une trajectoire initial est donnée à un contrôleur qui la discrétise et génère les commande qui permet au robot de l'exécuter. Mais l'apparition des obstacles (statique sur la trajectoire ou dynamique dont le chemin percute la trajectoire initiale) impose sur le contrôleur d'exécuter une séquence de commandes dans le but d'une part, de garantir la non collision avec les obstacles et

d'autre part de pouvoir rejoindre la trajectoire initiale une fois le risque de collision est dépassé.

I.9. Conclusion

Ce chapitre reprend quelques notions qui permettent la compréhension de notre travail, telles que : la robotique mobile, les systèmes de locomotion, l'organisation du système de commande, les notions de navigation, localisation, perception, etc. la dernière partie est dédiée aux méthodes de planification.

Chapitre II

Navigation dans un environnement sans obstacles

II.1. Introduction

Plusieurs méthodes ont été développées pour assurer la convergence vers la cible d'un robot mobile de façon autonome dans un environnement sans obstacles. Dans ce chapitre, deux méthodes, retour d'état et logique floue, ont été étudiées. Dans les deux cas, le contrôleur reçoit les coordonnées polaires de la cible, exprimées en termes de la distance entre le robot et de la cible, D_{cible} , et l'angle d'incidence ou d'orientation (angle entre la direction du robot et le vecteur robot-cible), φ_{cible} .

II.2. Méthode du retour d'état

Considérant le robot dans une posture initiale quelconque. L'objectif est de calculer le retour d'état permettant de placer le robot dans la posture finale désirée, $x_f = (x_f, y_f, \theta_f)^T$. Pour ce faire, on préfère utiliser le modèle en coordonnées polaires (*Figure.II.1.*). Ce qui nous permet d'exprimer la vitesse d'avancement en fonction de la distance et la vitesse d'orientation en fonction de l'angle d'incidence.

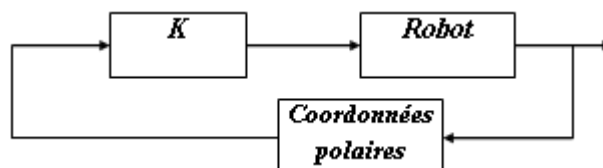


Figure.II.1. Schéma du système de commande

II.2.1. Modélisation cinématique

Ce paragraphe est consacré 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 jeux [15].

II.2.1.1. Posture du robot

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

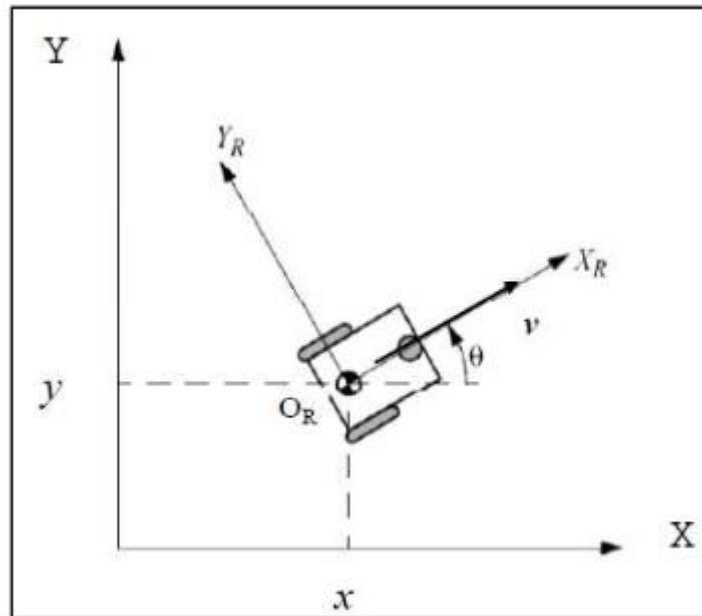


Figure.II.2. Situation du robot par rapport au repère de base.

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:

$$x_R = R(\Theta) \cdot x = \begin{bmatrix} \cos\Theta & \sin\Theta & 0 \\ -\sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot x \quad (1)$$

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

$$\dot{x}_R = R(\Theta) \cdot \dot{x} \quad (2)$$

II.2.1.2. 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{W}_c de la roue :

$$\vec{v}_Q = \vec{v}_P + \vec{W}_c \wedge \overrightarrow{PQ} = \vec{0} \quad (3)$$

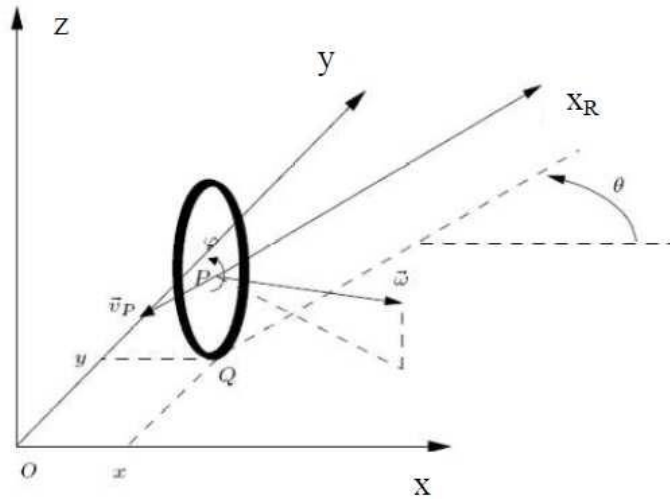


Figure.II.3. Caractérisation du roulement sans glissement

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

$$\vec{v}_Q = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ 0 \end{pmatrix} + \begin{pmatrix} \dot{\phi} \sin\Theta \\ \dot{\phi} \cos\Theta \\ 0 \end{pmatrix} * \begin{pmatrix} 0 \\ 0 \\ -r \end{pmatrix} \quad (4)$$

On obtient alors :

$$\vec{v}_Q = \begin{pmatrix} \dot{x} + r \cdot \dot{\phi} \cos\Theta \\ \dot{y} + r \cdot \dot{\phi} \sin\Theta \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (5)$$

Ce qui se traduit par les deux contraintes scalaires :

$$\dot{x} + r \cdot \dot{\phi} \cos\Theta = 0 \quad (6)$$

$$\dot{y} + r \cdot \dot{\phi} \sin\Theta = 0 \quad (7)$$

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 perpendiculaire 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 $r\dot{\phi}$ [15].

II.2.1.3 Robot mobile de type unicycle (différentiel) :

Une des configurations les plus utilisées pour les robots mobiles est la configuration différentielle (*differential drive*) qui comporte deux roues fixes non orientables commandées indépendamment. Une ou plusieurs roues folles sont ajoutées à l'avant ou à l'arrière du robot pour assurer la stabilité. On note qu'il existe certains robots différentiels avec quatre roues commandées indépendamment, sauf qu'ils sont modélisés en deux roues. Le schéma du robot différentiel est présenté à la *figure II.4*. On y a omis les roues folles, parce qu'ils n'interviennent pas dans la cinématique et assurent juste l'équilibre.[16]

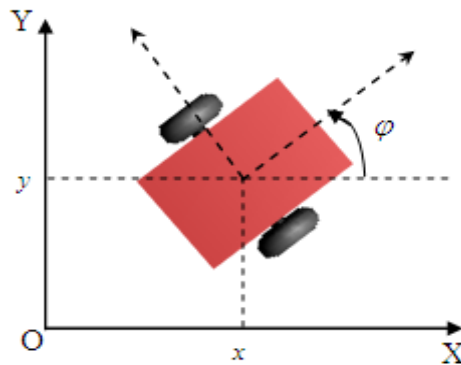


Figure.II.4. Robot mobile de type unicycle

Ce type de robot est très répandu en raison de sa simplicité de construction et de ses propriétés cinématiques intéressantes, comme sa capacité à tourner sur lui-même.

Les roues motrices ayant même axes de rotation, le CIR (centre instantané de rotation) du robot est un point de cet axe. Soit ρ le rayon de courbure de la trajectoire du robot, c'est-à-dire la distance du CIR au point O' (*Figure II.5*). Soit $2L$ la distance qui sépare

les deux roues, et W_c la vitesse angulaire du robot par rapport au CIR. Alors les vitesses de roues droite et gauche, respectivement notées v_d et v_g vérifient :

$$v_d = (\rho + L) W_c \quad (9)$$

$$v_g = (\rho - L) W_c \quad (10)$$

Ce qui permet de déterminer ρ et W_c à partir des vitesses de roues :

$$W_c = \frac{v_d - v_g}{2L} \quad (11)$$

$$\rho = R = L \frac{v_d + v_g}{v_d - v_g} \quad (12)$$

La vitesse linéaire v du robot au point O' est :

$$V_{ac} = \frac{v_d + v_g}{2} \quad (13)$$

L'équation (12) permet de situer le CIR sur l'axe des roues. Par ailleurs, ces équations expliquent deux propriétés particulières du mouvement des robots différentiels. Si $v_d = v_g$, la vitesse angulaire W_c sera nulle et le rayon de courbure ρ est infinie et le robot se déplace en ligne droite. Si $v_d = -v_g$, $W_c \neq 0$ et ρ nulle alors le robot effectue une rotation sur lui-même. Cependant dans le cas où $v_d \neq v_g$ le déplacement du robot est un virage à gauche ou à droite et cela, selon la supériorité ou l'infériorité de v_d par rapport à v_g (dans une direction qui correspond à la vitesse inférieure).

L'utilisation de ce mode de locomotion fournit une solution simple pour amener le robot d'une position à une autre. C'est sans doute l'une des raisons du succès de ce type de robots.

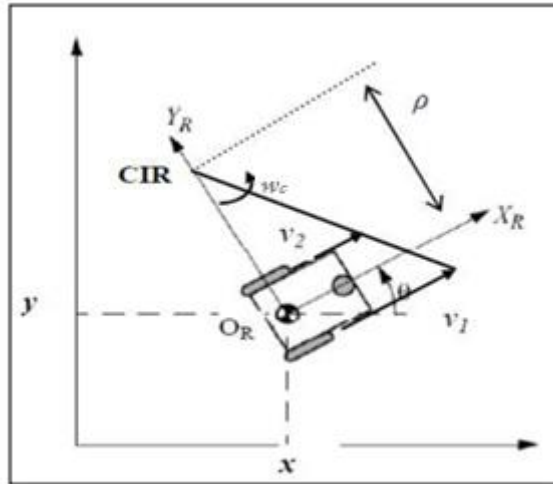


Figure.II.5. Centre instantané de rotation d'un robot mobile de type unicycle.

Le modèle cinématique du robot différentiel est donné par :

$$\dot{x} = v \cos \theta \quad (14)$$

$$\dot{y} = v \sin \theta \quad (15)$$

$$\dot{\theta} = W_c \quad (16)$$

Ces équations relient la dérivée de la posture $(\dot{x}, \dot{y}, \dot{\theta})$ du robot à la commande $u = (V_{ac}, W_c)^T$, avec w rotation instantanée du robot par rapport au repère (O, X, Y) . De ce fait, la position du robot est donnée par :

$$x(t) = \int_0^t v(\sigma) \cos(\theta(\sigma)) d\sigma \quad (17)$$

$$y(t) = \int_0^t v(\sigma) \sin(\theta(\sigma)) d\sigma \quad (18)$$

$$\theta(t) = \int_0^t \Omega(\sigma) d\sigma \quad (19)$$

II.2.2. Commande par retour d'état

Le modèle est donné par les équations suivantes :

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

Avec : $\Delta x = x_c - x$ et $\Delta y = y_c - y$

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

Ce qui donne le modèle suivant :

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

Avec :

d : distance entre le robot et le point cible.

$\alpha \in [-\pi/2, \pi/2]$: 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).

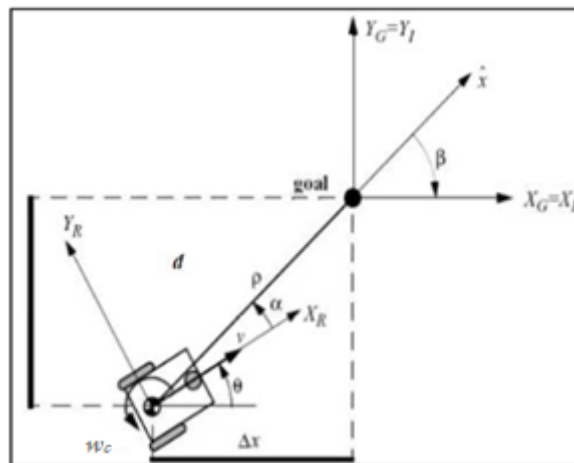


Figure.II.6. Distance et orientation du robot par rapport à la cible avec retour d'état

Considérant maintenant la loi de commande linéaire :

$V_{ac} = k_p d$: la vitesse d'avancement de robot.

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

dans cet exemple en prenant : $(k_p, k_a, k_b) = (3, 10, 1.5)$ [15]

Qui donnera le modèle cinématique suivant:

$$\begin{cases} \dot{x} = V_{ac} \cos(\theta) \\ \dot{y} = V_{ac} \sin(\theta) \\ \dot{\theta} = W_c \end{cases}$$

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 (II.1), d'où :

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

Tel que : T_e est la période d'échantillonnage.

II.2.3. Résultats de la simulation

D'après les résultats de la simulation, on constate que le robot s'oriente vers la cible avec une vitesse de braquage modérée et une vitesse d'avancement inversement proportionnelle à l'erreur d'orientation durant cette manœuvre. Quand le robot se rapproche de la cible, il décélère jusqu'à l'arrêt total, une fois le point cible est atteint.

1^{er} cas: Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d'arrivée $\begin{cases} x_c=40 \\ y_c=40 \end{cases}$

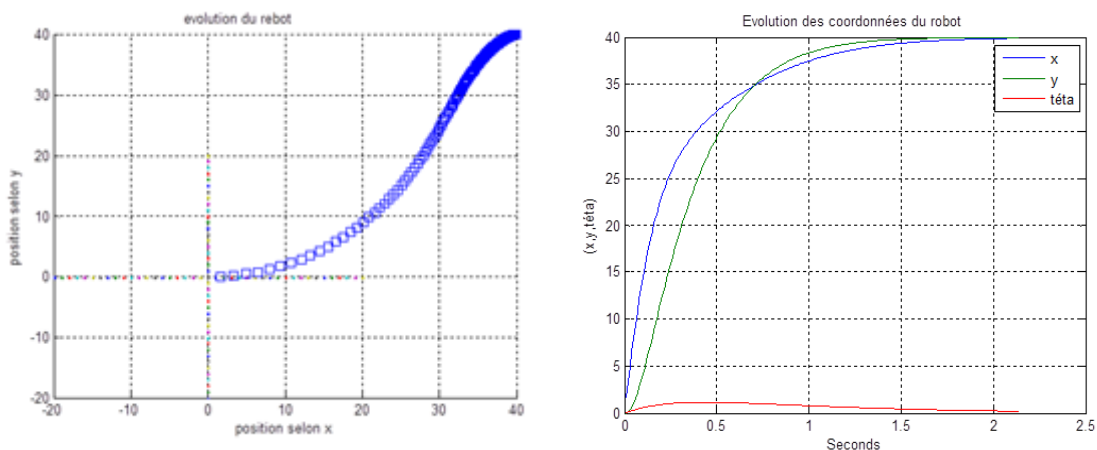


Figure.II.7. Navigation du robot en utilisant le retour d'état dans un environnement sans obstacles
exemple 1

2^{ème} cas: Point de départ $\begin{cases} x_0=20 \\ y_0=20 \end{cases}$ point d'arrivée $\begin{cases} x_c=100 \\ y_c=50 \end{cases}$

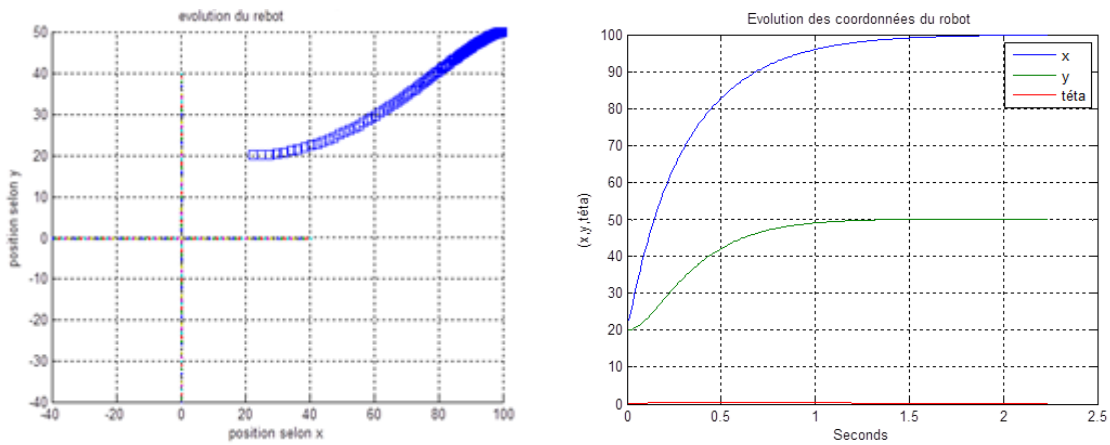


Figure.II.8. Navigation du robot en utilisant le retour d'état dans un environnement sans obstacles exemple 2

3^{ème} cas : Point de départ $\begin{cases} x_0=70 \\ y_0=60 \end{cases}$ point d'arrivée $\begin{cases} x_c=0 \\ y_c=0 \end{cases}$

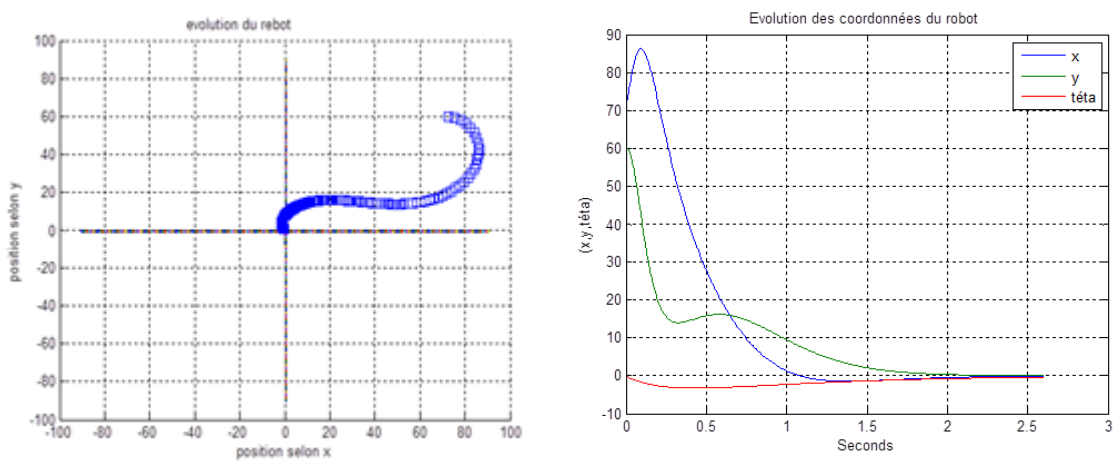


Figure.II.9. Navigation du robot en utilisant le retour d'état dans un environnement sans obstacles exemple 3

II.3. Navigation floue

La théorie des ensembles flous a fait son apparition en 1965, suite à la publication d'un article d'une quinzaine de pages intitulé « fuzzy sets » (ensembles flous) par Lotfi A.Zadeh, professeur à l'université de Berkley en Californie, considéré aujourd'hui comme le fondateur de la logique floue [17].

Dans ses travaux en automatique et théorie des systèmes, Lotfi A.Zadeh a formalisé la représentation et le traitement de connaissances imprécises ou approximatives afin de pouvoir traiter des systèmes de grande complexité dans lesquels sont présents par exemple des facteurs humains.

Dix ans plus tard, il publie un important travail « The concept of a linguistic variable and its application to approximate reasoning » qui va marquer le point de départ de nombreuses études et expérimentations .

En 1975 E.H. Mamdani expérimente un régulateur flou qu'il perfectionne au cours des années suivantes [17]. En 1985 M. Suengo décrit des applications industrielles possibles en régulation floue. En 1995, suite aux travaux de J.S.R Jang, la logique floue est élargie aux systèmes à réseaux de neurones et à l'intelligence artificielle.

Les applications de la commande floue ont connu un essor remarquable au Japon où elles sont devenues un véritable argument de vente à travers de nombreuses applications grand public [18]: machines à laver, appareils photographiques, cameras vidéo, etc.

La toute première régulation floue a été implémentée en 1979 dans une cimenterie au Danemark [18]. Depuis, la commande floue a fait ses preuves à travers de nombreuses applications industrielles : ascenseurs, usine de papier au Portugal en 1992, pilote d'hélicoptère, métro de sandai (Japon) en service depuis 1987 qui utilise un double régulateur flou (C.S.C: Constant Speed Control et T.A.S.C : Train Automatic Speed Controlleur) [19].

II.3.1. Contrôleur flou

La logique floue a été conçue essentiellement, 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 [2].

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.10)

- **La fuzzification** : consiste à associer pour chaque valeur d'entrée, un ou plusieurs sous-ensembles flous. Cette opération permet le passage du domaine réel vers le domaine flou. Généralement, on utilise, la fuzzification singleton.
- **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 Mamdani.
- **La defuzzification** : L'ensemble flou de sortie est déterminé à la fin de l'inférence, mais il n'est pas directement utilisable. Il est alors nécessaire de passer du « monde flou » au « monde réel » et ainsi obtenir une valeur numérique précise qui sera appliquée au système. C'est la defuzzification. 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).

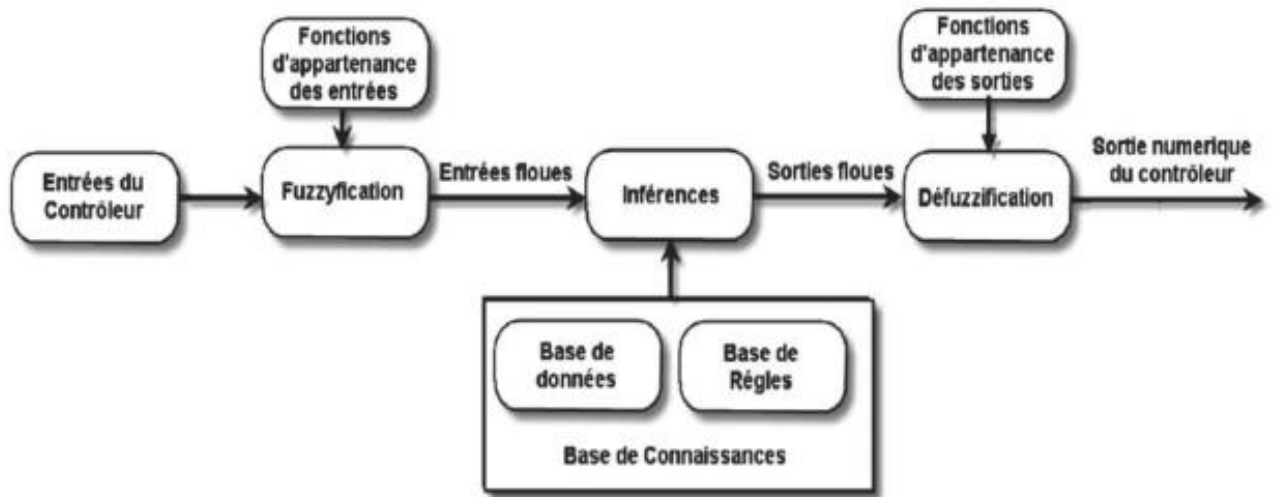


Figure.II.10. Structure d'un contrôleur flou

II.3.2. 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_c ») et la vitesse d'avancement (V_{ac}) du robot (Figure II.11) (où $q=[x, y, \theta]$ est la position du robot).

Le modèle cinématique utilisé est donné par [20] :

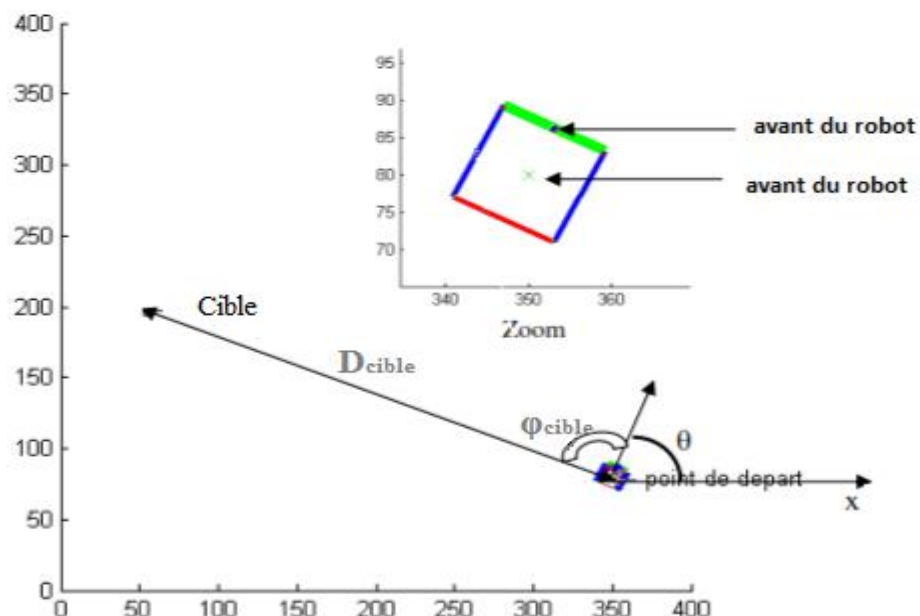


Figure.II.11. Distance et orientation initiales du robot par rapport à la cible.

Le modèle cinématique utilisé est donné par :

$$\begin{cases} \dot{x} = V_{ac} \cos(\theta) \\ \dot{y} = V_{ac} \sin(\theta) \\ \dot{\theta} = W_c \end{cases}$$

La position et orientation du robot sont obtenues par intégration de l'équation (II.1), d'où :

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

Tel que : T_e est la période d'échantillonnage.

L'erreur d'orientation du robot elle est donnée par la relation :

$$\varphi(i+1) = \varphi(i) - v_o * T_e$$

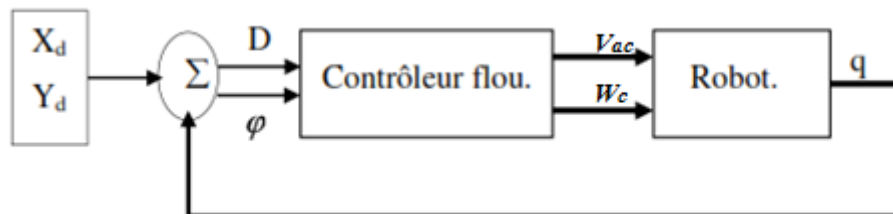


Figure.II.12. Schéma du système de commande

II.3.3. Présentation schématique du robot

Puisque nous avons fait appel au contrôleur flou, nous allons d'abord procéder 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) [20].

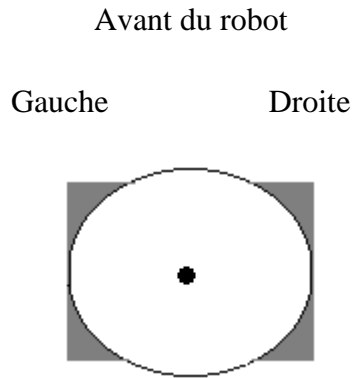


Figure.II.13. Schéma représentatif du robot mobile

II.3.4. Comportement du contrôleur flou

Dans cette section, nous allons concevoir un contrôleur flou, qui guidera les déplacements du robot en lui assurant la convergence vers la cible. Le mouvement du robot est possible, en lui introduisant, les positions, orientations courantes et les coordonnées de la cible. Ce comportement ne pourra opérer que dans un environnement sans obstacles gênant la progression du robot.

II.3.4.1. Définition des fonctions d'appartenances

Les univers de discours des variables (D_{cible} et φ_{cible}), retenus après plusieurs essais, sont respectivement décomposés en trois et cinq sous-ensembles flous selon la figure II.14.

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)

-pour la variable angle d'orientation φ : **CD** (complètement droite), **MD** (moyennement droit), **C** (centre), **MG** (moyennement gauche), **CG** (complètement gauche) [2].

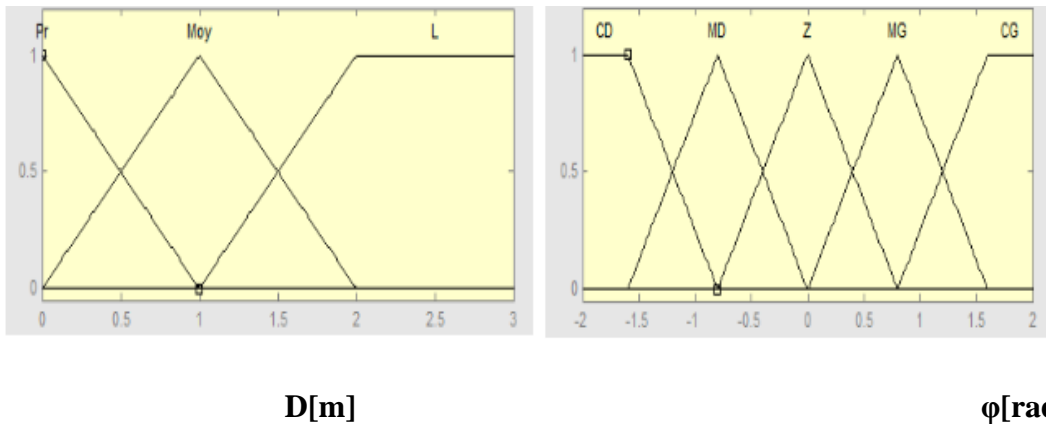


Figure.II.14. Les fonctions d'appartenance des variables d'entrée du contrôleur

« Convergence vers la cible »

Les univers de discours des variables de sortie du contrôleur (vitesse d'avancement V_{ac} et vitesse de braquage W_c), sont partitionnées comme suit :

-la première variable est partitionnée en trois sous-ensembles flous :

Z (Zéro) , **Rap** (rapide) et **TRap** (très rapide)

-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).

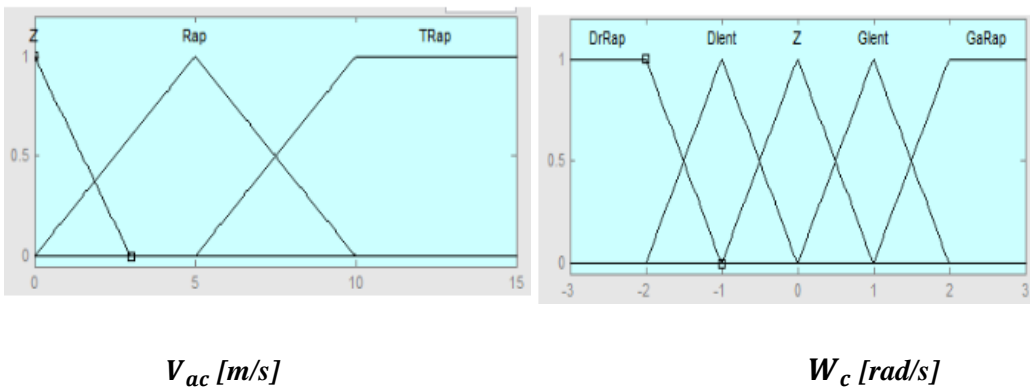


Figure.II.15. Les fonctions d'appartenances des variables de sortie du contrôleur

II.3.4.2 Base de règles

Le contrôleur flou utilisé est de type Mamdani. 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 V_{ac} et W_c . Les règles floues sont des expressions formées des prémisses (conditions) et de conclusions (actions) de la forme:

Si D est L *et* φ est C *alors* W_c est $DrRap$ *et* V_{ac} est lente.

Les deux tableaux (II.1 et II.2) représentent les bases de règles floues pour les deux variables de sortie V_{ac} Tableau (II.1) et W_c tableau (II.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 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 [2].

Tableau.II.1. Base de règles floues de la vitesse d'avancement « V_{ac} ».

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

Tableau.II.2. Base de règles floues de la vitesse de braquage « W_c ».

φ_{cible} \ D_{cible}	Pro	Moy	L
CD	DrRap	DrRap	DrRap
MD	DrRap	DrLent	DrLent
C	Z	Z	Z
MG	GaRap	GaLent	GaLent
CG	GaRap	GaRap	GaRap

II.3.5. Résultats de simulation

Les tests de simulation ont montré que le robot s’oriente vers la cible avec une vitesse de braquage modérée. Quand le robot se rapproche de la cible, il décélère jusqu’à l’arrêt total. Quelques exemples sont illustrés par les figures.

1^{er} cas: Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d’arrivée $\begin{cases} x_c=40 \\ y_c=40 \end{cases}$

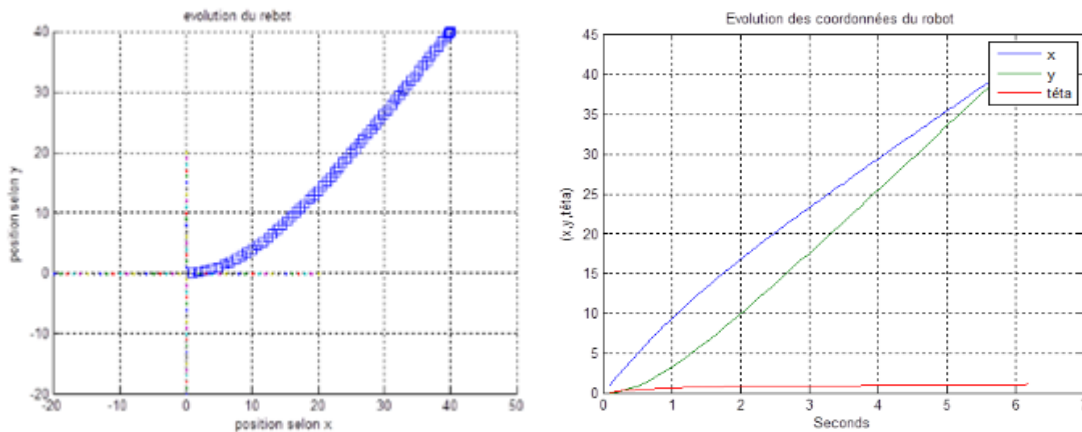


Figure.III.16. Navigation du robot en utilisant la logique floue Dans un environnement sans obstacles exemple(1)

2^{ème} cas: Point de départ $\begin{cases} x_0=20 \\ y_0=20 \end{cases}$ point d’arrivée $\begin{cases} x_c=100 \\ y_c=50 \end{cases}$

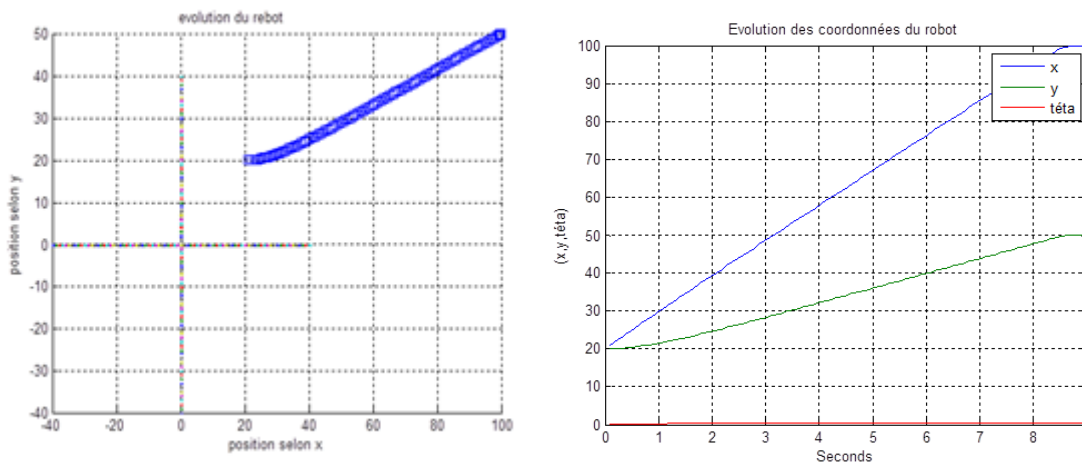


Figure.III.17. Navigation du robot en utilisant la logique floue Dans un environnement sans obstacles: exemple(2)

II.4. Conclusion

Dans ce chapitre, nous avons exploré deux approches de navigation des robots mobiles. La première concerne le concept d'une commande par retour d'état. La deuxième, utilise les principes de la logique floue.

La première partie, concernant l'approche de la commande par retour d'état, a permis au robot de réaliser la tâche « *convergence vers la cible* », en se déplaçant d'une situation initiale vers une situation finale.

Dans la deuxième partie on a constaté, la simplicité de l'utilisation de la logique floue, et sa grande capacité de détermination des actions. Nous avons développé un contrôleur flou permettant au robot d'atteindre la cible dans un environnement sans obstacles. Les résultats obtenus par cette approche ont montré l'efficacité de la conception du contrôleur.

Chapitre III

Navigation dans un environnement avec obstacles

III.1. Introduction

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. Dans ce chapitre nous, nous présentons une étude de deux algorithmes d'évitement d'obstacles d'un robot mobile. La première méthode de navigation est la navigation en utilisant des contrôleurs flous. La deuxième est celle des champs de potentiel. La tâche confiée au robot est l'atteinte de cibles sans aucune intervention externe, et cela, dans un environnement sur lequel peu d'informations a priori lui sont fournies. Le robot est amené à exploiter les informations issues des capteurs ultrasons pour se guider de sorte à éviter les obstacles qui peuvent surgir sur son chemin.

III.2. Le contrôleur Flou d'évitement d'obstacles

La problématique traitée dans le chapitre précédant est la convergence du robot vers la cible. Les résultats obtenus dans un environnement sans obstacles ont montré les performances du contrôleur flou « convergence vers la cible ». Ce dernier, utilisé seul, garantit l'atteinte de cible sans intervention externe, mais sans pouvoir garantir la non collision du robot avec les obstacles gênant la progression du robot. Donc, le robot doit être munit d'un deuxième contrôleur lui permettant d'éviter la collision avec ces obstacles. Ce dernier exploitera les données (positions des obstacles par rapport au robot) issues des capteurs ultrasons, installés sur l'avant du robot, pour guider le déplacement.

III.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 un obstacle se trouve dans un périmètre de rayon « D » (simulant la portée du capteur), on calcule l'état du capteur, et cela, en fonction de la distance robot-obstacle et son angle d'incidence (angle entre la direction du robot et celle de l'obstacle).

III.2.2. Conception du navigateur réactif flou

La détection de ces obstacles est assurée par un module de perception. Ce dernier détermine la position de chaque obstacle se trouvant dans le rayon D et transmet les données (distance et angle d'orientation) de l'obstacle le plus proche au *contrôleur 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 contrôleur, utilisé seul, garantit la non collision de robot avec les obstacles rencontrés sur sa trajectoire, mais sans pouvoir autant le guider vers la destination. Un *contrôleur convergence vers la cible*, exploitant les données issues des capteurs et la position du robot par rapport à la cible, est alors nécessaire. Ce qui implique la fusion des deux comportements.

Le contrôleur d'évitement d'obstacles est sollicité à **chaque fois que le capteur 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 afin d'éviter l'obstacle. Par exemple, si le capteur C détecte un obstacle proche sur le côté droit du robot, le robot tourne à gauche et dans le cas, où C détecte un obstacle proche sur le côté gauche du robot, le robot tourne à droite. La figure.III.1 donne le schéma de principe du système de contrôle. Les sorties du contrôleur sont la vitesse de braquage (W_{ob}) et la vitesse d'avancement V_{aob} .

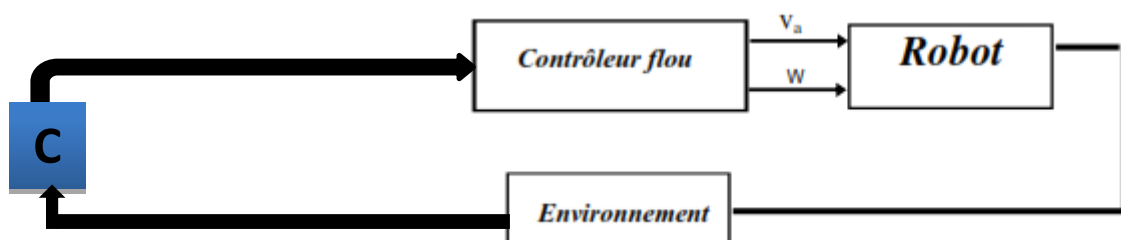


Figure.III.1. Schéma du système de contrôle.

III.2.3. Définition des fonctions d'appartenance

Les deux variables d'entrée du contrôleur sont :

D_{ob} : Distance robot-obstacle

φ_{ob} : Angle entre l'orientation du robot et la direction robot-obstacle.

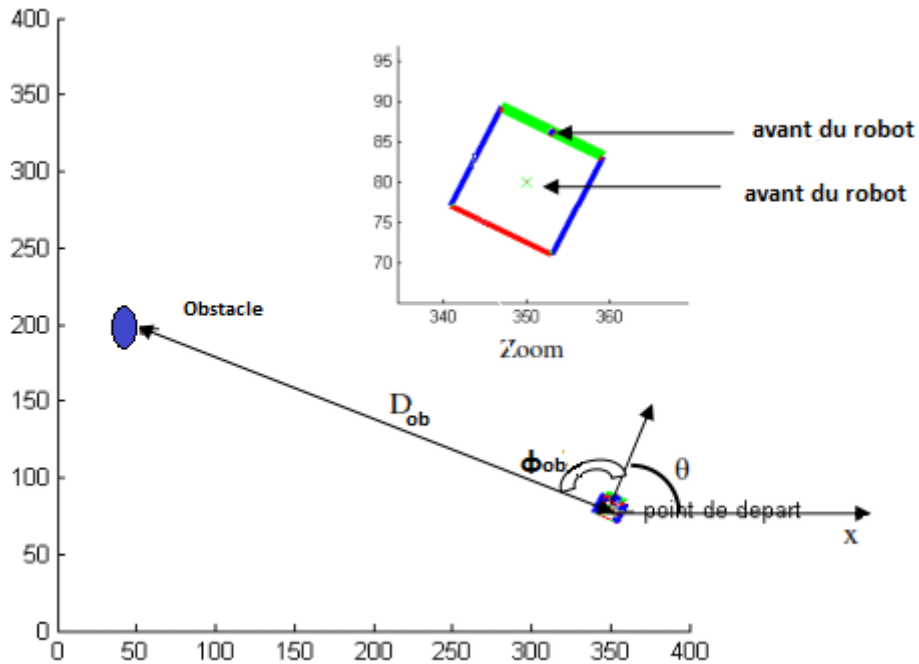


Figure III.2. Distance et orientation du robot par rapport à l'obstacle

Les univers de discours des variables (D_{ob} et φ_{ob}), retenus après plusieurs essais, sont respectivement décomposés en trois et cinq sous-ensembles flous selon la figure III.3. 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)

-pour la variable angle d'orientation : **CD** (complètement droite), **MD** (moyennement droit), **C** (centre), **MG** (moyennement gauche), **CG** (complètement gauche) [2].

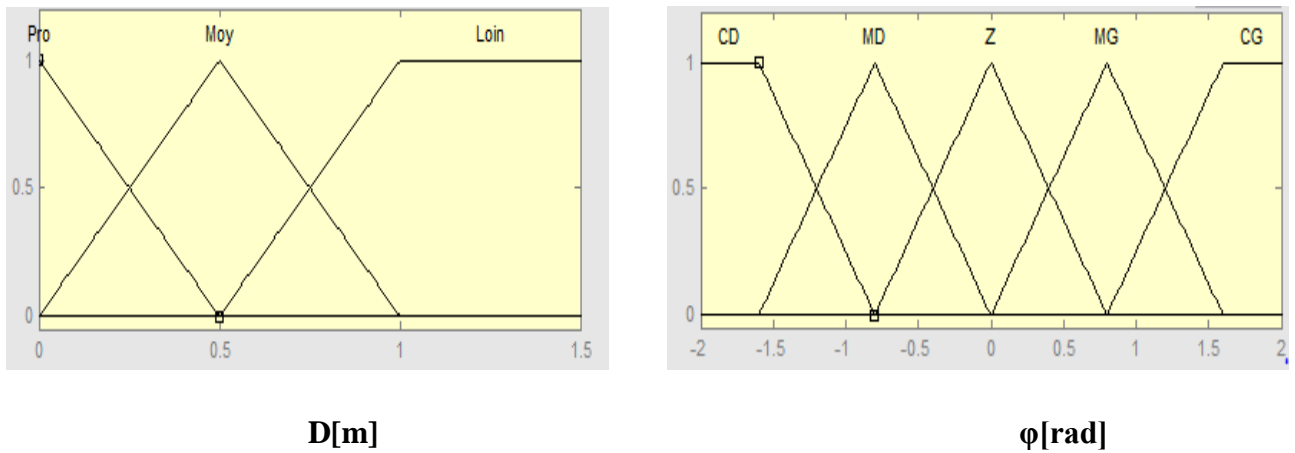


Figure.III.3 Les fonctions d'appartenance des variables d'entrée du contrôleur d'évitement d'obstacles.

Les univers de discours des variables de sortie du contrôleur (vitesse d'avancement V_{aob} et vitesse de braquage W_{ob}), sont partitionnées comme suit :

-la première variable est partitionnée en quatre sous-ensembles flous :

Z (Zéro), **Lent**, **Rap** (rapide) et **TRap** (très rapide)

-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) [2].

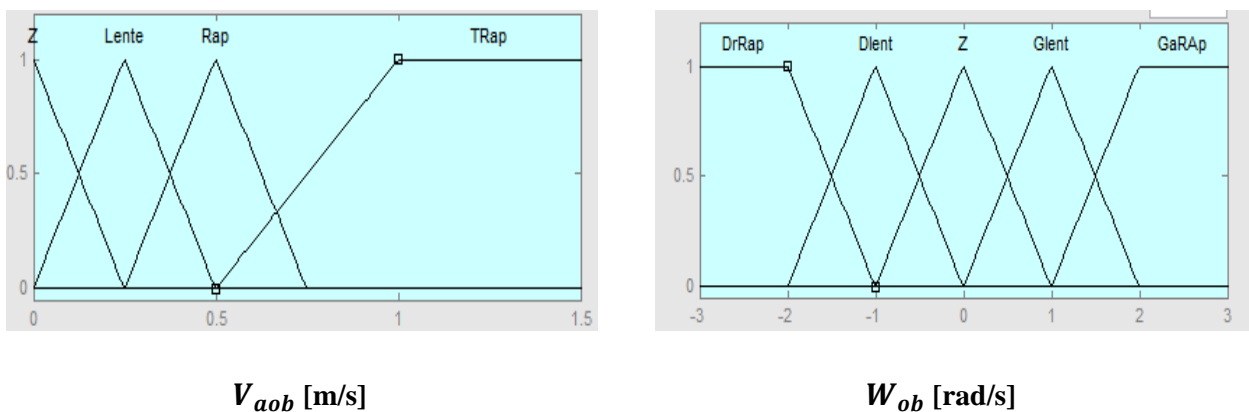


Figure.III.4. Les fonctions d'appartenance des variables de sortie du contrôleur d'évitement d'obstacles.

III.2.4. Base de règles

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

Si D_{obs} est L *et* φ_{obs} est C *alors* W_{ob} est DrRap *et* V_{aob} est lente.

Les deux tableaux (III.1 et III.2) représentent les bases de règles floues pour les deux variables de sortie V_{aob} Tableau (III.1) et W_{ob} tableau (III.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.

Tableau.III.1. Base de règles floues de la vitesse d'avancement « V_{aob} ».

φ_{ob} \ D_{ob}	Pro	Moy	L
CD	Z	Rap	TRap
MD	Z	Lente	Rap
C	Z	Z	Lente
MG	Z	Lente	Rap
CG	Z	Rap	TRap

Tableau.III.2. Base de règles floues de la vitesse de braquage « W_{ob} ».

φ_{ob} \ D_{obs}	Pro	Moy	L
CD	Z	Z	Z
MD	GaRap	GaLent	GaLent
C	DrRap	DrRap	DrRap
MG	DrRap	DrLent	DrLent
CG	Z	Z	Z

III.2.5 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 [2].

En fait, la sortie, V_{aob} , du contrôleur d'évitement d'obstacles est comprise entre 0 et 1. Elle est directement multipliée par la vitesse d'avancement générée par le contrôleur de convergence vers la cible et joue le rôle de frein en s'approchant de l'obstacle. Par contre, la sortie W est directement additionnée à la vitesse de braquage générée par ce dernier.

III.2.6 Résultats de simulation

Les tests de simulation ont montré que le robot évolue dans l'environnement, contenant des obstacles, sans collision. A chaque fois qu'un obstacle est détecté par le capteur, le contrôleur d'évitement d'obstacles réagit en générant une consigne qui permet au robot d'éviter ce dernier. Quelques exemples sont illustrés par les figures.

1^{er} cas: trajectoire du robot entre quatre obstacles,

$$\text{Point de départ } \begin{cases} x_0=0 \\ y_0=0 \end{cases} \quad \text{point d'arrivée } \begin{cases} x_c=20 \\ y_c=19 \end{cases}$$

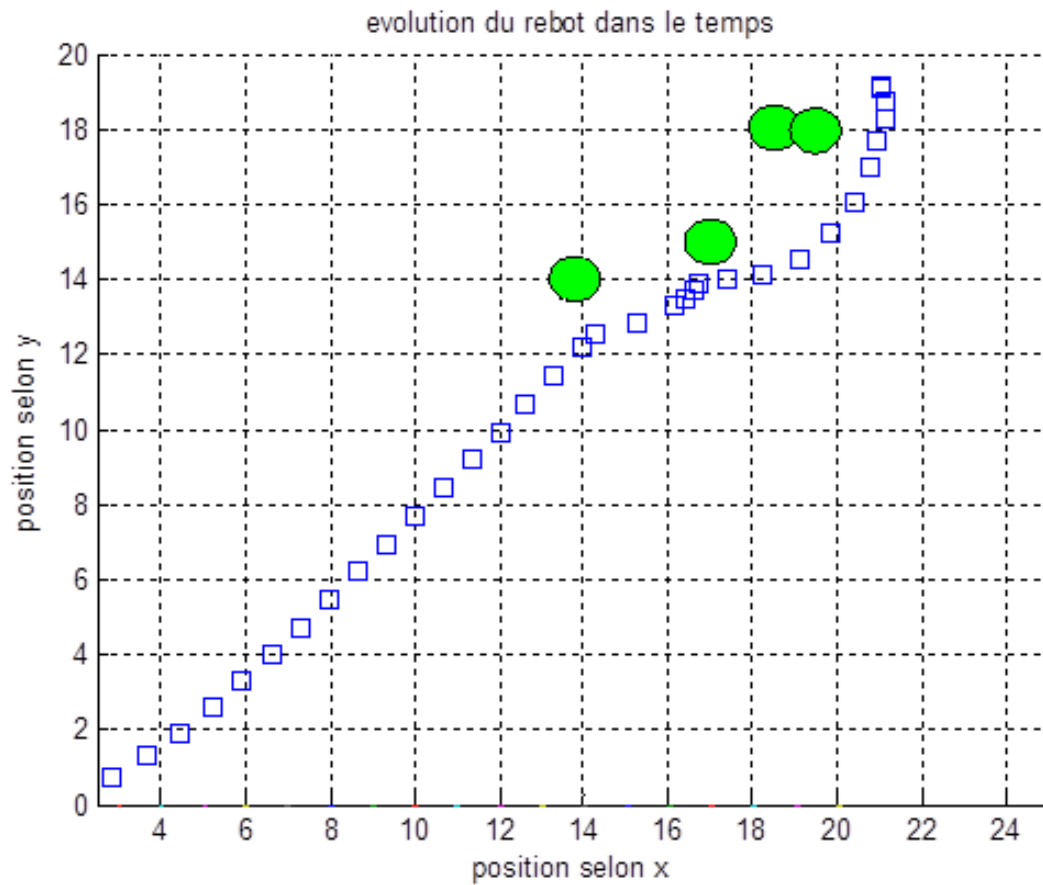


Figure.III.6. navigation du robot dans un environnement avec obstacles, exemple(1)

2^{ème} cas: trajectoire du robot entre dix obstacles,

Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d'arrivée $\begin{cases} x_c=26.5 \\ y_c=26 \end{cases}$

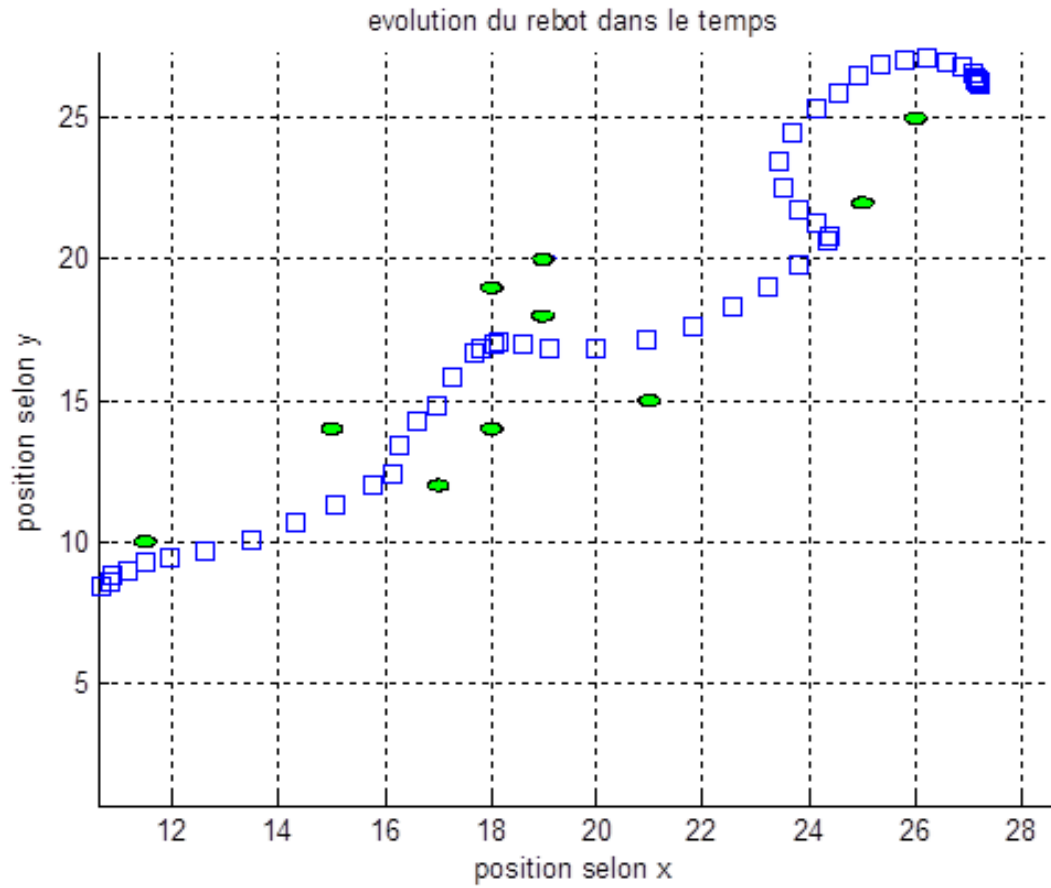


Figure.III.7. Navigation du robot dans un environnement avec obstacles, exemple(2)

3^{ème} cas: évitement du premier obstacle par sa gauche le deuxième par sa droite

$$\text{Point de départ } \begin{cases} x_0=0 \\ y_0=0 \end{cases} \text{ point d'arrivée } \begin{cases} x_c=75 \\ y_c=75 \end{cases}$$

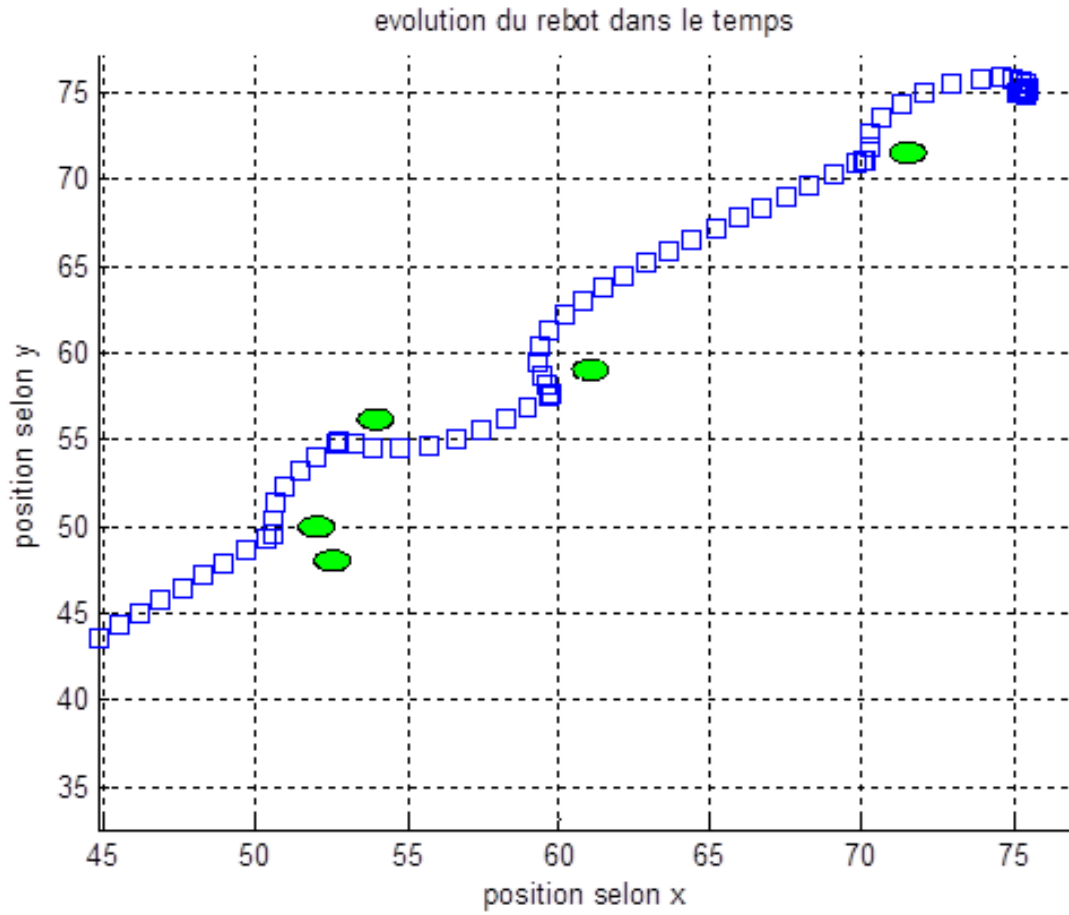


Figure.III.8. Navigation du robot dans un environnement avec obstacle, exemple(3)

4^{ème} cas: Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d'arrivée $\begin{cases} x_c=40 \\ y_c=40 \end{cases}$

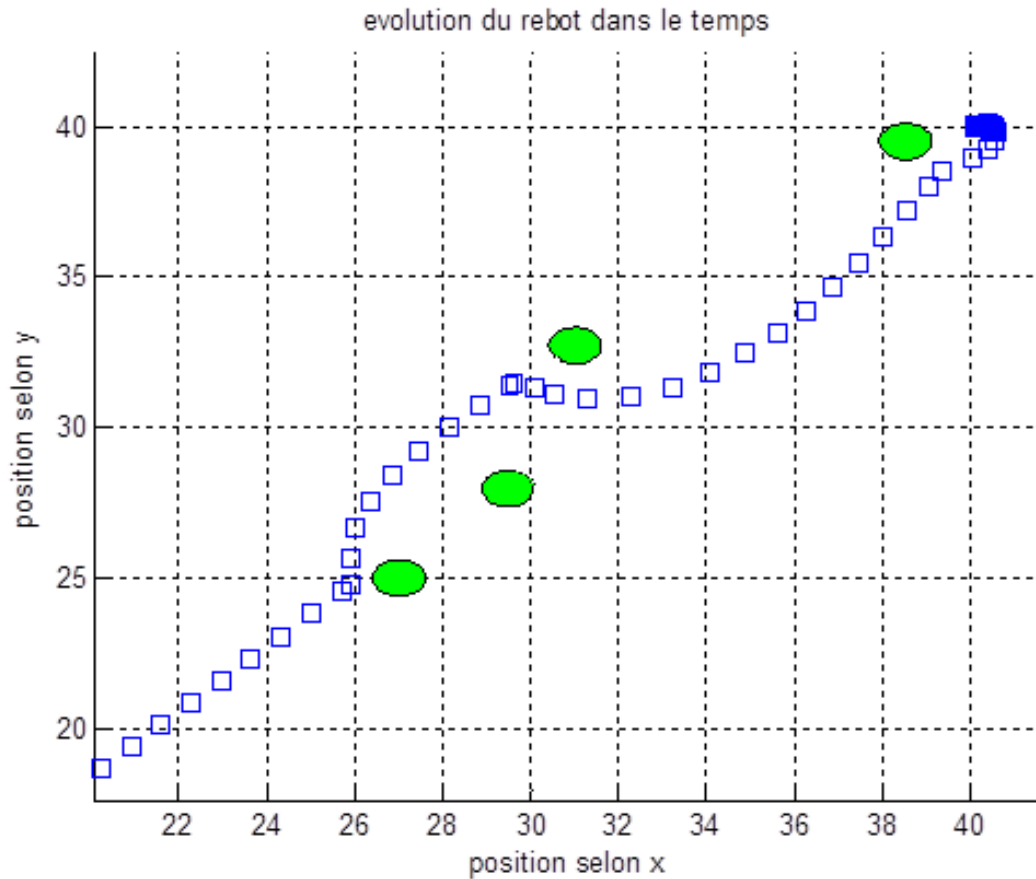


Figure.III.9. Navigation du robot dans un environnement avec obstacles, exemple(4)

III.3. Navigation avec la méthode du potentiel

III.3.1. Formulation de la Méthode

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 $U_{rep}(q)$, liés aux obstacles :

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

La force résultante est égale au gradient. 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 et le gradient :

$$\nabla U = \left(\frac{\partial U}{\partial x}, \frac{\partial U}{\partial y} \right)^T \quad (02)$$

La force résultante est constituée de deux parties :

$$F(q) = F_{att}(q) + F_{Rep}(q) \quad (03)$$

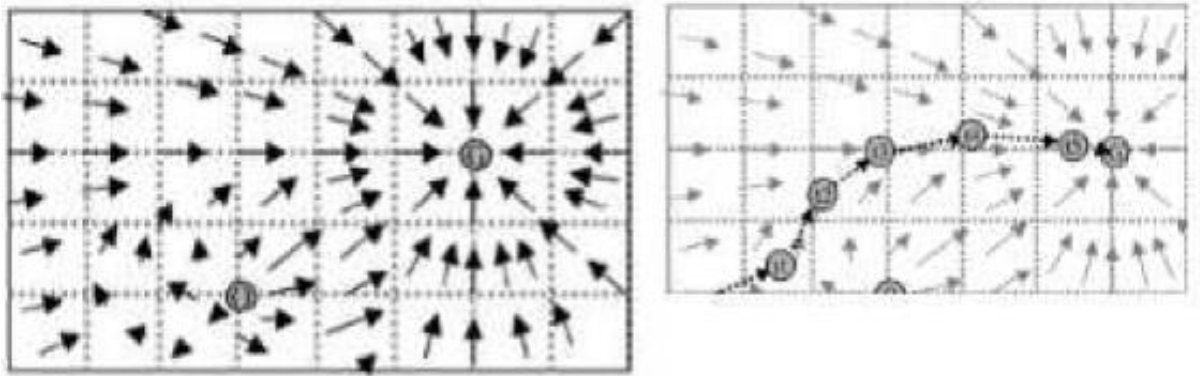


Figure.III.10. Carte des potentiels attractifs et répulsifs.

Le potentiel attractif prend généralement la forme :

$$U_{att} = \frac{1}{2} \cdot K_{att} \cdot d_c(q)^2 \quad (04)$$

Avec $d_c(q) = (q - q_f)$: distance robot – point cible q : configuration courante

q_f : configuration finale désirée K_{att} : gain de pondération

La force attractive

$$F_{att} = - \nabla U_{att}(q) = -K_{att} \cdot (q - q_f) \quad (05)$$

Pour chaque obstacle Obs de l'environnement, le potentiel répulsif est de la forme :

$$U_{rep} = \begin{cases} \frac{1}{2} K_{rep} \cdot \left(\frac{1}{d(q)} - \frac{1}{d_0} \right)^2 & \text{si } d(q) < d_0 \\ 0 & \text{si } d(q) > d_0 \end{cases} \quad (06)$$

Où $d(q)$ représente la distance du robot à l'obstacle, d_0 la distance d'influence des obstacles. La force répulsive résultante :

$$F_{rep}(q) = -\nabla U_{rep}(q) \begin{cases} K_{rep} \left(\frac{1}{d(q)} - \frac{1}{d_0} \right) \frac{1}{d^2(q)} \frac{q - q_{obs}}{d(q)} & \text{si } d(q) \leq d_0 \\ 0 & \text{si } d(q) > d_0 \end{cases} \quad (07)$$

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.[15]

III.3.2. Résultats de simulation

Les tests de simulation ont montré que le robot évolue dans l'environnement contenant des obstacles. A chaque fois qu'un obstacle est détecté par le capteur, le contrôleur d'évitement d'obstacles réagit en générant une consigne qui permet au robot d'éviter ce dernier, sauf dans un cas où le robot a été piégé par un minimum local (Le potentiel qui influence le robot est nul), et cela l'a conduit à une situation de blocage, comme l'illustre la figure III.14.

1^{er} cas: trajectoire du robot entre quatre obstacles,

Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d'arrivée $\begin{cases} x_c=20 \\ y_c=19 \end{cases}$

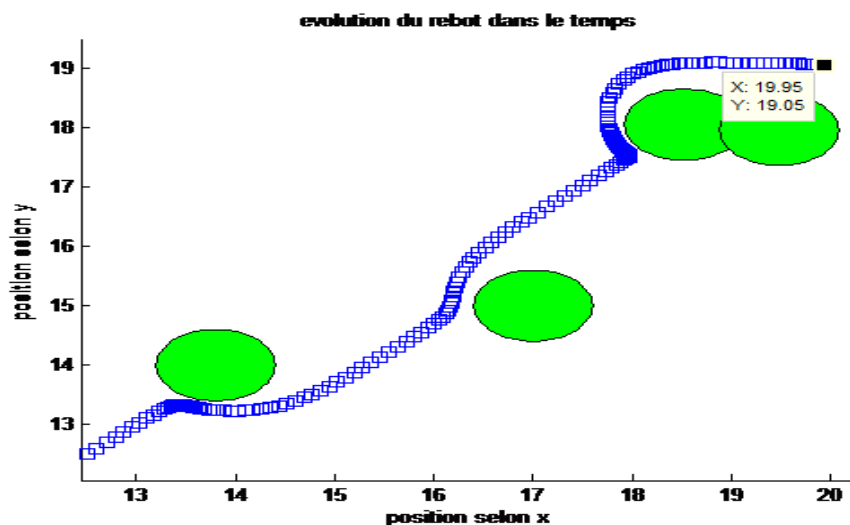


Figure.III.11. Navigation du robot dans un environnement avec obstacles, exemple(a)

2^{ème} cas: trajectoire du robot entre dix obstacles puis treize obstacles,

Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d'arrivée $\begin{cases} x_c=26.5 \\ y_c=26 \end{cases}$

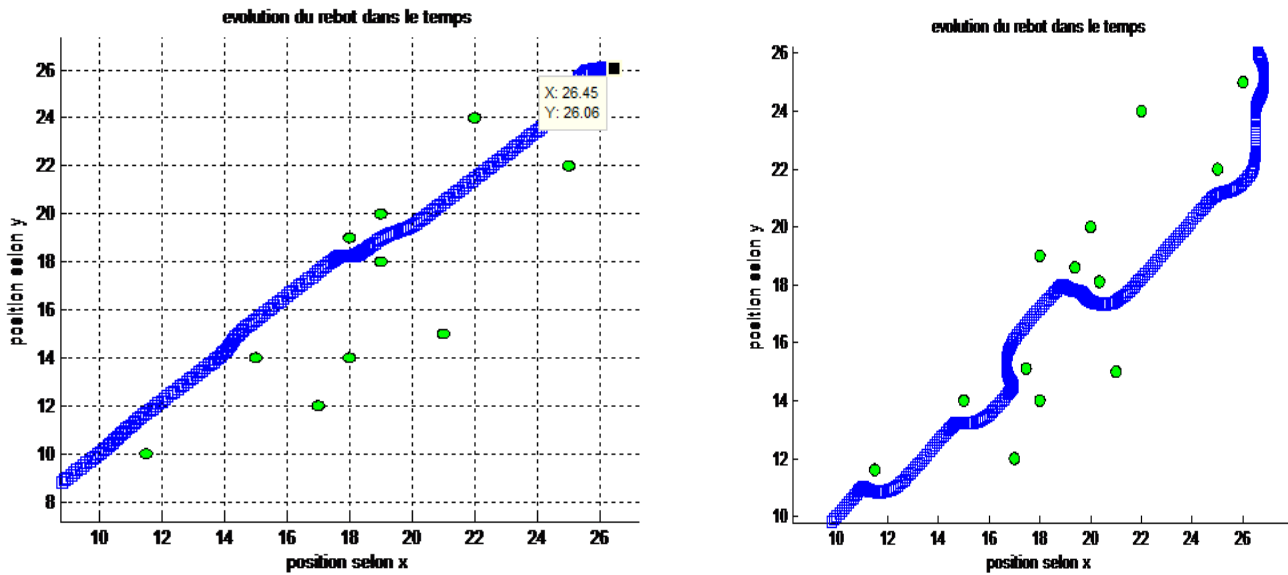


Figure.III.12. Navigation du robot dans un environnement avec obstacles, exemple(b)

3^{ème} cas : Point de départ $\begin{cases} x_0=0 \\ y_0=0 \end{cases}$ point d'arrivée $\begin{cases} x_c=38 \\ y_c=38 \end{cases}$

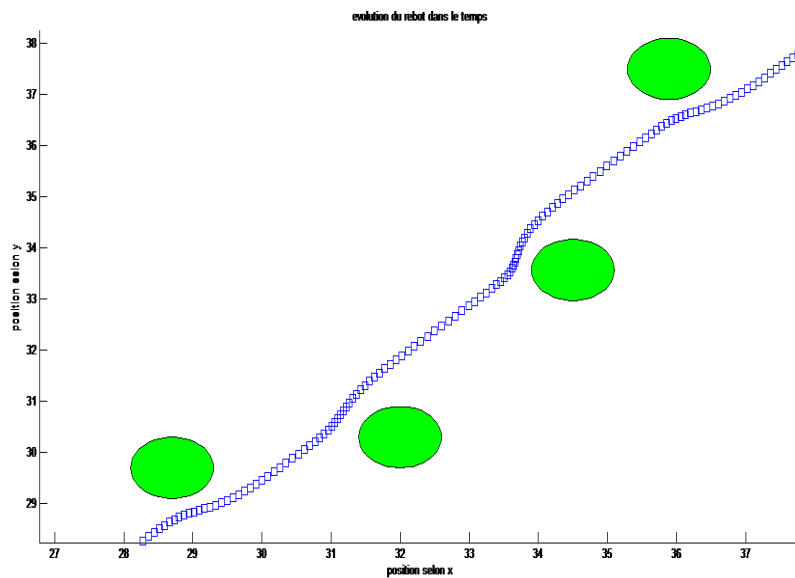


Figure.III.13. Navigation du robot dans un environnement avec obstacles, exemple(c)

4^{ème} cas : Illustration du problème des minima locaux

Point de départ $\begin{cases} x_0=40 \\ y_0=40 \end{cases}$ point d'arrivée $\begin{cases} x_c=75 \\ y_c=75 \end{cases}$

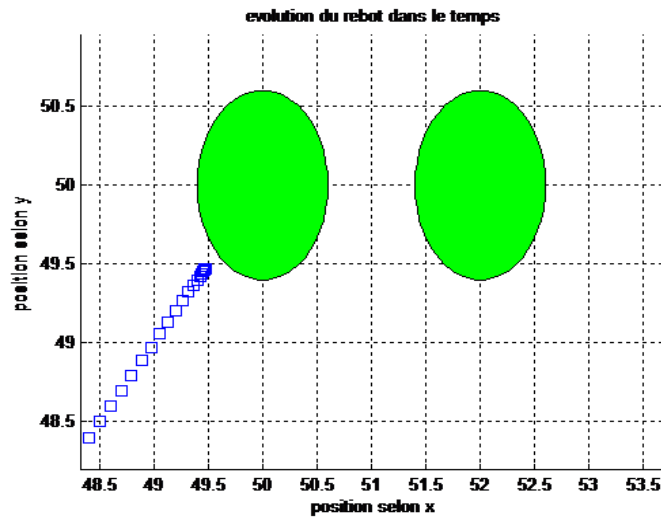


Figure.III.14. Illustration du problème du Minimum local.

Commentaire :

En observant **la figure III.14**, on voit bien que le robot s'arrête devant le 1^{er} obstacle et ne progresse pas.

La grande simplicité de la méthode du potentiel ne doit cependant pas masquer son défaut majeur.

III.4. Conclusion

Dans ce chapitre, deux approches de navigation des robots mobiles ont été étudiées par simulation. La première utilise les principes de la logique floue. La deuxième concerne la méthode des champs de potentiel.

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 par 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 statiques.

La deuxième approche, concernant la méthode des champs de potentiel, est simple et élégante. Mais, elle se heurte à de nombreux inconvénients. D'une part, cette approche est sujette à des minima locaux. Par conséquent, la convergence vers la cible n'est pas tout le temps assurée. D'autres part, ces potentiels peuvent donner lieu à des fortes oscillations des mouvements du robot en présence des obstacles et principalement lorsque celui-ci navigue dans des passages étroits (couloirs, portes). Enfin, le vecteur de déplacement désigné par le champ du potentiel ne prend, en aucun cas, en compte la cinématique ou la dynamique du système robotique considéré. Un robot disposant de contraintes non-holonomes aura de sérieuses difficultés à suivre une telle direction.

Conclusion générale

Le travail présenté dans ce mémoire concerne une étude de deux méthodes de navigation des robots mobiles, à savoir : la méthode de champs de potentiel et la logique floue.

En premier lieu, nous avons présenté des généralités sur la robotique mobile en rappelant quelques définitions et notions de bases, les différents types de locomotion, l'architecture de système de commande ainsi que les notions de perception, localisation, planification du chemin. Nous avons aussi repris l'étude de quelques méthodes de navigation. En second lieu, nous avons présenté une modélisation cinématique d'un robot mobile unicycle (différentiel), tout en expliquant la notion du roulement sans glissement ainsi que la définition du centre instantané de rotation.

Nous avons abordé notre travail en étudiant quelques méthodes de convergence vers la cible dans un environnement sans obstacles, à savoir : la méthode de retour d'état et la logique floue.

Le troisième chapitre expose le fond de notre travail concernant l'étude par simulation et la comparaison de deux méthodes d'évitement d'obstacles : la méthode de champs de potentiel et la logique floue. Les résultats obtenus par l'implémentation du contrôleur flou sont satisfaisants. Car, le robot peut se déplacer d'une situation initiale vers une situation finale, en évitant tous les obstacles. La méthode des champs de potentiel est simple et élégante. Mais, elle se heurte à quelques inconvénients. D'une part, cette approche est sujette à des minima locaux. D'autre part, le vecteur de déplacement désigné par le champ de potentiel ne prend pas en compte la cinématique ou la dynamique du système robotique considéré. Un robot disposant de contraintes non-holonomes aura de sérieuses difficultés à suivre une telle direction.

Ce travail nous a permis de consolider et de mieux comprendre des connaissances acquises durant notre formation. Nous espérons bien que cette étude soit étendue aux autres techniques de navigation et poursuivie par une implémentation réelle.

Références bibliographiques

- [1] Youcef MECHALIKH et Ali MILOUDI, *Développement d'algorithmes d'évitement d'obstacles statiques et dynamiques*. PFE pour l'obtention du diplôme MASTER en automatique, Université Kasdi Merbah-Ouargla, 2012.
- [2] SOUALMI Boussaad, *Planification de la trajectoire dans un environnement partiellement connu à base des techniques d'intelligence artificielle*. en vue de l'obtention du diplôme magister, Université de Bejaia, 2009.
- [3] David Fillat , *Cours de Robotique Mobile*, Ecole National Supérieur de Techniques Avancées de Paris, 2006.
- [4] Robert Holmberg, Oussama Khatib, *Development and Control of a Holonomic Mobile Robot for Mobile Manipulation Tasks*, International Journal of Robotics. Research, vol. 19, N°11, november 2000.
- [5] Stephane Petti and Thierry Fraichard. *Partial motion planning framework for reactive planning within dynamic environments*. In *IEEE Int. Conf. On Robotics and Automation, Barcelona, Spain, 2005*.
- [6] Marie-José Aldon, *Capteurs et méthodes pour la localisation des robots mobiles*. Techniques de l'Ingénieur, traité Informatique industrielle, 1998.
- [7] G. Frappier, *Système inertiels de navigation pour robots mobiles*, Séminaire sur Les robots mobiles, EC2, Paris, 1990.
- [8] N. J. Nilsson. A mobile automaton : *An application of artificial intelligence techniques*. In the 1st International conference on artificial Intelligence, 1969.
- [9] B. Chazelle, *Voronoi diagrams : A survey of a fundamental geometric data structure*. Algorithmic and Geometric Aspects of Robotics, 1987.
- [10] A. Lingas. *The power of non-rectilinear holes*. In *In Proceedings, 9th International Colloquium on Automata*, 1982.

- [11] P. Bessiere, J. Ahuactzin, T. El-Ghazali, and E. Mazer. *The ariane's clewalgorithm : Global planning with local methods*. In *IEEE-RSJ Int. Conference on Intelligent Robots and Systems*, 1993.
- [12] Khatib, O. *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*. The International Journal of Robotics Research, Vol N° 5, 1986.
- [13] Y. Koren and J. Borenstein. *Potential field methods and their inherent limitations for mobile robot navigation*. In *IEEE International Conference on Robotics and Automation*, April 1991.
- [14] S.S. Ge and Y.J. Cui. *Dynamic motion planning for mobil robots using potential field method*. *Autonomous Robots*, page : 207–222, 2002.
- [15] Boubkeur MENDIL, Cours de Robotique, Université de Targa Ouzemmour Bejaia, 2014.
- [16] Hachemi BOUICHE, *Etude & implémentation d'une méthode de localisation basée SLAM*. Projet final pour l'obtention du grade MASTER en AUTOMATIQUE, Bejaia 2012.
- [17] C. Rosenthal. *Histoire de la logique floue*. Une approche sociologique des pratiques de démonstration, *Revue de synthèse*, 1998.
- [18] F. Chevrier, F. Guély. *La logique floue*. Cahier technique Schneider N°191, mars 1998.
- [19] H. Buhler. *Réglage par la logique floue*. Presse Polytechnique et Universitaire Romandes, 1994.
- [20] Roland Siegwart and Illah R. Nourbakhsh : *"Introduction to Autonomous mobile robot"*, sfMIT Press Cambridge, Massachusetts 2004.

Table des matières

Avant-propos	i
Liste des tableaux	v
Introduction générale	1
Chapitre I. Introduction à la robotique mobile	
I.1. Introduction.....	2
I.2. Concepts de base	2
I.2.1. Définition d'un robot mobile.....	2
I.2.2. Exemples d'application.....	2
I.2.3 Notions d'holonomie/non-holonomie.....	3
I.3. Systèmes de locomotion.....	4
I.3.1 Les mobiles à roues.....	4
I.3.2 les mobiles à chenilles.....	5
I.3.3 Les mobiles à pattes.....	5
I.4. Architecture du système de commande.....	6
I.4.1. Niveau interface homme-machine.....	7
I.4.2 Planificateur (Générateur) de tâches.....	7
I.4.3 Le contrôleur d'exécution.....	8
I.4.4 Le niveau fonctionnel.....	8
I.5. Navigation.....	9
I.6. La localisation.....	9
I.6.1 La localisation à l'estime (relative).....	10
I.6.2 La localisation absolue.....	10
I.7. La perception.....	11
I.7.1 capteurs proprioceptifs.....	12
I.7.1.1 Les odomètres.....	12
I.7.1.2 Les accéléromètres.....	12
I.7.1.3 Radars Doppler.....	13
I.7.1.4 Le gyroscope.....	13

1.7.2 Capteurs extéroceptifs.....	13
1.7.2.1 Les télémètres.....	14
1.7.3 Notion d'incertitude.....	14
I.8. Méthodes de planification du chemin.....	14
I.8.1 Méthodes délibératifs.....	14
I.8.1.1 Graphe de visibilité.....	14
I.8.1.2 Décomposition cellulaire.....	15
I.8.2 Méthodes replanification.....	16
I.8.2.1 Le fil d'Ariane.....	16
I.8.3 Approche réactives.....	17
I.8.3.1 Approche par champs de potentiel.....	17
I.8.3.2 Déformation de la trajectoire.....	18
I.9. Conclusion.....	19

Chapitre II. Navigation dans un environnement sans obstacles

II.1. Introduction.....	20
II.2. Méthode du retour d'état.....	20
II.2.1. Modélisation cinématique.....	20
II.2.1.1. Posture du robot.....	21
II.2.1.2. Roulement sans glissement.....	22
II.2.1.3 Robot mobile de type unicycle (différentiel).....	24
II.2.2. Commande par retour d'état.....	26
II.2.3. Résultats de la simulation.....	28
II.3. Navigation floue.....	30
II.3.1. Contrôleur flou.....	31
II.3.2. Commande du robot par la logique floue.....	32
II.3.3. Présentation schématique du robot.....	33
II.2.4. Comportement du contrôleur flou.....	34
II.3.4.1. Définition des fonctions d'appartenances.....	34
II.3.4.2 Base de règles.....	35
II.3.5. Résultats de simulation.....	37
II.4. Conclusion.....	38

Chapitre III. Navigation dans un environnement avec obstacles

III.1. Introduction.....	39
III.2. Le contrôleur Flou d'évitement d'obstacles.....	39
III.2.1. Perception et détection des obstacles.....	39
III.2.2. Conception du navigateur réactif flou.....	40
III.2.3. Définition des fonctions d'appartenance.....	40
III.2.4. Base de règles.....	43
III.2.5 Fusion de comportements.....	44
III.2.6 Résultats de simulation.....	44
III.3. Navigation avec la méthode du potentiel.....	48
III.3.1. Formulation de la Méthode.....	48
III.3.2. Résultats de simulation.....	50
III.4. Conclusion.....	52
Conclusion générale	54
Référence bibliographique	vi