



Université Mohamed Khider de Biskra
Faculté des Sciences et de la Technologie
Département de génie électrique

MÉMOIRE DE MASTER

Sciences et Technologies
Électronique
Électronique des systèmes embarqués

Réf. :

Présenté et soutenu par :
Krid abderrahmane

Le : dimanche 24 juin 2018

Commande d'un robot mobile par logique flou

Jury :

Mr	Benakcha Abdelhamid	Pr	Université de Biskra	Président
Mr	Benelmir Okba	M.C.B	Université de Biskra	Encadreur
Mme	Ouarhlent Saloua	M.A.A	Université de Biskra	Examineur

Année universitaire : 2018

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement Supérieur et de la Recherche scientifique



Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Electronique
Option : Electronique des systèmes embarqués

Mémoire de Fin d'Etudes
En vue de l'obtention du diplôme:

MASTER

Thème

Commande d'un robot mobile par logique floue

Présenté par :

Krid Abderrahmane

Avis favorable de l'encadreur :

Benelmir Okba

signature

Avis favorable du Président du Jury

Benakcha Abdelhamid

Signature

Cachet et signature

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement Supérieur et de la Recherche scientifique



Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Electronique
Option : Electronique des systèmes embarqués

Thème :

Commande d'un robot mobile par logique floue

Proposé par : Benelmir Okba

Dirigé par : Benelmir Okba

Résumé

Le travail proposé consiste à développer des algorithmes qui aideront le robot mobile à surmonter les obstacles et à poursuivre son chemin de manière optimale du point de départ à la destination finale. Nous proposons donc l'utilisation d'un contrôleur flou, qui est une application de la logique floue pour contrôler des systèmes dynamiques pour lesquels nous n'avons pas de modèles satisfaisants. Pour aider le robot à suivre les meilleurs chemins et d'atteindre indépendamment un objectif clairement défini, en évitant les obstacles en temps réel, sans avoir besoin d'un modèle environnemental complet, pour ce la on a fait la comparaison entre deux méthodes de la logique floue.

ملخص

الهدف من هذا العمل هو دراسة وتطوير خوارزميات من شأنها أن تساعد روبوت متحرك مستقل في بيئة غير متغيرة بتجاوز العقبات ومواصلة مساره في الطريق الأمثل من نقطة الإنطلاق إلى الوجهة النهائية، لذلك نقترح استخدام متحكم غامض و الذي هو تطبيق من المنطق الضبابي للسيطرة على الأنظمة الديناميكية، لضمان السلوك الصحيح ومساعدة الروبوت لمتابعة أفضل مسارات للوصول بشكل مستقل إلى هدف واضح المعالم، وتجنب العقبات في الوقت الحقيقي، و قمنا بالمقارنة بين طريقتين من المنطق الضبابي للتحكم و اثبتنا ايهما الافضل في هذا العمل.

Dédicace

Je dédie ce travail à :

Mes très chers Parents, que Dieu me les garde.

Mes frères et ma sœur.

Tous mes amis.

Tous les membres de ma Famille.

Tous mes enseignants.

Ceux qui m'ont aidé.

Ceux qui estiment le savoir.

Remerciements

Avant tout, Je remercie Dieu le Tout Puissant de m'avoir donné le courage, la patience et la force durant toutes ces années d'étude.

Je tiens à remercier vivement mon encadrant Monsieur Benelmir Okba, docteur à l'université de Biskra, pour m'avoir dirigé et soutenu au cours de la réalisation de ce travail.

J'ai été profondément honoré que Monsieur Benakcha Abdelhamid, professeur à l'université de Biskra ait accepté de présider ce jury, et Je mesure tout l'honneur qu'à bien voulu me faire Madame Ouarhlent Saloua, docteur à l'université de Biskra, en acceptant d'examiner ce mémoire.

J'associe à ces remerciements tous les enseignants du département de génie électrique ainsi que toutes les personnes ayant contribué par leur soutien.

Je profite de cette occasion qui m'est offerte pour remercier tous mes amis et collègues qui m'ont beaucoup aidé durant cette période.

Enfin, une reconnaissance et des remerciements particuliers à toute ma famille pour son soutien permanent le long de mes études.

Table des matières

Table des matières	I
Liste des figures	IV
Liste des tableaux	VI
Symboles et Abréviations	VII
Introduction Générale	1

Chapitre I: Les robots mobiles

I.1. Introduction	4
I.2. Définition de robot mobile	4
I.3. Autonomie d'un robot mobile.....	5
I.4. Classification des robots mobiles	6
I.5. Architecture des robots mobiles	7
I.5.1. Module locomotion	7
I.5.1.1. Les mobiles à roues	7
I.5.1.2. Les mobiles à chenilles	9
I.5.1.3. Les mobiles marcheurs	9
I.5.1.4. Les robots rampants.....	10
I.5.2. Module perception	10
I.5.2.1. Capteurs intéroceptifs	10
I.5.2.2. Capteurs extéroceptifs	10
I.5.3. Module décision.....	11
I.6. Navigation autonome des robots mobiles	11
I.6.1. Planification de mouvement	11
I.6.1.1. Planification globale de trajectoire	11
I.6.1.2. Planification locale de trajectoire	12
I.6.2. Localisation	12
I.6.3. Suivi de trajectoire	12
I.6.4. Évitement réactif d'obstacles.....	12
I.7. Architectures de contrôle des robots mobiles	13
I.7.1. Architecture délibérative (Hiérarchique).....	13
I.7.2. Architecture comportementale (réactive)	13
I.7.3. Architecture hybride.....	15
I.8. Conclusion.....	16

Chapitre II: Modélisation

II.1. Introduction.....	18
II.2. Modélisation cinématique.....	18
II.2.1. Mise en place des repères.....	18
II.2.2. Roulement sans glissement.....	19
II.2.3. Disposition des roues et centre instantané de rotation.....	20
II.2.4. Modélisation cinématique du robot mobile de type uni-cycle.....	21
II.2.5. Modèle cinématique en posture.....	24
II.3. conclusion.....	24

Chapitre III: La logique floue

II.1. Introduction.....	26
III.2. Logique floue et ensemble flou.....	26
III.2.1. Concept de base d'un ensemble floue.....	26
III.2.1.1. Caractéristiques d'un ensemble flou.....	28
III.2.1.2. Fonctions d'appartenance.....	28
III.2.1.3. Variables linguistiques.....	30
III.2.2. Opérations sur les ensembles flous.....	31
III.3. Systèmes d'inférence floue (SIF).....	32
III.4. Représentation mathématique des systèmes flous.....	33
III.4.1. Fuzzification.....	33
III.4.2. Inférence.....	34
III.4.2.1. Base de règles floues.....	34
III.4.3.2. Moteur d'inférence flou.....	34
III.4.3.2.1. Méthode de Mamdani.....	35
III.4.3.2.2. Méthode de Takagi-Sugeno.....	36
III.4.4. Défuzzification.....	37
III.5. Conclusion.....	38

Chapitre IV: Résultat de simulation

IV.1. Introduction.....	40
IV.2. Modèle du robot mobile utilisé dans notre travail.....	40
IV.3. Méthode de Mamdani.....	42
IV.3.1. Comportement de navigation robot-cible.....	42
IV.3.1.1. Structures de base de notre contrôleur flou robot-cible.....	42
IV.3.1.1.1. Fuzzification.....	43
IV.3.1.1.2. Inférence.....	45

IV.3.1.1.3. Défuzzification	45
IV.3.1.2. Organigramme de simulation	46
IV.3.1.3. Résultats de simulation	47
IV.3.2. Comportements de navigation avec évitement d'obstacle	49
IV.3.2.1. Structures de contrôleur flou de navigation avec évitement d'obstacle	49
IV.3.2.1.1. Fuzzification	50
IV.3.2.1.2. Inférence	51
IV.3.2.2. Organigramme de simulation	52
IV.3.2.3. Résultats de simulation	53
IV.4. Méthode de Sugeno.....	56
IV.4.1. Structures de base de notre contrôleur flou.....	56
IV.4.1.1. Fuzzification.....	56
IV.4.1.2. Inférence	57
IV.4.2. Résultats de simulation.....	58
IV.5. Comparaison entre les deux méthodes	61
IV.6. Conclusion.....	63
Conclusion Générale	64
Bibliographie	66

Liste des figures

Figure I.1	le robot spirit.....	4
Figure I.2	(a) Interaction entre le robot et l'environnement, (b) Etapes de traitement automatique	6
Figure I.3	Structure d'un robot mobile.....	7
Figure I.4	Robot mobile a roués	8
Figure I.5	Robot mobile de type uni-cycle.....	8
Figure I.6	Robot mobile différentielle.....	8
Figure I.7	Robot mobile de type tricycle.....	8
Figure I.8	Robot mobile de type voiture	8
Figure I.9	Robot omnidirectionnel.....	9
Figure I.10	Robot mobile à traction synchrone	9
Figure I.11	Robot mobile à chenille.....	9
Figure I.12	Robot mobile marcheur.....	9
Figure I.13	Architecture délibérative (hiérarchique)	13
Figure I.14	Exemple d'architecture comportementale.....	14
Figure I.15	Évitement d'obstacle.....	15
Figure I.16	Suivi de chemin	15
Figure I.17	Architecture hybride	16
Figure II.1	Repérage d'un robot mobile.....	19
Figure II.2	Caractérisation du roulement sans glissement	19
Figure II.3	Les principaux types de roues	21
Figure II.4	Centre instantané de rotation.....	21
Figure II.5	Robot mobile de type uni-cycle.....	22
Figure II.6	Le Robot mobile Pioneer P3-DX.....	22
Figure II.7	Centre instantané de rotation d'un robot de type uni-cycle.....	22
Figure III.1	Exemples des fonctions d'appartenance (a) Logique classique (b) Logique floue	27
Figure III.2	Représentation d'un sous-ensemble floue et principales caractéristiques	28
Figure III.3	Types de fonctions d'appartenances: (a) fonction – Γ (demi trappez droit);(b)fonction – S (sigmoïde); (c) fonction – L (demi trappez gauche); (d) fonction – Λ (triangle); (e) fonction Gaussienne; (f) fonction - Π (trapézoïdale).....	29
Figure III.4	L'ensemble flou de variable "taille"	31
Figure III.5	Les fonctions d'appartenance associées aux opérations :	32
Figure III.6	Structure d'un SIF	33
Figure IV.1	Le robot mobile utilisé	40
Figure IV.2	Structure du contrôleur de navigation robot-cible (Mamdani)	43
Figure IV.3	Fonctions d'appartenance de Drc (Mamdani).....	43
Figure IV.4	Fonctions d'appartenance de θ_{rc} (Mamdani).....	44
Figure IV.5	Fonctions d'appartenance de Vd (Mamdani).....	44
Figure IV.6	Fonctions d'appartenance de Vg (Mamdani).....	44
Figure IV.7	L'organigramme de navigation robot-cible	46
Figure IV.8	Navigation robot-cible R1 (Mamdani)	47
Figure IV.9	Variation de Vd et Vg de R1 de navigation robot-cible (Mamdani).....	47
Figure IV.10	Navigation robot-cible R2 (Mamdani)	48

Figure IV.11 Variation de Vd et Vg de R2 de navigation robot-cible (Mamdani)	48
Figure IV.12 Structure du contrôleur de navigation avec évitement d'obstacle (Mamdani)	49
Figure IV.13 Fonctions d'appartenance de Dro (Mamdani)	50
Figure IV.14 Fonctions d'appartenance de θ_0 (Mamdani)	50
Figure IV.15 Fonctions d'appartenance de Vd cas d'obstacle (Mamdani)	50
Figure IV.16 Fonctions d'appartenance de Vd cas d'obstacle (Mamdani)	51
Figure IV.17 L'organigramme de navigation avec évitement d'obstacle	52
Figure IV.18 Navigation avec évitement d'obstacle R1 (Mamdani)	53
Figure IV.19 Variation de Vd et Vg de R1 de navigation avec évitement d'obstacle (Mamdani)	53
Figure IV.20 Navigation avec évitement d'obstacle R1 (Mamdani)	54
Figure IV.21 Variation de Vd et Vg de R2 de navigation avec évitement d'obstacle (Mamdani)	54
Figure IV.22 Navigation avec évitement d'obstacle R3 (Mamdani)	55
Figure IV.23 Variation de Vd et Vg de R3 de navigation avec évitement d'obstacle (Mamdani)	55
Figure IV.24 La structure de notre contrôleur flou (FLC) (Sugeno)	56
Figure IV.25 Navigation robot-cible R1 (Sugeno)	58
Figure IV.26 Variation de Vd et Vg de R1 de navigation robot-cible(Sugeno)	59
Figure IV.27 Navigation avec évitement d'obstacle R2 (Sugeno)	59
Figure IV.28 Variation de Vd et Vg de R2 de navigation avec évitement d'obstacle (Sugeno)	60
Figure IV.29 Navigation avec évitement d'obstacle R3 (Sugeno)	60
Figure IV.30 Variation de Vd et Vg de R3 de navigation avec évitement d'obstacle (Sugeno)	61
Figure IV.31 Comparaison entre la trajectoire de méthode Sugeno et Mamdani	62
Figure IV.32 Comparaison entre la vitesse Vd de méthode Sugeno et Mamdani	62
Figure IV.33 Comparaison entre la vitesse Vg de méthode Sugeno et Mamdani	63

Liste des tableaux

Tableau II.1	Les deux principales méthodes de défuzzification	37
Tableau IV.1	Les règles d'Inférence du contrôleur navigation robot-cible (Mamdani).....	45
Tableau IV.2	Les règles d'Inférence du contrôleur de navigation avec évitement d'obstacle (Mamdani).....	51
Tableau IV.3	Les valeurs de Vd de contrôleur de navigation robot-cible (Sugeno).....	56
Tableau IV.4	Les valeurs de Vg de contrôleur de navigation robot-cible (Sugeno).....	57
Tableau IV.5	Les valeurs de Vd de contrôleur de navigation avec évitement d'obstacle (Sugeno)....	57
Tableau IV.6	Les valeurs de Vg de contrôleur de navigation avec évitement d'obstacle (Sugeno)....	57
Tableau IV.7	Les règles d'Inférence du contrôleur de navigation robot-cible (Sugeno)	57
Tableau IV.8	Les règles d'Inférence du contrôleur de navigation avec évitement d'obstacle (Sugeno)	58

Symboles et Abréviations

q	Vecteur des coordonnées généralisées du robot
θ	Angle entre le plan de la roue
O'	Le centre de l'axe des roues motrices
CIR	Le centre Instantané de Rotation
ρ	Le rayon de courbure de la trajectoire du robot
ω	La vitesse de rotation du robot autour du CIR
v_d	La vitesse de roue droite
v_g	La vitesse de roue gauche
L	La distance entre chaque roue et O'
r	Le rayon de la roue
$\dot{\phi}_d$	La vitesse de rotation roue droite
$\dot{\phi}_g$	La vitesse de rotation roue gauche
v	Vitesse linéaire du robot
$\dot{\theta}$	Vitesse de rotation du robot
u	la commande de robot
A	Sous-ensemble floue
U	univers de discours
$\mu_A(x)$	fonction d'appartenance
SIF	Système d'Inférence Floue
TS	Takagi-Sugeno
TS0	le modèle Takagi-Sugeno d'ordre zéro
(x_r, y_r)	La position de robot
(x_c, y_c)	Les coordonnées de point cible
θ_r	L'orientation du robot
θ_c	L'orientation de la cible
θ_{rc}	L'angle entre l'orientation actuelle du robot et celle de la cible
D_{rc}	La distance entre le robot et la cible
e_x	L'erreur entre x_r de robot et x_c de cible
e_y	L'erreur entre y_r de robot et y_c de cible
D_{ro}	La distance entre le robot et l'obstacle
θ_{ro}	L'angle entre l'orientation actuelle du robot et l'obstacle

Introduction Générale

Introduction générale

La robotique mobile est une branche de l'intelligence artificielle concernée par l'étude des systèmes automatiques capables d'une interaction directe avec le monde physique. Elle est l'ensemble des techniques permettant la conception et la réalisation de machines automatiques ou des robots mobiles. C'est un domaine de recherche essentiellement pluridisciplinaire. Sa problématique porte sur la conception et l'étude de fonction de perception, de décision et d'action, et sur l'intégration cohérente de ces fonctions dans une machine physique ou robot mobile [1]. Le robot mobile doit être capable d'adapter l'exécution d'un ensemble d'actions suivant les variations dans son environnement. Il a donc besoin de percevoir l'environnement et de raisonner sur les données perceptuelles acquises.

Il y a plusieurs solutions proposées pour la navigation autonome d'un robot mobile, Cependant cette navigation est dans tous les cas assez limitée, rencontrant des difficultés qui la rendent spécifique à une situation précise. Donc ici le suivi de trajectoire pour se déplacer de position initiale vers une cible est une mission importante pour un robot mobile. Il consiste à calculer les commandes envoyées aux actionneurs permettant de réaliser le mouvement planifié, tenant compte de la localisation du robot mobile. Pour un bon contrôle du robot mobile, nous suivons quelques étapes nécessaires pour diriger le robot dans la trajectoire contenant de nombreux obstacles, Notre travail concerne la conception d'un contrôleur flou pour la poursuite de trajectoire par un robot mobile et évitement d'obstacles.

Le mémoire est subdivisé en quatre chapitres :

Dans le premier chapitre, nous allons faire un tour rapide sur les robots mobiles pour examiner en bref les différents types de robots mobiles, les différentes architectures de contrôle, la navigation, localisation, la planification...etc.

Dans le deuxième chapitre, nous présentons la modélisation cinématique et la modélisation dynamique de robots mobiles de type uni-cycle.

Dans le troisième chapitre, nous présenterons en les principes fondamentaux de la logique floue, où, nous exposerons la théorie de la logique floue nécessaire à la conception d'un contrôleur flou, et les commandes floues de type Mamdani et Sugeno.

Dans le quatrième chapitre, nous présenterons les résultats de notre travail simulation en MATLAB, où, nous exposerons les résultats de la navigation de notre robot mobile du point initial vers la cible avec évitement d'obstacle pour les deux type de commande Mamdani est Sugeno.

Chapitre I

Les robots mobiles

I.1. Introduction

Les robots mobiles sont largement utilisés dans de nombreux domaines, notamment industriels, domestiques, scientifiques, exploratoires...etc. Nous devons donc en savoir plus sur les robots mobiles.

Dans ce chapitre, nous donnerons une idée générale sur les robots mobiles, où nous verrons des composants et des différents types de robots mobiles, et prendrons une idée sur la navigation du robot mobile dans son environnement.

I.2. Définition de robot mobile

Le robot mobile est un agent physique réalisant des tâches dans son environnement, doté de capacités de perception, de décision et d'action. L'objectif est de permettre au robot d'interagir rationnellement avec son environnement automatiquement (sans intervention humaine). Cette nouvelle machine est caractérisée par sa capacité à être programmée pour réaliser des tâches très diverses. Mettant en œuvre en particulier un ensemble des capteurs et un ensemble des actionneurs. Ses capacités en matière de manipulation d'objets lui ont permis de s'intégrer dans des lignes de production industrielle, où elle se substitue à l'homme dans les tâches difficiles, répétitives ou à risque pour l'être humain [2] [3].

Exemple de robot mobile : Le robot spirit de NASA explorateur de la planète Mars

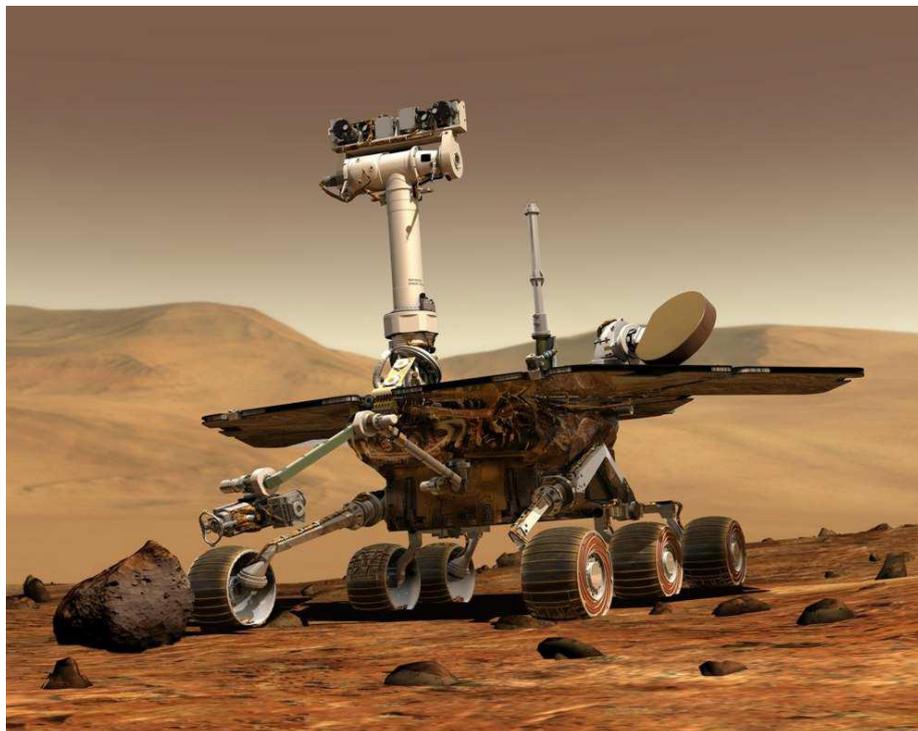


Figure I.1 Le robot spirit [4].

I.3. Autonomie d'un robot mobile

Nous considérons qu'un système est autonome si :

- Il est capable d'accomplir sans intervention humaine les objectifs pour lesquels il a été conçu.
- Il est capable de choisir ses actions afin d'accomplir ces missions.

Une machine autonome peut être définie comme étant l'association d'un système d'intelligence artificielle avec des capacités de perception, de modélisation de son environnement et de son propre état.

Elle doit être aussi dotée de capacités d'actions sur son propre état et sur son environnement. Pour cela, le robot doit suivre le schéma correspondant au paradigme (Percevoir-Décider-Agir). La figure I.2(a) présente l'interaction du robot avec son environnement. La manière dont le robot mobile gère ces différents éléments est définie par son architecture de contrôle, qui peut éventuellement faire appel à un modèle interne de l'environnement ou une stratégie intelligente pour lui permettre de planifier ses actions à long terme [2] [5].

Bien que, comme nous le verrons par la suite, plusieurs architectures existant au niveau de la satisfaction de ce paradigme, l'activité d'un tel robot se ramène aux tâches suivantes comme illustré sur la figure I.2(b):

- **Percevoir** : le robot doit acquérir des informations sur l'environnement dans lequel il évolue par l'intermédiaire de ses capteurs. Ces informations permettent de mettre à jour un modèle de l'environnement (architectures hiérarchiques ou délibératives) ou peuvent être directement utilisées comme entrées de comportement de bas niveau (architecture purement réactive).
- **Décider** : le robot doit définir des séquences d'actions résultant d'un raisonnement appliqué sur un modèle de l'environnement ou répondant de manière réflexe à des stimuli étroitement liés aux capteurs.
- **Agir** : il doit exécuter les séquences d'actions élaborées en envoyant des consignes aux actionneurs par l'intermédiaire des boucles d'asservissements [2] [5].

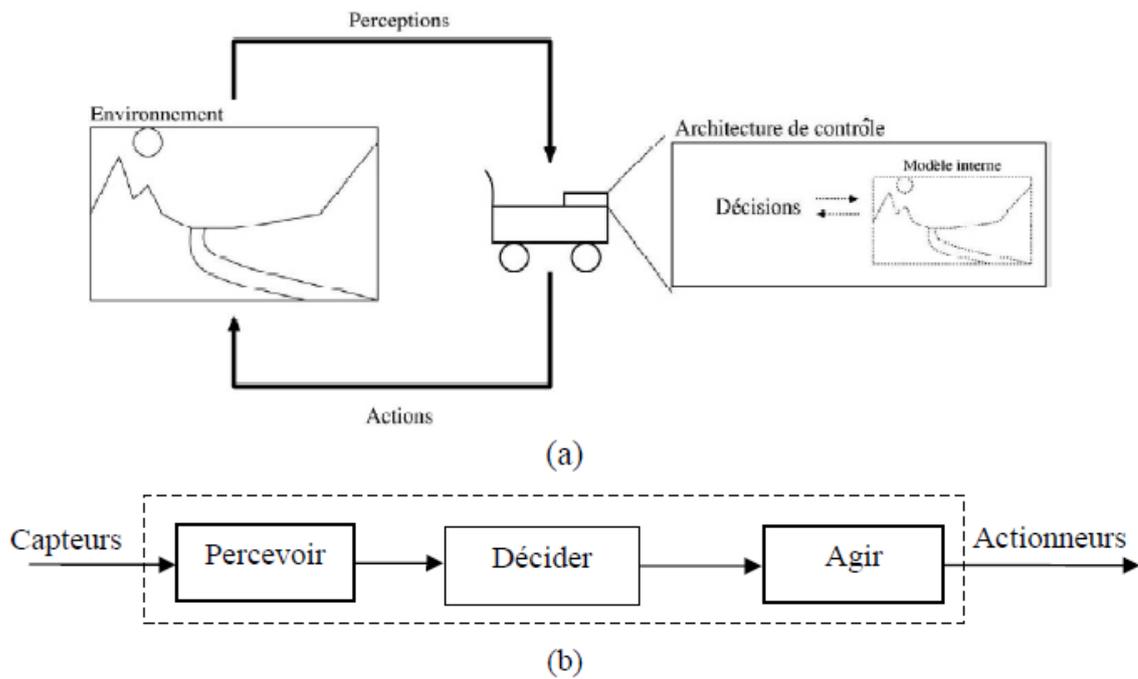


Figure I.2 (a) Interaction entre le robot et l'environnement, (b) Etapes de traitement automatique [2].

I.4. Classification des robots mobiles

Une classification est proposée dans la littérature qui définit le degré d'autonomie du robot mobile.

- Véhicule télécommandé par un opérateur qui lui impose chaque tâche élémentaire à réaliser.
- Véhicule télécommandé au sens de la tâche à réaliser. Le véhicule contrôle automatiquement ses actions.
- Véhicule semi-autonome réalisant sans l'aide de l'opérateur des tâches prédéfinies.
- Véhicule autonome qui réalise des tâches semi-définies. Ce type de véhicule pose des problèmes d'un niveau de complexité élevé de représentation des connaissances, de capacité décisionnelle et de génération de plans qui sont résolus à bord dans la mesure du possible.

L'ensemble des problèmes particuliers liés à la conception de tels robots sont :

- La conception mécanique liée à la mobilité
- La détermination de la position et de la latitude (orientation)
- La détermination du chemin optimal pour atteindre le lieu de la tâche [6] [7].

I.5. Architecture des robots mobiles

L'architecture d'un robot mobile se structure en trois modules fondamentaux :

- Module locomotion.
- Module perception.
- Module décision.

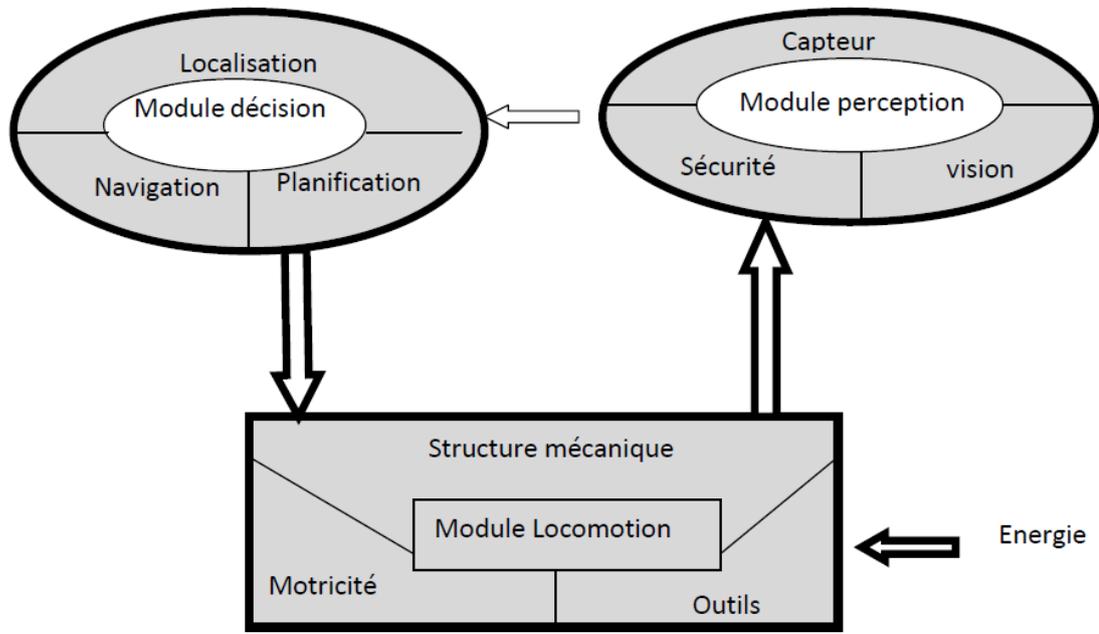


Figure I.3 Structure d'un robot mobile [1].

I.5.1. Module locomotion

Le module locomotion comporte la structure mécanique, motricité et l'énergie utilisée dans le déplacement du robot mobile. Les robots mobiles sont en effet le plus souvent désignés par leur type de locomotion [1] [6], On peut dénombrer quatre types de structures mécaniques assurant la motricité.

I.5.1.1. 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 [6].



Figure I.4 Robot mobile a roués [7].

Les modèles des robots mobiles à roues existant sont illustrés sur les figures suivantes (figure I.5 à la figure I.10).

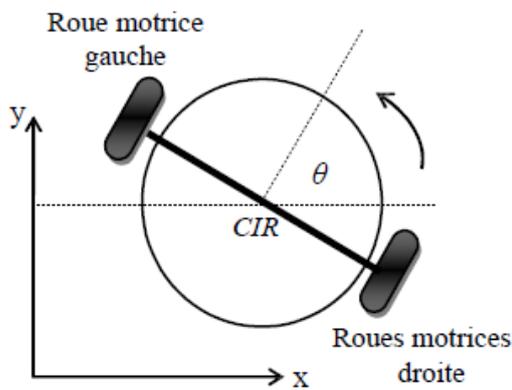


Figure I.5 Robot mobile de type uni-cycle [2].

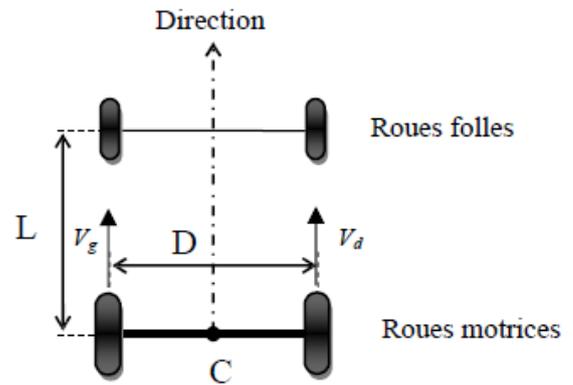


Figure I.6 Robot mobile différentielle [2].

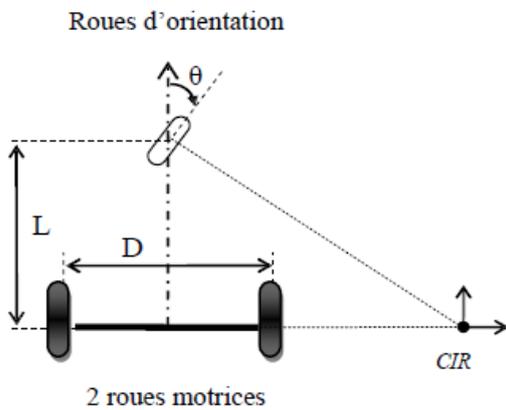


Figure I.7 Robot mobile de type tricycle [2].

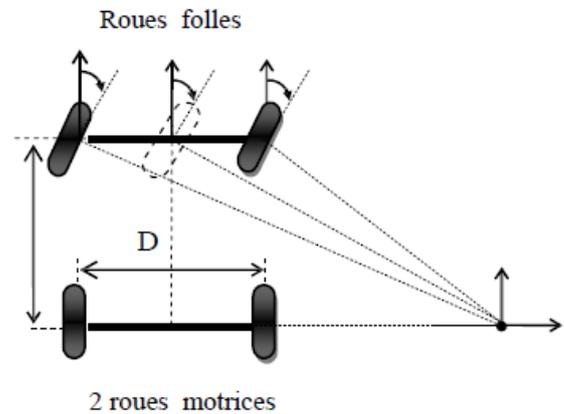


Figure I.8 Robot mobile de type voiture [2].

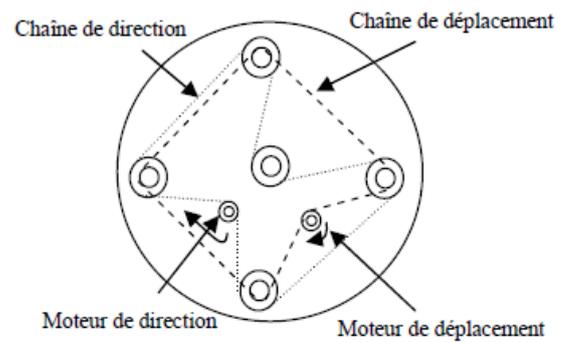
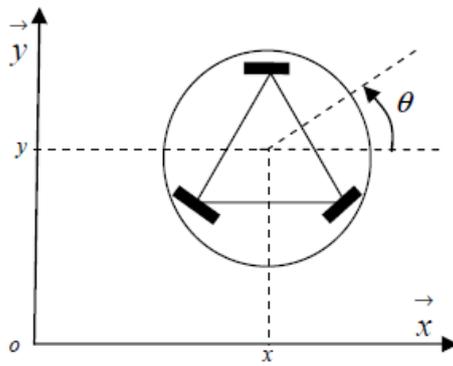


Figure I.9 Robot omnidirectionnel [2]. **Figure I.10** Robot mobile à traction synchrone [2].

I.5.1.2. 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 [6] [2].



Figure I.11 Robot mobile à chenille [7].

I.5.1.3. 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 anatomie à nombreux degrés de liberté permet un rapprochement avec les robots manipulateurs [6] [2].



Figure I.12 Robot mobile marcheur [7].

I.5.1.4. 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 [6] [7].

I.5.2. Module perception

Le système de perception est très important pour la sécurité du robot si l'environnement est encombré d'obstacles fixes ou bien mobiles (autres robot). Pour se focaliser sur le problème de navigation, nous allons deux types de capteurs utiles pour la réception de les informations proprioceptives et les informations extéroceptives qui utiles pour la tâche de navigation [2] [8].

I.5.2.1. Capteurs intéroceptifs

Fournissent des données sur l'état interne du robot (vitesse, position, orientation,...). Ces informations renseignent le robot en cas de mouvement, sur son déplacement dans l'espace (la localisation) [2] [8].

I.5.2.2. Capteurs extéroceptifs

Ont pour objectif d'acquérir des informations sur l'environnement proche du véhicule. Ils fournissent des mesures caractéristiques de la position que le robot peut acquérir dans son environnement par la détection des objets qui contourne. Ces informations peuvent être de natures très variée. Nous citons comme exemple les télémètres à ultrason, infrarouge, laser, les caméras,...etc [2] [8].

I.5.3. Module 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. 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 décision est souvent arbitraire au début, mais elle permet de développer une forme d'apprentissage qui tient compte des résultats des décisions précédentes [1].

I.6. Navigation autonome des robots mobiles

Pour exécuter une tâche de navigation autonome, un robot mobile doit mettre en œuvre un certain nombre de fonctionnalités, que nous détaillons ici.

I.6.1. Planification de mouvement

L'un des principaux objectifs du robot mobile est de pouvoir évoluer dans un environnement complexe encombré d'obstacles pour atteindre son but final. Il a besoin de construire une trajectoire définie comme une séquence de déplacement sans collision avec ces obstacles entre la position initiale (point de démarrage) et le point but ou cible.

La planification de trajectoire dans sa formulation classique est le problème du calcul de ce chemin, dans un modèle géométrique de l'environnement cela est fait en introduisant le concept d'espace des configurations qui permet de transformer le problème de la recherche d'un chemin pour un système à n degrés de liberté dans l'espace euclidien en celui du mouvement d'un point dans un espace à n dimensions où le robot est représenté par un point. Plusieurs approches sont proposées pour la planification de trajectoire. Cependant, les plus utilisées sont la planification globale et locale [2] [9].

I.6.1.1. Planification globale de trajectoire

C'est la modélisation de l'espace de l'environnement par un graphe, ou la recherche de la trajectoire est basée sur l'utilisation des algorithmes des graphes, on peut citer: Le graphe de visibilité, la décomposition cellulaire...etc [2] [9].

I.6.1.2. Planification locale de trajectoire

Généralement, l'environnement du robot mobile est inconnu, et le robot ne dispose pas, à priori, aucune information sur l'environnement. Il est nécessaire donc de réaliser une planification locale de type réflexe. Pendant le déplacement, le robot mobile doit analyser son environnement et prendre la décision en fonction de cette analyse. Les méthodes réactives de l'intelligence artificielle sont considérées comme des approches de planification locale [2] [9].

I.6.2. Localisation

Afin d'exécuter le mouvement planifié, le robot doit se localiser dans l'environnement. La localisation est l'estimation de la position du robot par rapport à un repère fixe de l'environnement, Cette estimation de la position peut s'effectuer soit par une mesure des déplacements du robot soit par une mesure de sa position absolue dans l'environnement. Plusieurs approches de localisation sont utilisées en robotique mobile. Elles peuvent être classées en trois catégories: localisation basée sur une carte, localisation par rapport à des balises et localisation par rapport à d'autres robots. Du fait des incertitudes sur les mesures utiles pour la localisation, le problème de la localisation est généralement modélisé dans un cadre probabiliste [7] [10].

I.6.3. Suivi de trajectoire

Le suivi de trajectoire consiste à calculer les commandes des actionneurs du système permettant de réaliser le mouvement planifié. Un robot étant considéré comme un système dynamique, on utilise des méthodes de commande par retour d'état pour asservir le système sur une trajectoire de référence [11].

I.6.4. Évitement réactif d'obstacles

Le suivi de la trajectoire planifiée ne permet pas de garantir l'absence de collision. En effet, des collisions peuvent se produire lors de l'exécution de la trajectoire, dues à :

- Une localisation imparfaite.
- Un plan imprécis.
- Des obstacles qui n'étaient pas dans le modèle de l'environnement utilisé pour la planification de trajectoire.

Tous ces éléments font que le mouvement initialement planifié doit être adapté lors de son exécution, et que des stratégies d'évitement réactif d'obstacles doivent être mises en œuvre.

On adoptera des stratégies différentes en fonction du type de système, de sa vitesse, et du champ d'application [11].

I.7. Architectures de contrôle des robots mobiles

Un robot est donc composé d'un ensemble de modules, chacun étant responsable d'une ou de plusieurs capacités. L'un des premiers défis à résoudre est de déterminer comment relier efficacement les différents modules. Pour ce faire, il faut élaborer une architecture décisionnelle qui dicte les responsabilités de chacun des modules et comment l'information circule entre ces derniers. Depuis les débuts de la robotique, beaucoup d'architectures ont été proposées. Elles peuvent être généralement classées en trois grandes catégories : délibérative, comportementale et hybride [8] [9].

I.7.1. Architecture délibérative (Hiérarchique)

Les architectures délibératives sont les premières à avoir été proposées. Comme son nom l'indique, les architectures de ce type sont basées sur des processus complètement planifiés. Par exemple, afin d'exécuter un déplacement, un robot basé sur ce type d'architecture calcule un plan complet, lui disant d'avancer de x mètres, ensuite de tourner de y degrés, et ainsi de suite. Lorsqu'un changement dans l'environnement est perçu, l'exécution est suspendue et un nouveau plan est généré.

Ce type d'architecture souffre de plusieurs lacunes importantes. Premièrement, puisque les capteurs sont imprécis et que l'environnement est dynamique et partiellement observable, il est très difficile de tout prévoir à l'avance. Pour ces raisons, il n'est pas d'une très grande utilité de tout planifier à l'avance, puisque les plans seront constamment à refaire. Un autre problème avec ce type d'architecture est que la génération de plans précis demande beaucoup de ressources (temps de calcul et mémoire) [12].

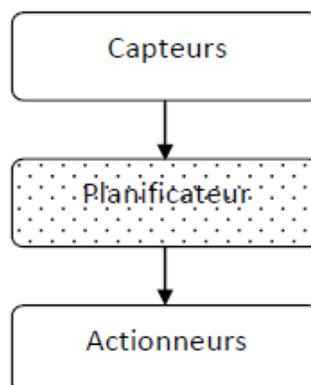


Figure I.13 Architecture délibérative (hiérarchique) [12].

I.7.2. Architecture comportementale (réactive)

L'architecture comportementale est inspirée par le comportement des insectes. L'idée générale est de développer plusieurs petits modules simples et indépendants les uns des autres et, une fois regroupés, un comportement plus intelligent émerge sans qu'il ait été spécifiquement programmé. Ce type d'architecture est complètement à l'opposé des architectures délibératives et ne fait aucune place à des processus raisonnés.

Un exemple d'architecture comportementale est illustré à la figure I.14. Celle-ci est composée de plusieurs modules simples appelés comportements et d'un module d'arbitration.

Dans sa conception, chaque comportement se limite à une seule fonctionnalité pour le contrôle du robot. Les comportements sont tous indépendants les uns des autres. Par exemple, on peut avoir des comportements pour l'évitement d'obstacles, le suivi de chemin, le suivi d'objets de couleur ou la manipulation d'objets. Les comportements sont exécutés parallèlement à une certaine fréquence.

Lors d'une itération, chaque comportement calcule une ou plusieurs commandes motrices qui sont envoyées à un module d'arbitration. Ce dernier fusionne l'ensemble des commandes reçues et calcule les commandes finales devant être envoyées à chacun des actionneurs du robot [12].

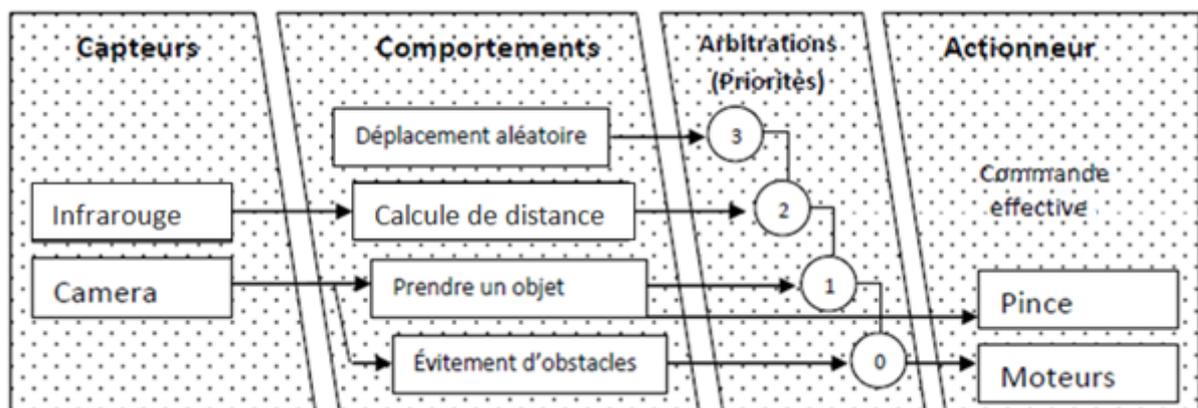


Figure I.14 Exemple d'architecture comportementale [12].

En guise d'exemple, examinons l'intégration d'un comportement d'évitement d'obstacles et d'un comportement de suivi de chemin. En présence d'un obstacle sur un côté (voir figure I.15), le comportement d'évitement d'obstacles ne fait que tourner vers le côté opposé. Indépendamment, le comportement de suivi de chemin aligne et dirige en ligne droite le robot vers la cible courante. Comme nous pouvons l'apercevoir à la figure I.16, ce comportement ignore la présence d'obstacles.

Afin de diriger le robot vers la cible sans collision, les capacités de ces deux comportements doivent être fusionnées par un mécanisme d'arbitration [12].

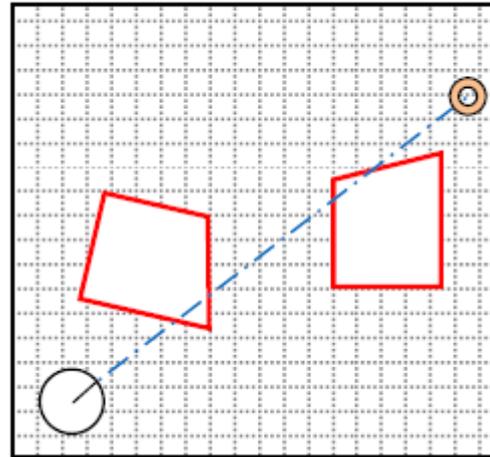
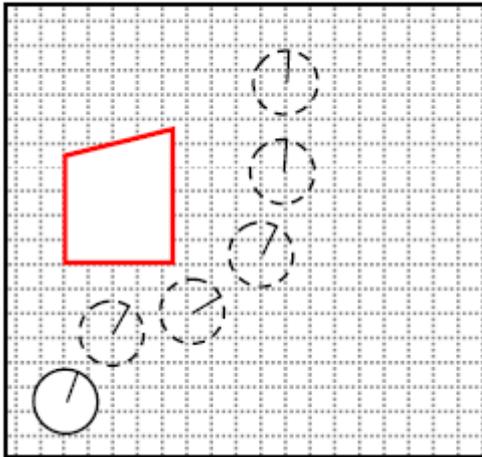


Figure I.15 Évitement d'obstacle [12]. **Figure I.16** Suivi de chemin [12].

L'approche classique consiste à utiliser une arbitration par priorité. Dans le cas présent, l'évitement d'obstacles a priorité sur le suivi de chemin. Lorsqu'aucun obstacle n'est détecté, le module d'évitement d'obstacles ne génère aucune commande et laisse le plein contrôle au module de suivi de chemin. Par contre, quand un obstacle est détecté, le module d'évitement substitue la commande du module de suivi de chemin par une commande de rotation dans le sens opposé à l'obstacle perçu. Lorsque l'obstacle est complètement évité et sorti du champ de vision des capteurs, le comportement de suivi de chemin reprend le plein contrôle du robot.

En contrepartie, les architectures comportementales ont de la difficulté à réaliser des tâches structurées, puisqu'elles ne contiennent aucun processus délibératif. En effet, les tâches complexes requièrent la capacité du robot à prédire les conséquences futures de ses actions afin de sélectionner celles qui conviennent le mieux pour la réalisation de ses activités. En d'autres mots, ces tâches complexes ont besoin d'être planifiées [12].

I.7.3. Architecture hybride

Les limitations des deux types d'architecture précédentes justifient l'émergence récente des architectures hybrides, tentant de combiner les avantages des architectures délibératives et comportementales. Elles sont généralement décomposées en plusieurs niveaux.

Dans la partie supérieure, on place les modules de type délibératif. Dans la partie inférieure, on retrouve les modules de type comportemental. La figure I.17 illustre un exemple d'architecture hybride où un planificateur s'occupe des capacités délibératives [12].

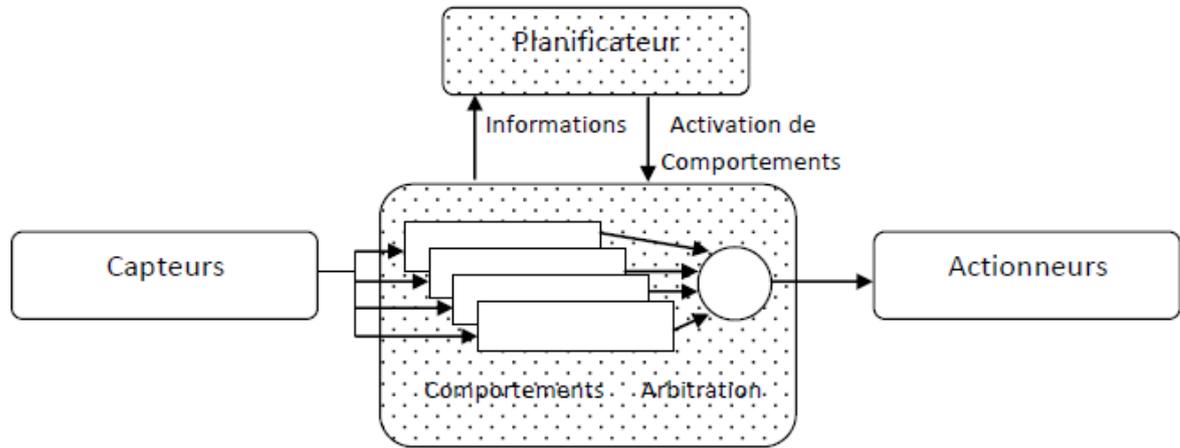


Figure I.17 Architecture hybride [12].

I.8. Conclusion

Dans ce chapitre nous avons résumé les notions de base nécessaires à la compréhension du domaine de la robotique mobile, les fonctions d'un robot mobile (la locomotion, la perception et la décision) qui donnera la vie à notre robot, ainsi que les techniques de navigation d'un robot autonome, et nous avons aussi présenté un aperçu sur les architectures de contrôle.

Chapitre II

Modélisation

II.1. Introduction

La modélisation d'un robot consiste à représenter le comportement du robot par des équations algébriques, On considère généralement, deux points de vue pour modéliser un robot mobile, un point de vue vitesse (modèle cinématique) et un point de vue des efforts mis en jeu (modèle dynamique). Les coordonnées articulaires et opérationnelles sont les paramètres qui caractérisent le modèle du robot. Les premières permettent au mécanisme de modifier sa géométrie et les secondes déterminent la position et l'orientation de robot mobile.

L'objectif de ce chapitre est de donner une représentation de robot mobile selon son modèle et structure. Nous ne présentons ici que le modèle le plus fréquemment rencontré, le robot de type uni-cycle et on représente son modèle cinématique sous forme d'équations. Ceci va nous servir notamment dans les parties suivantes pour les tests des commandes à synthétiser.

II.2. Modélisation cinématique

II.2.1. Mise en place des repères

On note $R = (O, \vec{x}, \vec{y}, \vec{z})$ un repère fixe quelconque, dont l'axe \vec{z} est vertical et $R' = (O', \vec{x}', \vec{y}', \vec{z}')$ un repère mobile lié au robot. On choisit généralement pour O' un point remarquable de la plate-forme du robot, typiquement le centre de l'axe des roues motrices s'il existe, comme illustré sur la figure (II.1).

On appelle position ou souvent posture du robot le vecteur :

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (2.1)$$

Où x et y sont respectivement l'abscisse et l'ordonnée du point O' dans R et θ l'angle (\vec{x}, \vec{x}') .

La position du robot est donc définie sur un espace M de dimension $m=3$.

La configuration d'un robot mobile est connue quand la position de tous ses points dans un repère donné est connue, on définira la configuration du robot mobile par un vecteur :

$$q = \begin{bmatrix} q_1 \\ q_2 \\ \cdot \\ \cdot \\ \cdot \\ q_n \end{bmatrix} \quad (2.2)$$

De n coordonnées appelées coordonnées généralisées. La configuration est ainsi définie sur un espace N de dimension n , appelé espace des configurations [13].

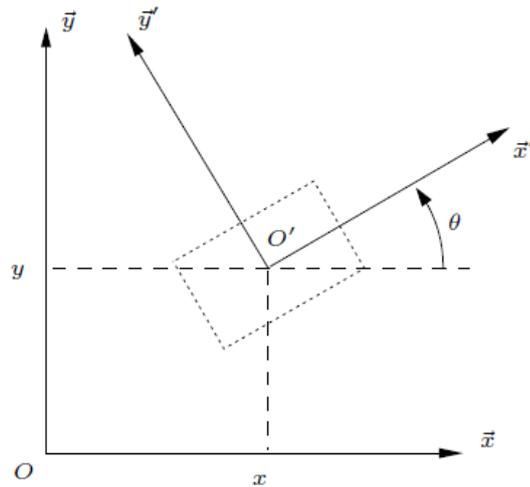


Figure II.1 Repérage d'un robot mobile [13].

II.2.2. Roulement sans glissement

Considérons une roue qui roule sans glisser sur un sol plan, comme illustré sur la figure (II.2). Soit P le centre de la roue, Q le point de contact de la roue avec le sol, φ l'angle de rotation propre de la roue et θ l'angle entre le plan de la roue et le plan (O, \vec{x}, \vec{y}) . 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 de vitesse de rotation $\vec{\omega}$ de la roue :

$$\vec{v}_Q = \vec{v}_p + \vec{\omega} \wedge \overrightarrow{PQ} = 0 \tag{2.3}$$

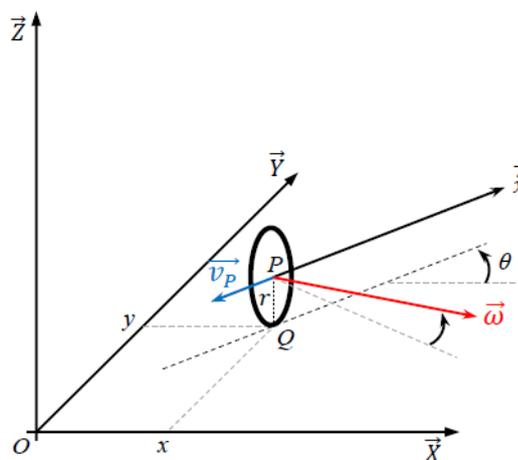


Figure II.2 Caractérisation du roulement sans glissement [13].

Les points P et Q ont pour coordonnées respectives $(x \ y \ r)^T$ et $(x \ y \ 0)^T$. Il vient alors :

$$\dot{x}\vec{x} + \dot{y}\vec{y} + \left(\dot{\theta}\vec{z} + \dot{\phi}(\sin\theta\vec{x} - \cos\theta\vec{y}) \right) \wedge (-r\vec{z}) = \vec{0} \quad (2.4)$$

$$(\dot{x} + r\dot{\phi}\cos\theta)\vec{x} + (\dot{y} + r\dot{\phi}\sin\theta)\vec{y} = \vec{0} \quad (2.5)$$

Il en résulte les deux contraintes cinématiques indépendante

$$\dot{x} + r\dot{\phi}\cos\theta = 0 \quad (2.6)$$

$$\dot{y} + r\dot{\phi}\sin\theta = 0 \quad (2.7)$$

Que l'on peut transformer pour faire apparaître les composantes de vitesse dans le plan de la roue d'une part et perpendiculairement à la roue d'autre part :

$$-\dot{x}\sin\theta + \dot{y}\cos\theta = 0 \quad (2.8)$$

$$\dot{x}\cos\theta + \dot{y}\sin\theta = -r\dot{\phi} \quad (2.9)$$

Ces contraintes traduisent le fait que le vecteur \vec{v}_p soit dans le plan de la roue et ait pour module $r\dot{\phi}$ [13].

II.2.3. Disposition des roues et centre instantané de rotation

C'est la combinaison du choix des roues et de leur disposition qui confère à un robot son mode de locomotion propre. Sur les robots mobiles, on rencontre principalement trois types de roues, voir figure (II.3) :

- Les roues fixes dont l'axe de rotation, de direction constante, passe par le centre de la roue.
- Les roues centrées orientables, dont l'axe d'orientation passe par le centre de la roue.
- Les roues décentrées orientables, souvent appelées roues folles, pour lesquelles l'axe d'orientation ne passe pas par le centre de la roue [14] [15].

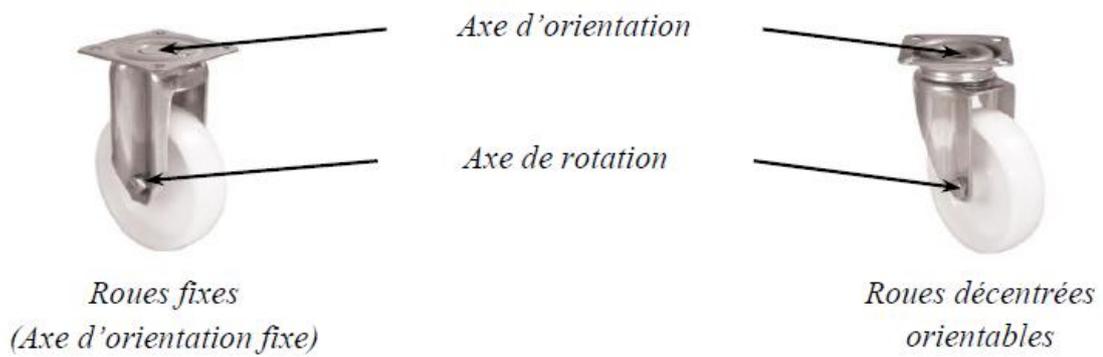


Figure II.3 Les principaux types de roues [14].

De manière anecdotique on rencontrera aussi des systèmes particuliers, tels que les roues suédoises, les roues à plusieurs directions de roulement... etc.

- Le Centre Instantané de Rotation (CIR) d'un robot mobile, est le point d'intersection des lignes joignant les axes de rotation des roues.
- Le CIR est un point de vitesse nulle autour duquel tourne le robot de façon instantanée [15].

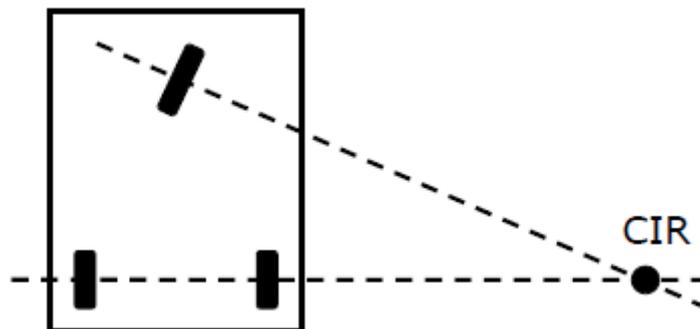


Figure II.4 Centre instantané de rotation [15].

II.2.4. Modélisation cinématique du robot mobile de type uni-cycle

Pour la modélisation cinématique du robot mobile, nous avons besoin de déterminer les équations de quelques paramètres du robot. Un robot mobile uni-cycle est actionné par deux roues indépendantes et possédant éventuellement un certain nombre de roues folles assurant sa stabilité, le schéma des robots de type uni-cycle est donné par la figure (II.5-6) [14].

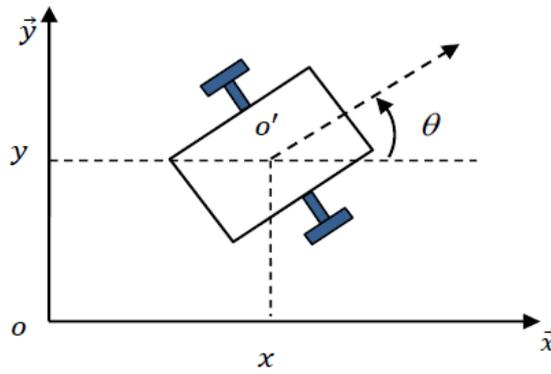


Figure II.5 Robot mobile de type uni-cycle [14].



Figure II.6 Le Robot mobile Pioneer P3-DX [13].

_ Centre instantané de rotation

Les roues motrices ayant même axe de rotation, le CIR 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' , voir figure (II.7). Soit L l'entre-axe et ω la vitesse de rotation du robot autour du CIR. Alors les vitesses des roues droite et gauche, respectivement notées v_d et v_g et définies dans la figure suivant, vérifient :

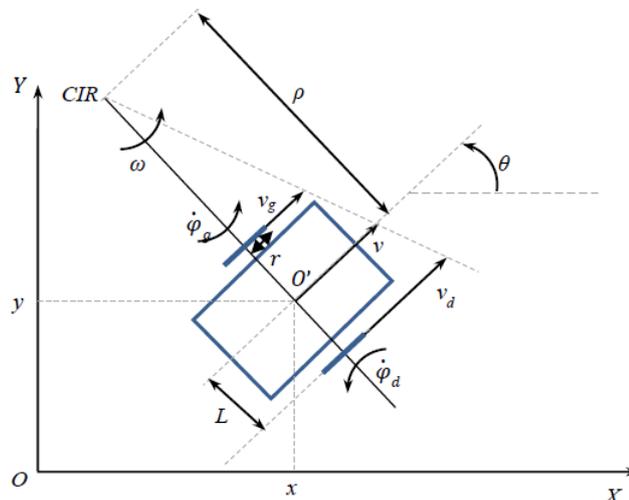


Figure II.7 Centre instantané de rotation d'un robot de type uni-cycle [14].

$$v_d = -r\dot{\varphi}_d = (\rho + L)\omega \quad (2.10)$$

$$v_g = r\dot{\varphi}_g = (\rho - L)\omega \quad (2.11)$$

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

$$\rho = L \frac{\dot{\varphi}_d - \dot{\varphi}_g}{\dot{\varphi}_d + \dot{\varphi}_g} \quad (2.12)$$

$$\omega = -\frac{r(\dot{\varphi}_d + \dot{\varphi}_g)}{2L} \quad (2.13)$$

L'équation (2.12) permet de situer le CIR sur l'axe des roues. Par ailleurs ces équations expliquent deux propriétés particulières du mouvement des robots de type uni-cycle :

- Si $\dot{\varphi}_d = \dot{\varphi}_g$, le robot se déplace en ligne droite.
- Si $\dot{\varphi}_d = -\dot{\varphi}_g$, alors le robot effectue une rotation sur lui-même.

L'utilisation de ces deux seuls modes de locomotion, bien que limitée, permet de découpler les mouvements et de fournir une solution simple pour amener le robot d'une position à une autre. C'est sans doute une des raisons du succès de ce type de robots. Pour élaborer une stratégie plus fine de déplacement, il est cependant intéressant de savoir comment la position du robot est reliée à la commande de ses roues [14].

En ce qui concerne la commande, si l'on se contente de traiter le cas cinématique, on peut considérer que celle-ci est donnée, au plus bas niveau, par les vitesses de rotation des roues. Ceci étant, on préfère généralement exprimer cette commande par la vitesse longitudinale du robot, notée v (en O') et sa vitesse de rotation $\dot{\theta}$ (autour de O'). Il y a en effet équivalence entre les deux représentations. D'une part, on a :

$$v = \frac{v_d + v_g}{2} \quad (2.14)$$

D'autre part, la vitesse de rotation du robot est égale à la vitesse de rotation autour du CIR :

$$\omega = \dot{\theta} = -\frac{r(\dot{\varphi}_d + \dot{\varphi}_g)}{2L} = \frac{v_d - v_g}{2L} \quad (2.15)$$

Conformément à l'équation (2.14), On montre que ces relations sont parfaitement inversibles et qu'il y a ainsi équivalence entre les couples $(\dot{\phi}_d, \dot{\phi}_g)$ et (v, ω) . Désormais on utilise plutôt ce dernier couple de grandeurs, plus parlantes, quitte à calculer ensuite les angles ou vitesses de consigne des asservissements des roues [14].

II.2.5. Modèle cinématique en posture

Relier la dérivée de la position à la commande $u = (v \ \omega)^T$ est facile. Une simple considération géométrique donne :

$$u = \begin{cases} \dot{x} = v \cos\theta \\ \dot{y} = v \sin\theta \\ \dot{\theta} = \omega \end{cases} \quad (2.16)$$

Ce qui s'écrit, sous forme matricielle :

$$\dot{q} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = C(q).u \quad (2.17)$$

$$C(q) = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \quad (2.18)$$

Avec, $C(q)$ et u sont, respectivement, la matrice de Jacobéen et le vecteur de vitesse linéaire et de vitesse de rotation du robot. Ce modèle est appelé modèle cinématique en posture du robot mobile [13] [16].

II.3. Conclusion

Nous avons présenté dans ce chapitre le modèle cinématique pour un robot mobile non holonome de type uni-cycle à deux roues motrices en arrière et une roue avant pour la stabilité du robot mobile. Ce modèle sera exploité dans les chapitres suivants pour commander le robot mobile de type uni-cycle.

Chapitre III

La logique floue

II.1. Introduction

Les bases théoriques de la logique floue ont été réellement introduites en 1965 par le professeur Lotfi Zadeh de l'université de Berkeley en Californie, et Les premières applications en contrôle floue ont été effectuées par Mamdani en 1974, et son équipe en utilisant une approche de type heuristique qui a ses racines dans le champ de l'intelligence artificielle. L'idée développée a été d'utiliser les techniques floues pour leur capacité à exprimer, de manière simple et sous une forme linguistique, des règles de comportement comme cela été mis en évidence par Zadeh.

Une autre approche, qui a donné des développements très riches, a été proposée par Takagi et Sugeno en 1985. Elle repose aussi sur une représentation à base de règle. Toutefois, à la différence des contrôleurs de type Mamdani, la partie conclusion des règles s'exprime de manière numérique sous la forme d'une constante, d'un polynôme ou, de manière plus générale, par une fonction ou une équation différentielle dépendant des variables associées aux prémisses des règles. Ce type de représentation peut être utilisé pour modéliser un processus ou pour synthétiser un contrôleur flou.

Dans ce chapitre, nous discutons sur la terminologie de base de la logique floue, et en donnant un rappel sur les Concepts de base de la logique floue, Et les différentes méthodes de la commande de la logique floue talquent Mamdani et Sugeno.

III.2. Logique floue et ensemble flou

La logique floue peut être vue comme une extension de la logique booléenne; de plus, elle permet de traiter des variables linguistiques dont les valeurs sont des mots ou expressions du langage naturel. On présente les propriétés de base des ensembles classiques et des ensembles flous [2].

III.2.1. Concept de base d'un ensemble floue

Un ensemble classique A de U est défini par une fonction caractéristique notée $\mu_A(x)$ figure (III.1. (a)) telle que:

$$\mu_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases} \quad (3,1)$$

Le concept de base de la théorie des ensembles flous est la notion de sous-ensemble floue. Cette notion provient du constat que «très souvent, les classes d'objets rencontrés dans le monde physique ne possèdent pas de critères d'appartenance bien définis». Soit U une

collection d'objets (par exemple $U = \mathbb{R}^n$) appelée univers de discours. Un ensemble flou A dans U est caractérisé par une fonction d'appartenance, notée: $\mu_A(x)$, ($\mu_A: U \rightarrow [0,1]$), qui appliquée à un élément x de U , retourne un degré d'appartenance $\mu_A(x)$ de x à A figure (III.1 (b)). Un ensemble flou peut être considéré comme une généralisation de l'ensemble classique. La fonction d'appartenance d'un ensemble classique peut prendre seulement deux valeurs $\{0,1\}$. Un ensemble flou peut être représenté comme un ensemble de paires ordonnées [7] [18] :

$$A = \{x, \mu_A(x) / x \in U\} \tag{3,2}$$

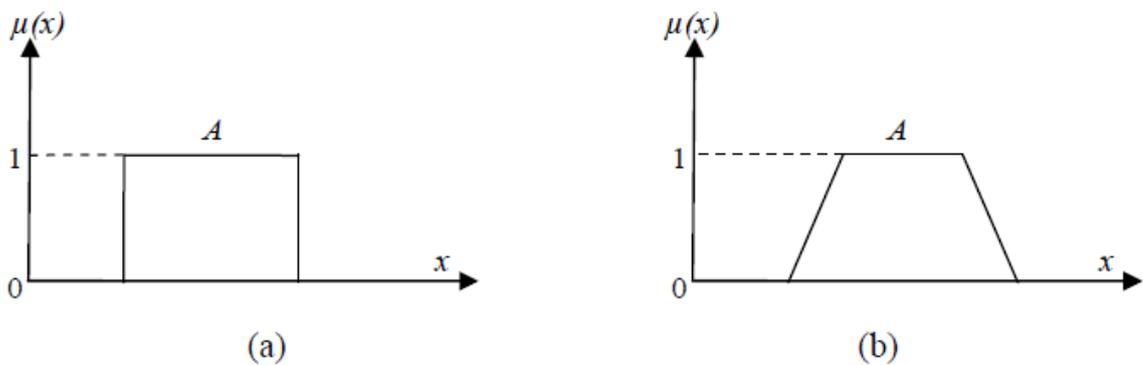


Figure III.1 Exemples des fonctions d'appartenance
 (a) Logique classique (b) Logique floue [2].

Si U est discret, A est représenté par :

$$A = \sum_{x_i} \mu_A(x_i) / x_i \tag{3.3}$$

Si U est continu, A est représenté par :

$$A = \int_x \mu_A(x_i) / x_i \tag{3.4}$$

Les ensembles flous ont le grand avantage de constituer une représentation mathématique des termes linguistiques largement utilisés dans le langage naturel des experts [2] [18].

III.2.1.1. Caractéristiques d'un ensemble flou

Un sous-ensemble flou A sur un univers de discours U , est représenté selon figure (III.2) à travers sa fonction caractéristique μ_A . Il peut également être décrit par un certain nombre de caractéristiques comme [17]:

- Son support :

$$\text{support}(A) = \{x \in U / \mu_A(x) \neq 0\} \quad (3,5)$$

- Sa hauteur :

$$\text{hauteur}(A) = \sup_{x \in U} \{\mu_A(x)\} \quad (3,6)$$

- Son noyau :

$$\text{noyau}(A) = \{x \in U / \mu_A(x) = 1\} \quad (3,7)$$

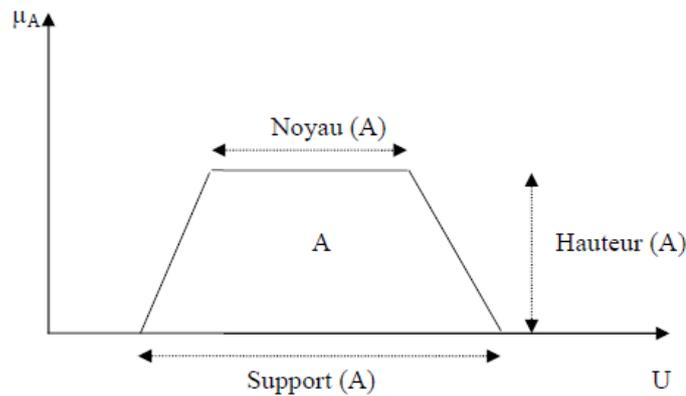


Figure III.2 Représentation d'un sous-ensemble flou et principales caractéristiques [17].

III.2.1.2 Fonctions d'appartenance

Il existe plusieurs types de fonctions d'appartenance et le choix de forme dépend de l'application individuelle, Voir figure (III.3) [18]:

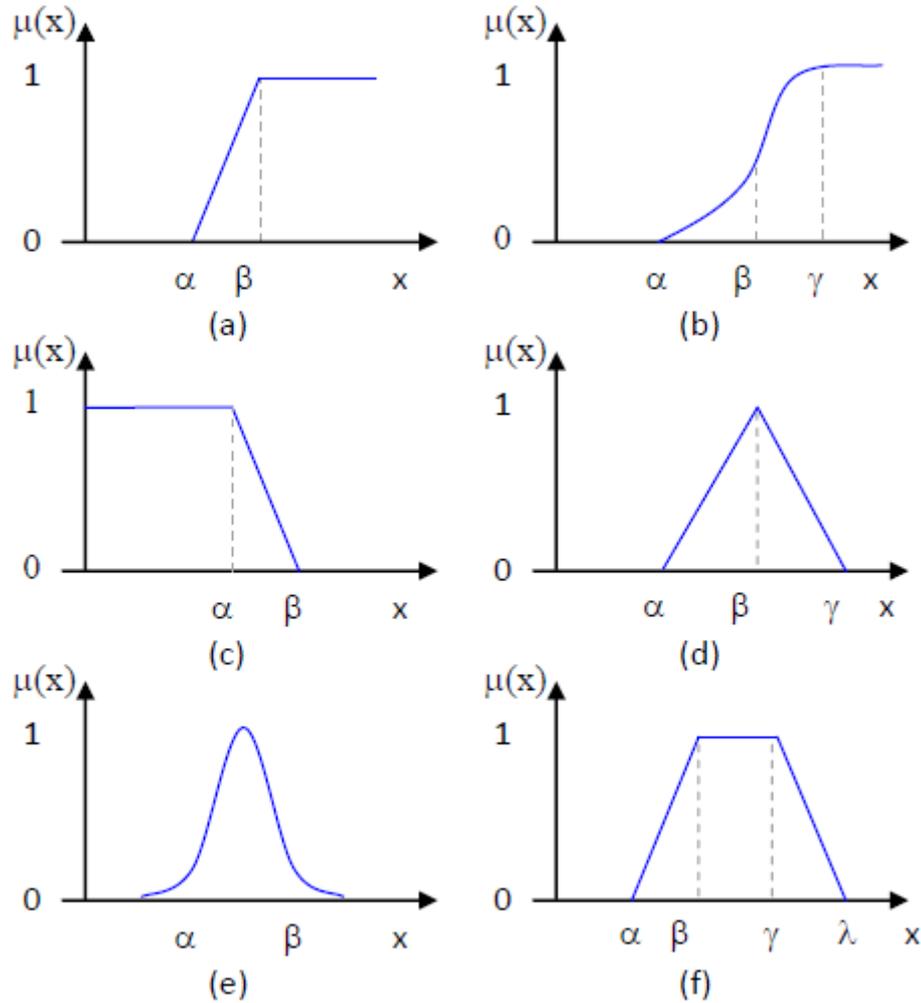


Figure III.3 Types de fonctions d'appartenances: (a) fonction – Γ (demi trappez droit); (b) fonction – S (sigmoïde); (c) fonction – L (demi trappez gauche); (d) fonction – Λ (triangle); (e) fonction Gaussienne; (f) fonction - Π (trapézoïdale) [12].

Définitions mathématique des fonctions d'appartenance [12] [19]:

Fonction - Γ , $\Gamma : U [0,1]$:

$$\Gamma(x, \alpha, \beta) = \begin{cases} 0 & x < \alpha \\ \frac{(x-\alpha)}{(\alpha-\beta)} & \alpha \leq x \leq \beta \\ 1 & x > \beta \end{cases} \quad (3,8)$$

Fonction - L, $L : U [0,1]$:

$$L(x, \alpha, \beta) = \begin{cases} 1 & x < \alpha \\ \frac{(x-\beta)}{(\alpha-\beta)} & \alpha \leq x \leq \beta \\ 0 & x > \beta \end{cases} \quad (3,9)$$

Fonction - Λ , $\Lambda : U [0,1]$:

$$\Lambda(x, \alpha, \beta) = \begin{cases} 0 & x < \alpha \\ \frac{(x-\alpha)}{(\beta-\alpha)} & \alpha \leq x \leq \beta \\ \frac{(x-\gamma)}{(\beta-\gamma)} & \beta \leq x \leq \gamma \\ 0 & x > \gamma \end{cases} \quad (3,10)$$

Fonction Gaussienne avec (c : le centre du gaussien, σ : l'écart-type)

$$\mu_A(x) = \exp\left(\frac{-0.5(x-c)^2}{\sigma^2}\right) \quad (3, 11)$$

Fonction - S, S: [0, 1] :

$$\mu_A(x) = \frac{1}{1+\exp\{-a(x-c)\}} \quad (3,12)$$

III.2.1.3. Variables linguistiques

La description d'une certaine situation, d'un phénomène ou d'un procédé contient en général des qualificatifs flous tels que :

- peu, beaucoup, énormément.
- rarement, fréquemment, souvent.
- froid, tiède, chaud.
- petit, moyen, grand ...etc.

Généralement, une variable linguistique est représentée par un triplet (V, U, Tv) où V est la variable linguistique elle-même, U est l'univers de discours et Tv l'ensemble des caractérisations floues de la variable.

Considérons par exemple la variable "taille" définie sur l'ensemble des entiers positifs et caractérisée par les ensembles flous (petit, moyen, grand.), La variable taille est alors représentée par le triplet suivant : {taille, \mathbb{R}^+ , (petit, moyen, grand)} [7].

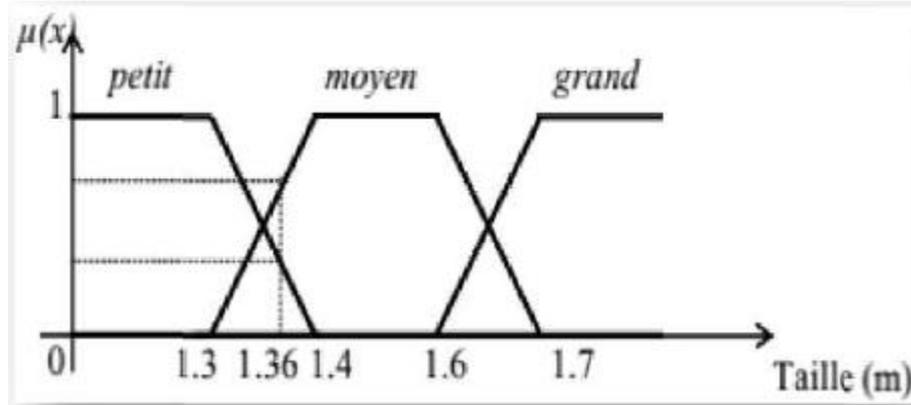


Figure III.4 L'ensemble flou de variable "taille" [7].

Une personne mesurant 1.36 m est petite avec un degré d'appartenance 0.4 et moyenne avec un degré de 0.6.

III.2.2. Opérations sur les ensembles flous

Soient A et B des ensembles flous dans l'univers X, décrits par leurs fonctions d'appartenance $\mu_A(x), \mu_B(x)$ respectivement [7].

A) Egalité

De deux ensembles flous A et B est notée $A=B$, elle est donnée par :

$$\forall x \in X, \mu_A(x) = \mu_B(x) \quad (3.13)$$

B) Inclusion

On dit que A est inclus dans B, noté $A \subset B$, si et seulement si :

$$\forall x \in X, \mu_A(x) \leq \mu_B(x) \quad (3.14)$$

C) Intersection

Notée $A \cap B$, définie par :

$$\forall x \in X, \mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (3.15)$$

D) Union

Notée $A \cup B$, définie par l'équation :

$$\forall x \in X, \mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (3.16)$$

E) Complément

D'un ensemble floue A, noté \bar{A} , est défini par :

$$\forall x \in X, \mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (3.17)$$

Remarque

$$\forall x \in X, A \cap \bar{A} = \emptyset \quad (3.18)$$

La figure suivante représente les opérations complément, intersection et union des ensembles flous [1]:

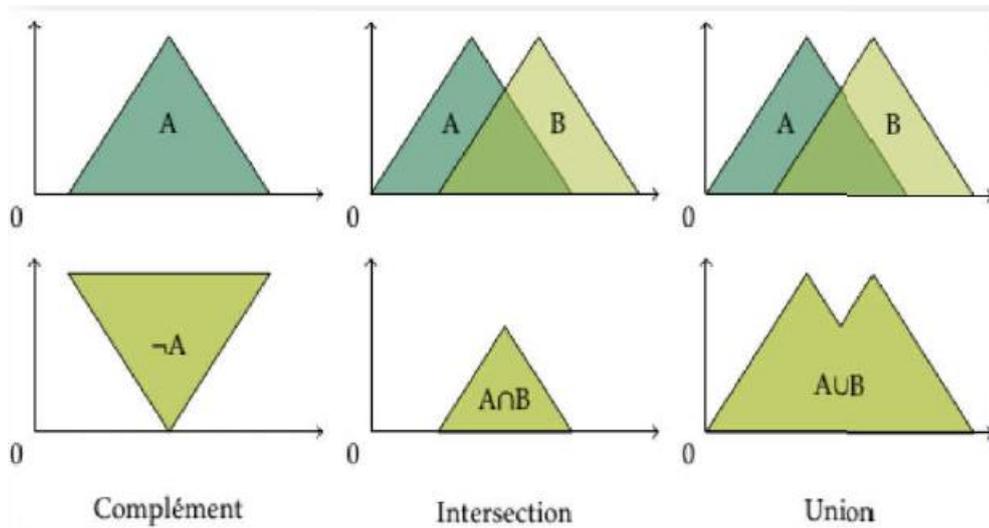


Figure III.5 Les fonctions d'appartenance associées aux opérations :
Complément ; Intersection ; Union [7].

III.3. Systèmes d'inférence floue (SIF)

Un Système d'Inférence Floue (SIF) a comme but de transformer les données d'entrée en données de sortie à partir de l'évaluation d'un ensemble des règles. Les entrées sont issues du processus de fuzzification et l'ensemble de règles sont définies par le sa l'expert. Un SIF voir figure (III.6) est constituée de trois étapes [7] [19]:

- 1) Fuzzification.
- 2) Inférence.
- 3) Défuzzification.

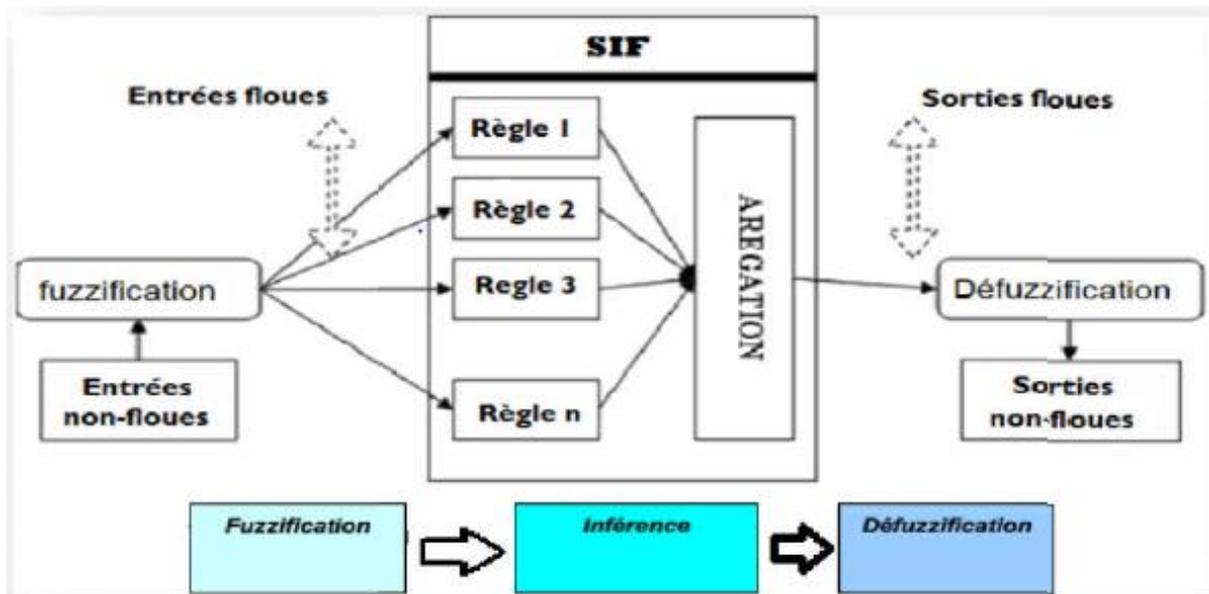


Figure III.6 Structure d'un SIF [7].

III.4. Représentation mathématique des systèmes flous

III.4.1. Fuzzification

La fuzzification est l'opération de rendre une entrée classique des valeurs d'entrée sont traduites en concepts linguistiques représentés comme des ensembles flous. Les fonctions d'appartenance sont appliquées aux mesures et des degrés de vérité sont établis pour chaque proposition.

Comment fuzzifier ?

Pour fuzzifier, il faut donner:

- 1) L'univers du discours est: Plage de variations possibles de l'entrée considérée.
- 2) Une partition en classe floue de cet univers.
- 3) Les fonctions d'appartenances de chacune de ces classes.
 - Il faut fuzzifier les entrées ET les sorties du processus floue.

Exemple :

Selon les valeurs des entrées, le système flou indiquera qu'en sortie la puissance de chauffe devra prendre les valeurs de sortie « faible » ou « moyenne » ou « forte » [7].

III.4.2. Inférence

III.4.2.1. Base de règles floues

Une base de règles floues R est une collection de règles floues de la forme « SI-ALORS », $R = [R_1, R_2, \dots, R_m]$. Une règle floue R_i est donnée sous le modèle de « Mamdani » comme suit :

$$R_i: \text{Si } x_1 \text{ est } A_{i1} \text{ et } x_2 \text{ est } A_{i2} \text{ et ... et } x_n \text{ est } A_{in} \text{ ALORS } y \text{ est } B_i \quad (3,19)$$

Ou sous le modèle de « Takagi-Sugeno (TS) » sous la forme:

$$R_i: \text{Si } x_1 \text{ est } A_{i1} \text{ et } x_2 \text{ est } A_{i2} \text{ et ... et } x_n \text{ est } A_{in} \text{ ALORS } y \text{ est } f_i(x) \quad (3,20)$$

$f_i(x)$ Est un polynôme (hyperplan). Si le polynôme est d'ordre zéro (sous forme d'une constante), on dit que le modèle est de TS d'ordre zéro TS(0), et si le polynôme est du premier ordre, on dit que le modèle est de TS d'ordre un [2] [18].

III.4.3.2. Moteur d'inférence flou

Le moteur d'inférence floue utilise la base des règles floues pour effectuer une transformation à partir des ensembles flous dans l'espace d'entrée vers les ensembles flous dans l'espace de sortie en se basant sur les opérations de la logique floue. L'antécédent de la règle R , définit un produit cartésien de $A_{i1}, A_{i2}, \dots, A_{in}$, et la règle elle-même R_i , est vue comme une implication. Soit $A_i = A_x \circ R_i$ dans V . La fonction d'appartenance de B_i est définie par la règle compositionnelle (3,21) :

$$\mu_{B_i}(y) = \text{Sup}_{x \in A_x} (\mu_{A_i}(x) * \mu_{R_i}(x, y)) \quad (3,21)$$

Dans le jeu de règles du système floue interviennent les opérateurs flous "ET(AND)" et "OU(OR)". L'opérateur "ET" s'applique aux variables à l'intérieur d'une règle, tandis que l'opérateur "OU" lie les différentes règles. Plusieurs types de raisonnement flou ont été proposés dans la littérature suivant la réalisation des opérateurs flous "ET" et "OU" et le type des règles floues utilisées. Les deux moteurs d'inférence floue les plus utilisés sont: le moteur de Mamdani, et de Sugeno [2].

III.4.3.2.1. Méthode de Mamdani

Mamdani fut le premier à utiliser la logique floue pour la synthèse des commandes. Il a utilisé le minimum comme opérateur de conjonction et d'implication.

Les règles correspondant à l'équation (3,14) où B^i est un sous ensemble floue. Généralement, les B^i forment une partition de l'espace de sortie. L'inférence floue correspond aux étapes suivantes, pour un vecteur d'entrée $x = (x_1, \dots, x_n)^i$:

1. Calcul du degré d'appartenance de chaque entrée aux différents sous ensembles floues :

$$\mu_{A_j}^i(x_j), \text{ pour } j = 1 \text{ à } n \text{ et } i = 1 \text{ à } n \quad (3,22)$$

Où : n est la dimension d'espace d'entrée et N : Nombre des règles floues.

2. Calcul de valeur de vérité de chaque règle, pour $i = 1$ à N :

$$\alpha_i(x) = \min_j[\mu_{A_j}^i(x_j)] \quad (3,23)$$

3. Calcul de la contribution de chaque règle selon l'équation (3.17) :

$$\mu_i(y) = \min(\alpha_i(x), u_{B_1}(y)) \quad (3,24)$$

4. Agrégation des règles en utilisant l'opérateur max :

$$\mu_i = \max_i(u_i(y)) \quad (3,25)$$

Le résultat est donc un sous ensemble floue caractérisée par sa fonction d'appartenance. Pour obtenir une conclusion exacte, il faut défuzzifier. Plusieurs méthodes peuvent être utilisées : la méthode de centre de gravité donne :

$$y = \frac{\int uU(u) du}{\int U(u) du} \quad (3,26)$$

Ou dans le cas discret:

$$y = \frac{\sum_k u_k U(u_k)}{\sum_k U(u_k)} \quad (3,27)$$

Cette implémentation est appelée (min, max, barycentre). Ils existent plusieurs variantes de la méthode de Mamdani comme (min, produit, barycentre) et (produit, somme, barycentre)...etc [7].

III.4.3.2.2. Méthode de Takagi-Sugeno

Les SIFs de type Takagi-Sugeno constituent un cas particulier important; ou la conclusion n'est pas symbolique mais une fonction des entrées : $B^i = f(x_1, \dots, x_n)$, la forme la plus utilisée est la suivante:

$$B^i = \sum_{j=0}^n b_j^i x_j \quad (3,28)$$

Un SIF de type Takagi-Sugeno réalise une fusion de modèles locaux par interpolation. Un SIF est dite d'ordre zéro ou de méthode de Takagi-Sugeno simplifiée si la conclusion est une constante, et SIF d'ordre 1 quand la conclusion est un hyperplan (l'équation 3.28). Le modèle de Takagi-Sugeno d'ordre 0 est le plus utilisé. Dont les conclusions sont des nombres réels, ou singletons, (pouvant être considérés comme la valeur modale d'un sous ensemble floue).

Pour le vecteur d'entrée $x = (x_1, \dots, x_n)^t$ la sortie d'un système d'inférence floue de type Takagi-Sugeno d'ordre 0 est calculée selon les étapes suivantes :

1. Calcul du degré d'appartenance de chaque entrée aux différents sous ensembles floues :

$$\mu_{A_j}^i(x_j), \text{ pour } j = 1 \text{ à } n \text{ et } i = 1 \text{ à } n \quad (3,29)$$

2. Calcul de la valeur de vérité de chaque règle pour un vecteur d'entrée x, pour i =1 à N

$$\alpha_i(x) = ET(u_{A_1}^i(x_1), \dots, u_{A_n}^i(x_n)) \quad (3,30)$$

3. Calcul de la sortie :

$$y = \frac{\sum_{i=1}^N \alpha_i(x).b^i}{\sum_{i=1}^N \alpha_j(x)} \quad (3,31)$$

Les SIFs de type Takagi-Sugeno permettent un passage aisé d'une expression symbolique définie par la base de règle à une valeur numérique exploitable [7].

III.4.4. Défuzzification

La défuzzification consiste à transformer l'ensemble flou résultant de l'agrégation des règles en une grandeur de commande précise. Là aussi il existe plusieurs méthodes [7]. Les deux principales méthodes de défuzzification sont :

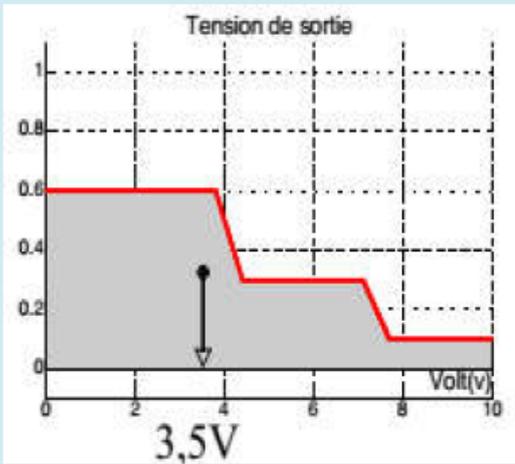
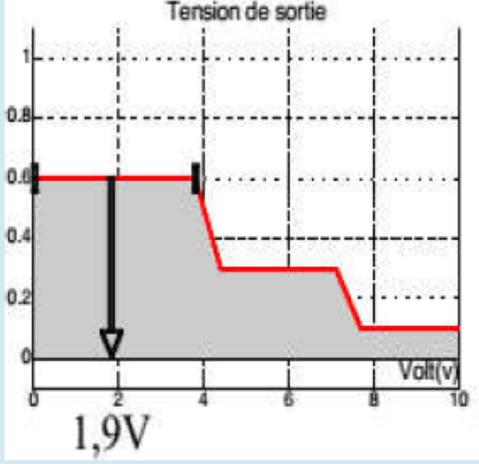
1. Méthode du centre de gravité (COG)	2. Méthode moyenne des maximums (MM)
C'est l'abscisse du centre de gravité de la surface sous la courbe résultat.	C'est la moyenne des valeurs de sorties les plus vraisemblables.
$\text{Sortie} = \frac{\int_U Y \cdot u(y) \cdot dy}{\int_U u(y) \cdot dy}$	$\text{Sortie} = \frac{\int_S Y \cdot dy}{\int_S dy}$
U=Univers du discours = Toutes les valeurs de sorties considérées.	où $S = \{y_0 \in U / u(y_0) = \text{SUB}(u(y))\}$, $y \in U$
	

Tableau III.1 : Les deux principales méthodes de défuzzification [7].

III.5. Conclusion

Dans ce chapitre, nous avons présentés les notions générales de la logique floue et la composition d'un système de commande floue. L'accent a particulièrement été mis sur les différentes étapes dans le traitement des règles d'un contrôleur flou et les différents types de commande en utilisant la logique floue.

Retenons que l'intérêt majeur de la logique floue en commande réside dans sa capacité à traduire une stratégie de contrôle d'un opérateur qualifié en un ensemble de règles linguistique facilement interprétables.

Malheureusement la manipulation de règles non précises peut générer un nombre d'erreurs non négligeable. La mise en place d'un système floue nécessite donc une attention particulière lors de la phase de test de manière à détecter les éventuelles aberrations du système.

Dans le chapitre suivant, nous allons présenter notre architecture de contrôleur floue pour une navigation autonome d'un robot mobile.

Chapitre IV

Résultat de simulation

IV.1. Introduction

La théorie de la logique floue est caractérisée par la capacité de modéliser et de traiter des informations incertaines et imprécises. L'idée principale de la navigation d'un robot mobile basée sur les comportements et d'identifier des réponses différentes selon les données sensorielles.

La connaissance de l'opérateur humain se présentera sous la forme d'un ensemble de règles linguistiques floues de type Si-Alors. Ces règles produisent une décision approximative de la même manière comme un expert.

Dans ce chapitre, on va appliquer une commande floue type-1 pour commander la vitesse des roues d'un robot mobile non holonome de type uni-cycle, et définit certains comportements élémentaires tels que: le déplacement du robot mobile d'une position initiale vers une quelconque destination désirée, et l'évitement d'obstacles, on va utiliser deux méthodes de commande Mamdani et Sugeno.

IV.2. Modèle du robot mobile utilisé dans notre travail

Le robot mobile utilisé dans la simulation est un robot mobile de type uni-cycle est actionné par deux roues indépendantes commandées par deux moteurs et possédant éventuellement un certain nombre de roues folles assurant sa stabilité figure (IV.1), On suppose que le robot mobile dans le cas réel est muni d'un ensemble des capteurs pour mesurer la distance entre le robot et l'obstacle, et la vitesse du robot mobile, La stratégie de navigation dans notre travaille est la navigation réactive.

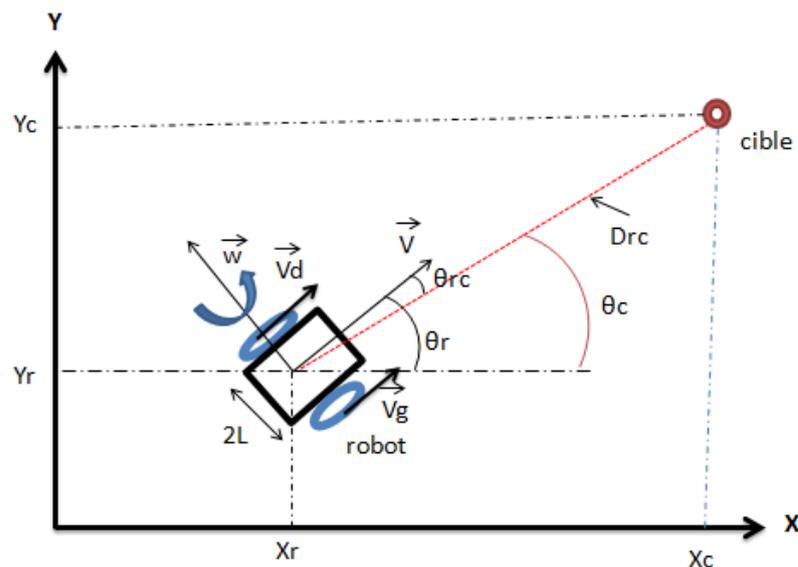


Figure IV.1 le robot mobile utilisé.

(x_r, y_r) : La position de robot.

(x_c, y_c) : Les coordonnées de point cible.

V : La vitesse linéaire.

W : La vitesse angulaire.

V_d : La vitesse de la roue droite.

V_g : La vitesse de la roue gauche.

θ_r : L'orientation du robot.

θ_c : L'orientation de la cible.

θ_{rc} : L'angle entre l'orientation actuelle du robot et celle de la cible.

D_{rc} : La distance entre le robot et la cible.

Le modèle cinématique du robot mobile est donné par :

$$\begin{cases} \dot{x} = V \cdot \cos(\theta_r) \\ \dot{y} = V \cdot \sin(\theta_r) \\ \dot{\theta} = W \end{cases} \quad (4.1)$$

$$\begin{cases} \dot{x} = \frac{V_d + V_g}{2} \cdot \cos(\theta_r) \\ \dot{y} = \frac{V_d + V_g}{2} \cdot \sin(\theta_r) \\ \dot{\theta} = \frac{V_d - V_g}{2L} \end{cases} \quad (4.2)$$

On utilise la version discrète de ce modèle qui est comme suit :

$$\begin{cases} x_{k+1} = x_k + \frac{V_{dk} + V_{gk}}{2} \cdot T \cdot \cos(\theta_{rk}) \\ y_{k+1} = y_k + \frac{V_{dk} + V_{gk}}{2} \cdot T \cdot \sin(\theta_{rk}) \\ \theta_{rk+1} = \theta_{rk} + T \cdot \frac{V_{dk} - V_{gk}}{2L} \end{cases} \quad (4.3)$$

✚ Calcul de la distance entre le robot et la cible

$$\begin{cases} e_x = x_c - x_r \\ e_y = y_c - y_r \end{cases} \quad (4.4)$$

e_x : L'erreur entre x_r de robot et x_c de cible.

e_y : L'erreur entre y_r de robot et y_c de cible.

$$D_{rc} = \sqrt{e_x^2 + e_y^2} \quad (4.5)$$

✚ Calcul de l'orientation entre le robot et la cible

$$\theta_c = \tan^{-1} \left(\frac{e_y}{e_x} \right) \quad (4.6)$$

$$\theta_{rc} = \theta_c - \theta_r \quad (4.7)$$

Notre travail s'articule essentiellement sur le développement d'un contrôleur flou d'exécution de mouvement pour un robot mobile, Le contrôleur flou développé gère à la fois les comportements de navigation.

IV.3. Méthode de Mamdani

IV.3.1. Comportement de navigation robot-cible

La navigation robot-cible c'est une navigation définie la planification de robot vers la cible dans un environnement sans obstacles.

IV.3.1.1. Structures de base de notre contrôleur flou robot-cible

La structure de base de notre contrôleur flou robot-cible est composée de trois blocs: Le fuzzification, inférence, et défuzzification.

Pour le comportement de navigation on utilise deux entrées: la distance entre le robot mobile et la cible (D_{rc}), et L'angle entre l'orientation actuelle du robot et celle de la cible (θ_{rc}). Les sorties de notre contrôleur sont la vitesse de les roues gauche V_g et droite V_d .

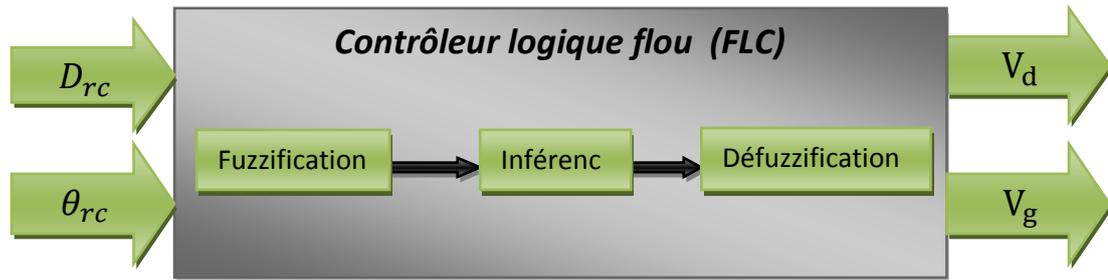


Figure IV.2 Structure du contrôleur de navigation robot-cible (Mamdani).

IV.3.1.1.1. Fuzzification

Les valeurs délivrées par les capteurs du robot sont traduites en sous-ensemble flous caractérisant les variables linguistiques associées.

Après plusieurs test de fonction d'appartenance on utilisé les fonctions d'appartenance (trapézoïdale, triangle) pour les variables d'entrée et de sortie.

- Les valeurs de l'entrée, sont indiquées par les symboles linguistiques :

➤ D_{rc} : petit (P), moyen (M), grand (G).

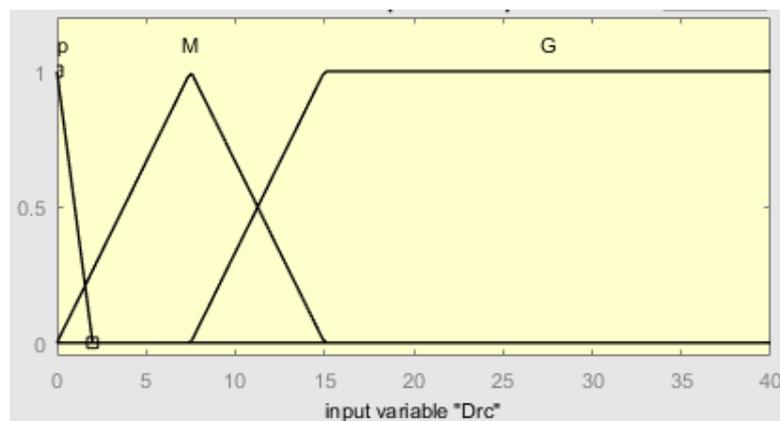


Figure IV.3 Fonctions d'appartenance de D_{rc} (Mamdani).

➤ θ_{rc} : grand négatif (GN), négatif (N), zéro (Z), positif (P), grand positif (GP).

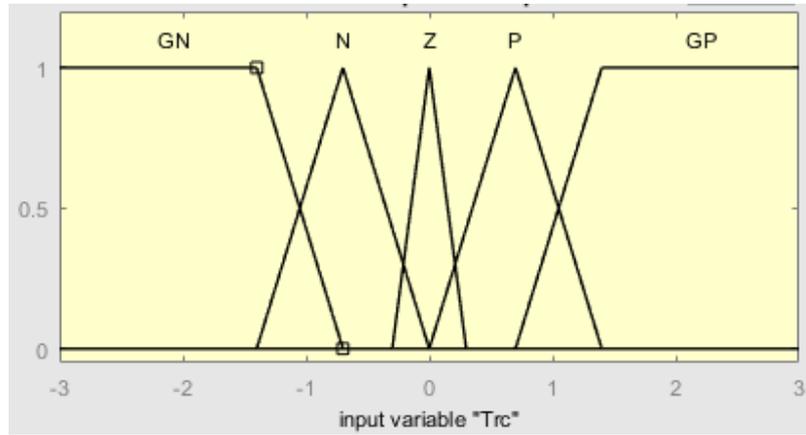


Figure IV.4 Fonctions d'appartenance de θ_{rc} (Mamdani).

- Les valeurs de sortie, sont indiquées par les symboles linguistiques :

➤ V_d : zéro (**Z**), faible (**F**), moyen (**M**), rapide (**R**).

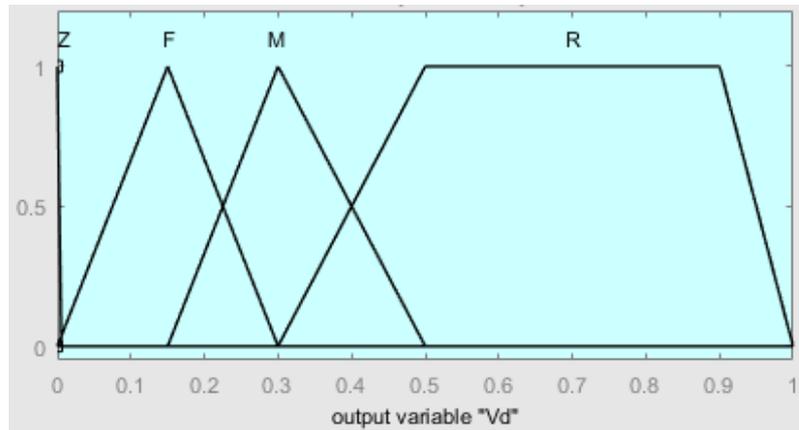


Figure IV.5 Fonctions d'appartenance de V_d (Mamdani).

➤ V_g : zéro (**Z**), faible (**F**), moyen (**M**), rapide (**R**).

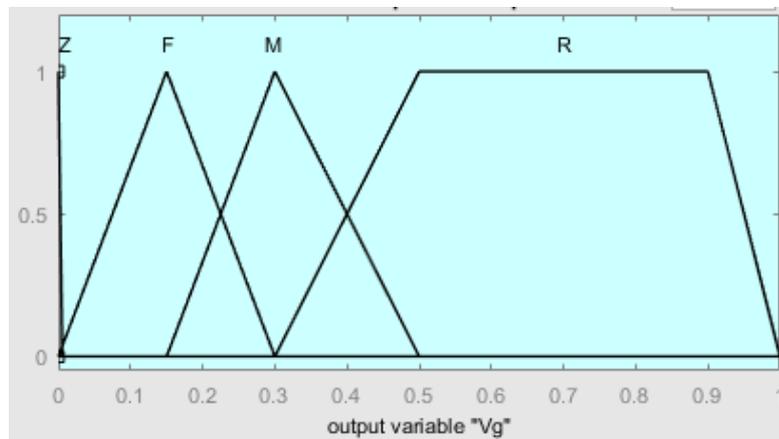


Figure IV.6 Fonctions d'appartenance de V_g (Mamdani).

IV.3.1.1.2. Inférence

L'étape d'inférence fait appel à 15 règles « Si- alors » représentant le lien entre les différentes variables entrée-sortie. Le tableau (IV.1) décrit les règles floues utilisées pour ce comportement.

θ_{rc} D_{rc}	GN		N		Z		P		GP	
	V_d	V_g								
P	M	Z	F	Z	Z	Z	Z	F	Z	M
M	R	F	M	F	M	M	F	M	F	R
G	R	M	M	F	R	R	F	M	M	R

Tableau IV.1 Les règles d'Inférence du contrôleur navigation robot-cible (Mamdani).

IV.3.1.1.3. Défuzzification

La méthode de défuzzification la plus utilisée est de la détermination du centre de gravité de la fonction d'appartenance résultante. Cette méthode est la plus utilisée car elle est caractérisée par un simple calcul et très rapide.

IV.3.1.2. Organigramme de simulation

L'organigramme de simulation est représenté dans la figure (IV.7) ci-dessous

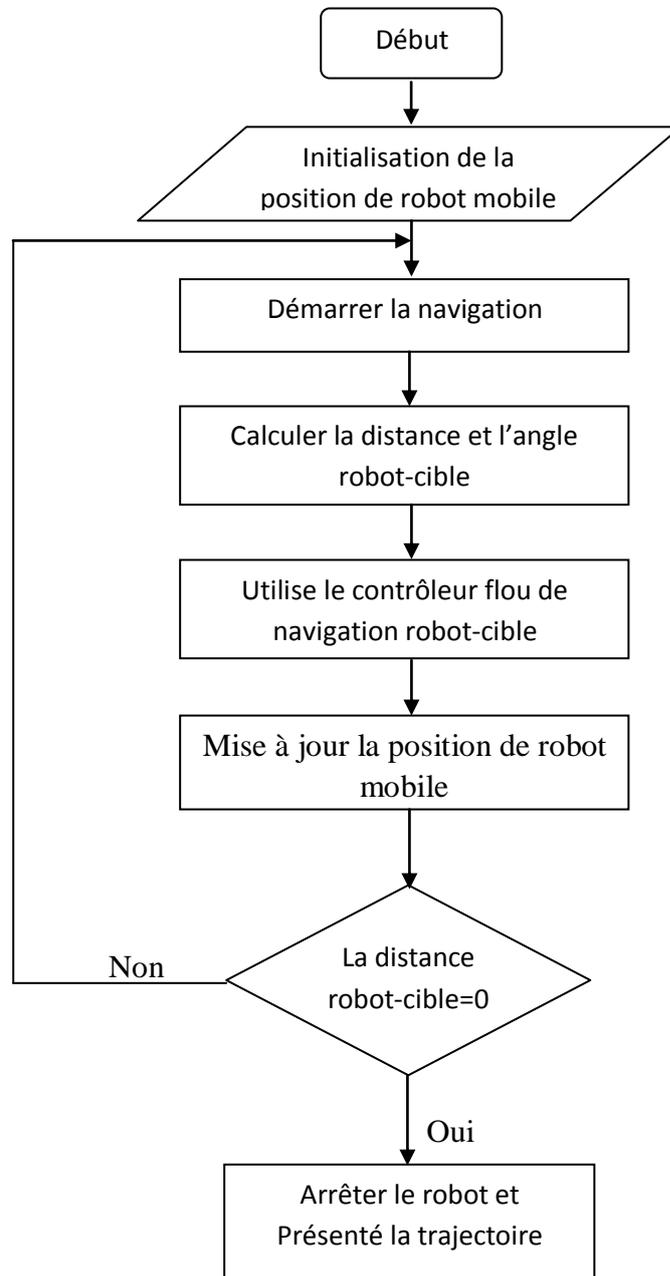


Figure IV.7 L'organigramme de navigation robot-cible.

IV.3.1.3. Résultats de simulation

La position initiale de robot mobile est : $(x, y, \theta) = (0, 0, 0)$

Résultat 1(R1)

Le cible est le point $(x,y)=(10,10)$:

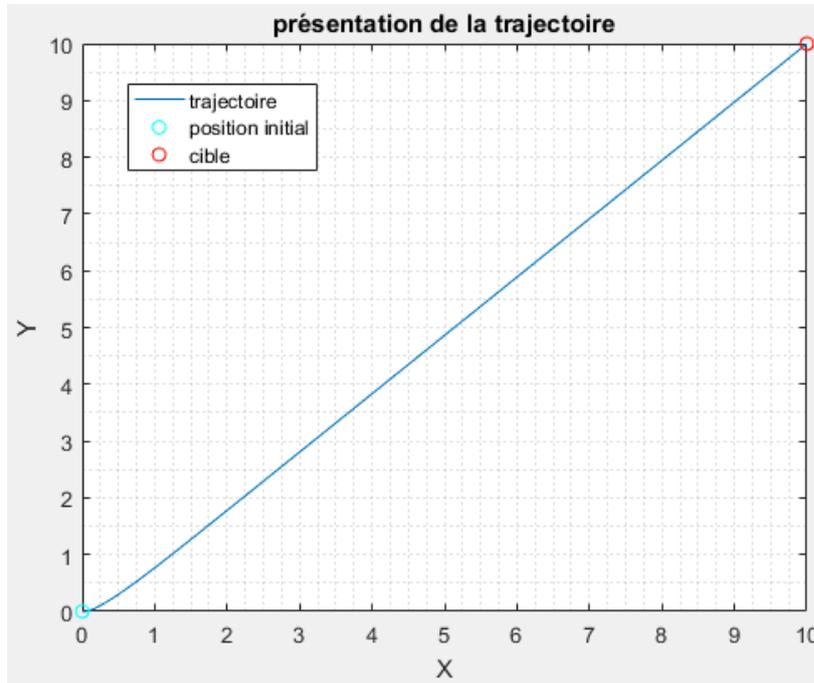


Figure IV.8 Navigation robot-cible R1 (Mamdani).

Au début le robot mobile fait une orientation puis continu vers la cible.

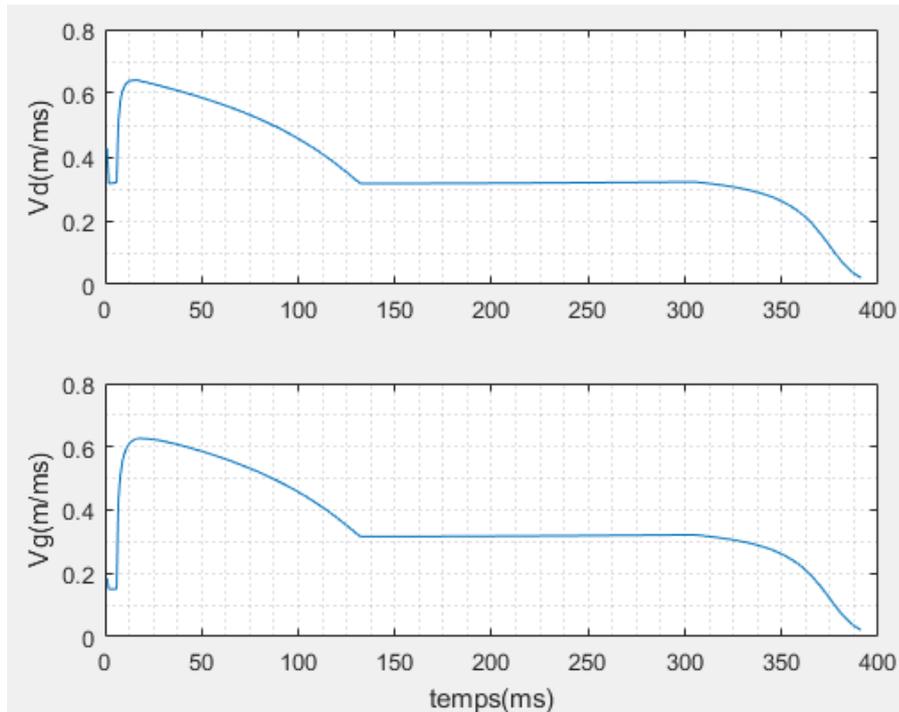


Figure IV.9 Variation de V_d et V_g de R1 de navigation robot-cible (Mamdani).

Résultat 2(R2)

Le cible est le point $(x,y)=(-4,8)$:

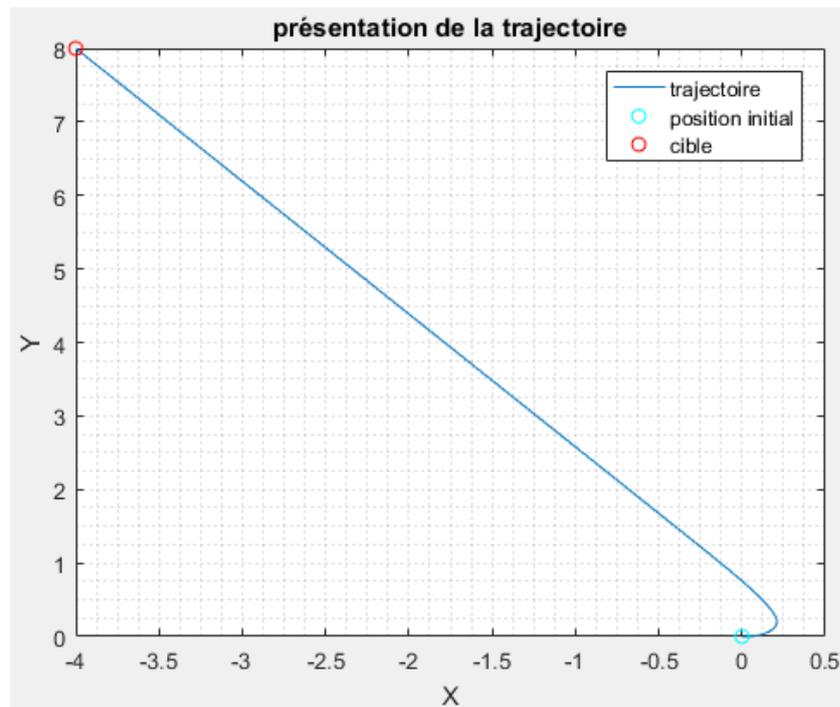


Figure IV.10 Navigation robot-cible R2 (Mamdani).

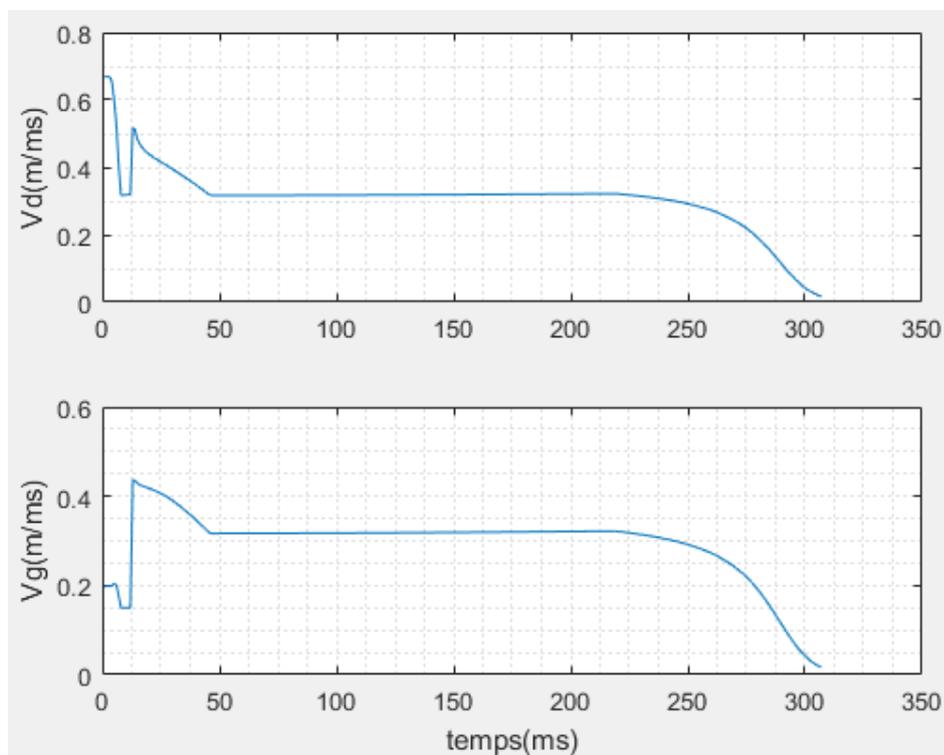


Figure IV.11 Variation de V_d et V_g de R2 de navigation robot-cible (Mamdani).

Au point de départ et pour faire l'orientation, la vitesse droite est à 77% et la vitesse gauche est de 20% comme l'indique la courbe dans la figure (IV.11).

IV.3.2. Comportements de navigation avec évitement d'obstacle

On utilise deux capteurs ultrason pour observer l'environnement de la navigation, Nous avons limité le champ global de perception du robot à 120° . La perception est décomposée dans deux secteurs de détection, chaque secteur Couvrant un angle de 60° à gauche ($\theta_r - 60^\circ$) et 60° à droite ($\theta_r + 60^\circ$) de l'angle de robot, pour obtenir une précision relative à la position des obstacles par rapport à notre robot.

Pour la navigation avec évitement d'obstacle on utilise deux contrôleurs flous, contrôleur de navigation robot-cible et contrôleur de navigation avec évitement d'obstacle, chaque contrôleur fonction selon la présence ou l'absence des obstacles, dans le cas de aucune obstacle le contrôleur de navigation robot-cible guide le robot vers la cible, et s'il y a d'obstacles le contrôleur de navigation avec évitement d'obstacle guide le robot pour éviter les obstacles.

IV.3.2.1. Structures de contrôleur flou de navigation avec évitement d'obstacle

Pour le comportement de navigation avec évitement d'obstacles on utilisé deux entrées: la distance entre le robot mobile et l'obstacle (D_{ro}), et L'angle entre l'orientation actuelle du robot et l'obstacle (θ_{ro}). Les sorties de contrôleur sont la vitesse de roue droite (V_d) et la vitesse de roue gauche (V_g) de robot mobile.

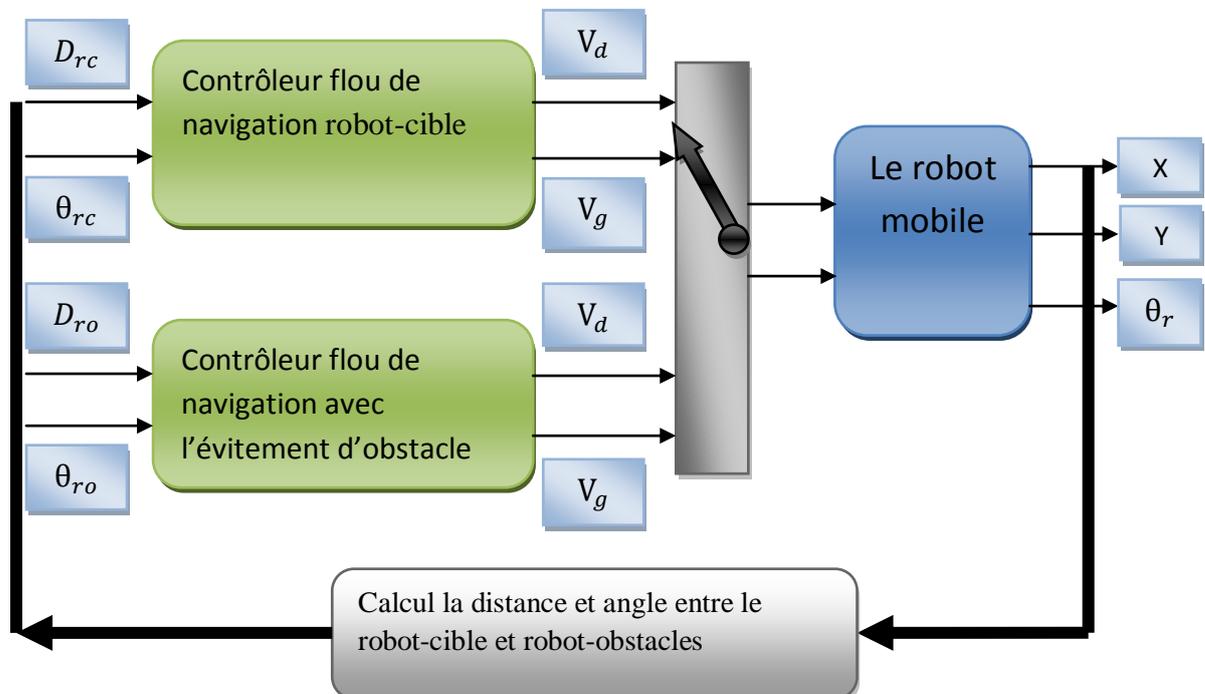


Figure IV.12 Structure du contrôleur de navigation avec évitement d'obstacle (Mamdani).

IV.3.2.1.1. Fuzzification

- Les valeurs de l'entrée, sont indiquées par les symboles linguistiques :
 - D_{r0} : Petit (**P**), moyen (**M**), grand (**G**).

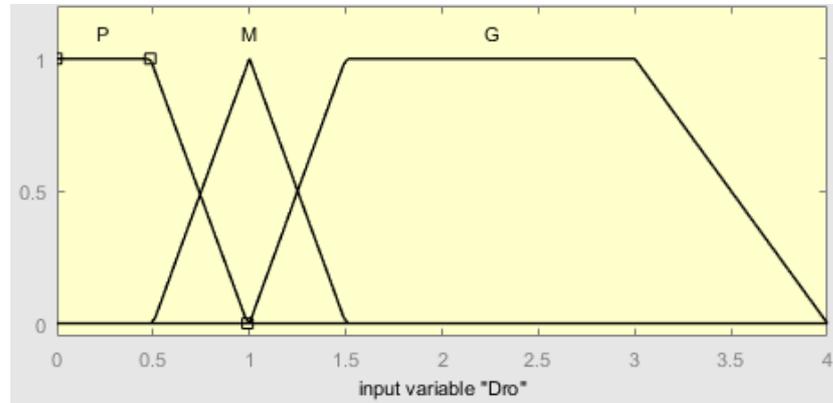


Figure IV.13 Fonctions d'appartenance de D_{r0} (Mamdani).

- θ_{r0} : grand négatif (**GN**), négatif (**N**), zéro (**Z**), positif (**P**), grand positif (**GP**).

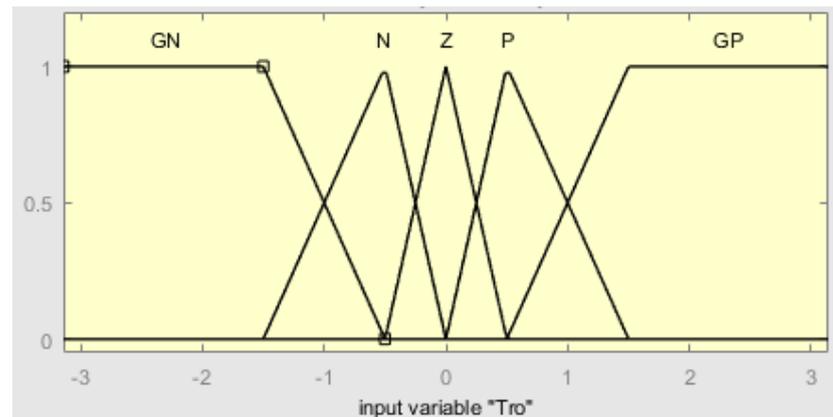


Figure IV.14 Fonctions d'appartenance de θ_{r0} (Mamdani).

- Les valeurs de sortie, sont indiquées par les symboles linguistiques :
 - V_d : zéro (**Z**), moyen (**M**), grand (**G**).

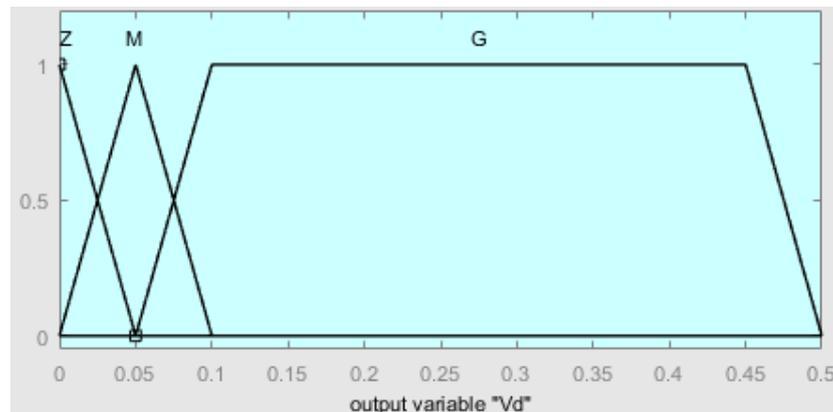


Figure IV.15 Fonctions d'appartenance de V_d cas d'obstacle (Mamdani).

- V_g : grand négatif (GN), négatif (N), zéro (Z), positif (P), grand positif (GP).

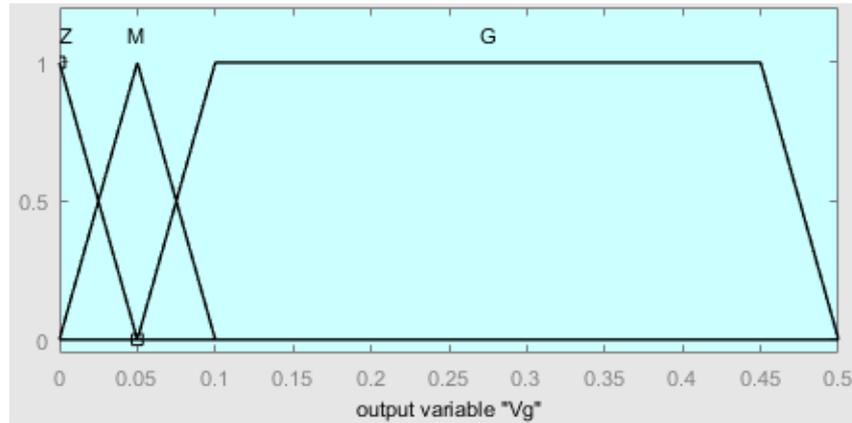


Figure IV.16 Fonctions d'appartenance de V_d cas d'obstacle (Mamdani).

IV.3.2.1.2. Inférence

Les règles floues utilisées pour ce comportement sont 15 règles, le tableau (IV.2) décrit ces règles :

θ_{ro} \ D_{ro}	GN		N		Z		P		GP	
	V_d	V_g								
P	Z	M	Z	G	G	Z	G	Z	M	Z
M	M	M	M	G	M	Z	G	M	M	M
G	G	G	M	M	G	M	M	M	G	G

Tableau IV.2 Les règles d'Inférence du contrôleur de navigation avec évitement d'obstacle (Mamdani).

IV.3.2.2. Organigramme de simulation

L'organigramme de simulation est représenté dans la figure (IV.17) ci-dessous :

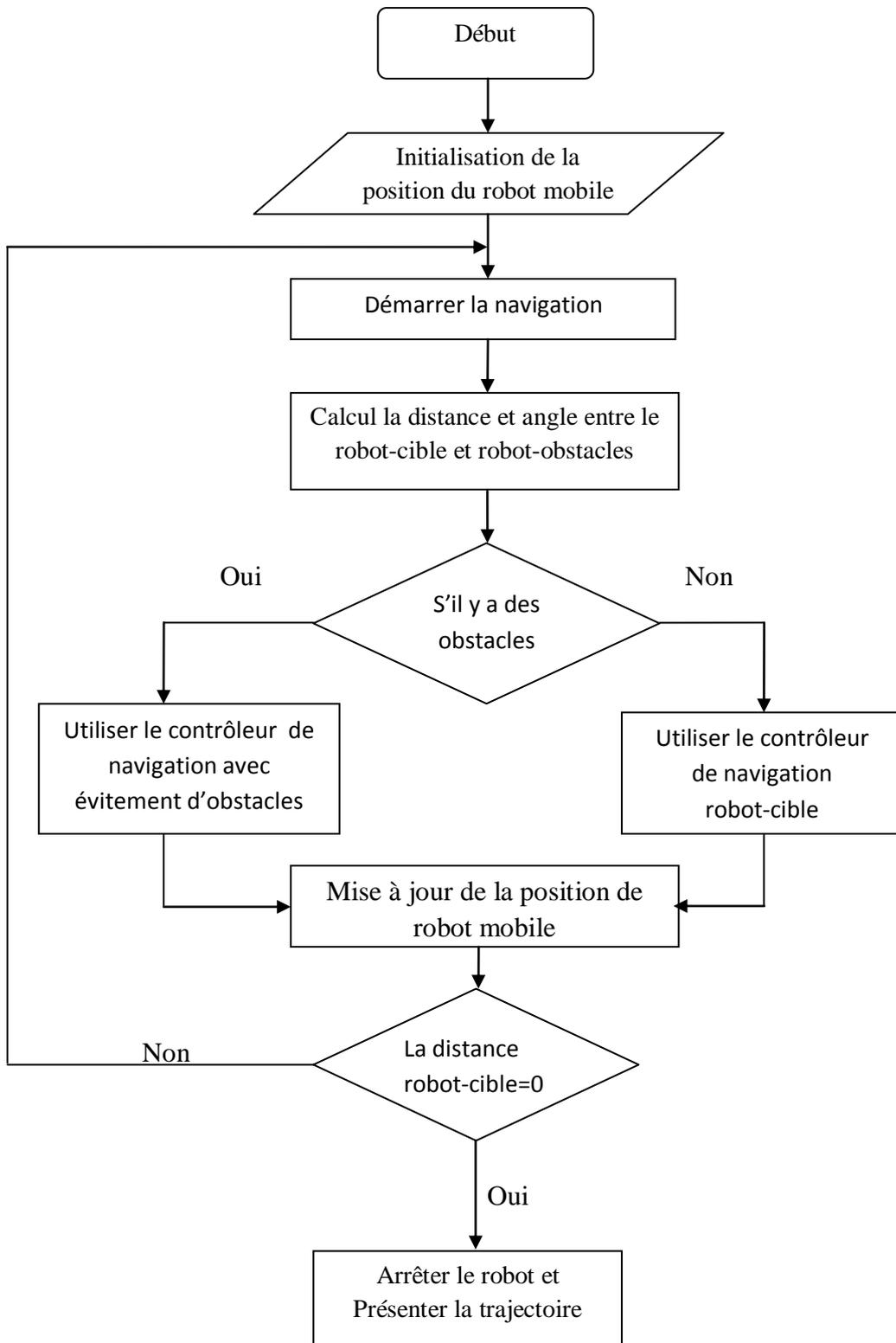


Figure IV.17 L'organigramme de navigation avec évitement d'obstacle.

IV.3.2.3. Résultats de simulation

La position initiale de robot mobile est : $(x, y, \theta) = (0, 0, 0)$.

Résultat 1(R1) cas d'un seul obstacle.

Le point cible est $(x, y) = (10, 10)$.

Les coordonnées de l'obstacle $(x, y) = (5, 5)$.

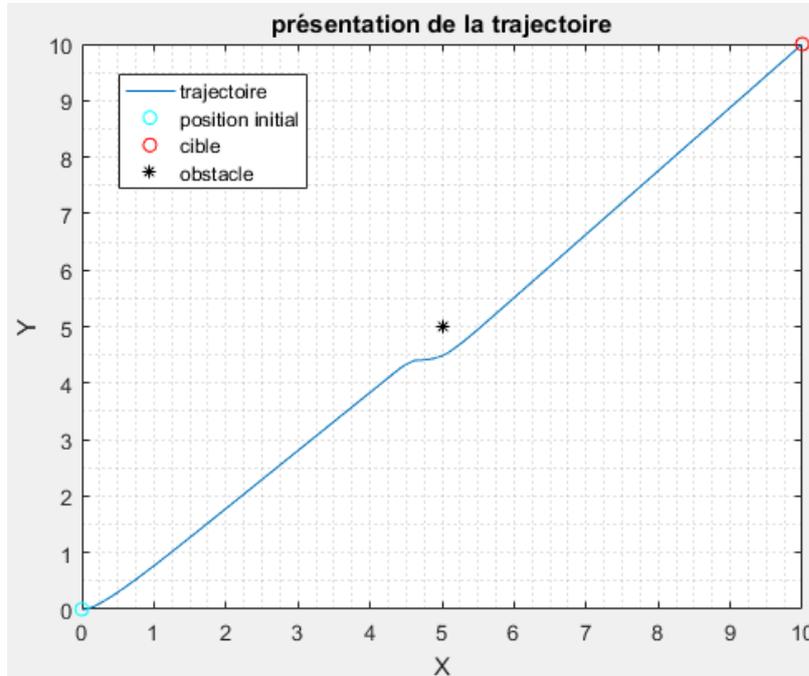


Figure IV.18 Navigation avec évitement d'obstacle R1 (Mamdani).

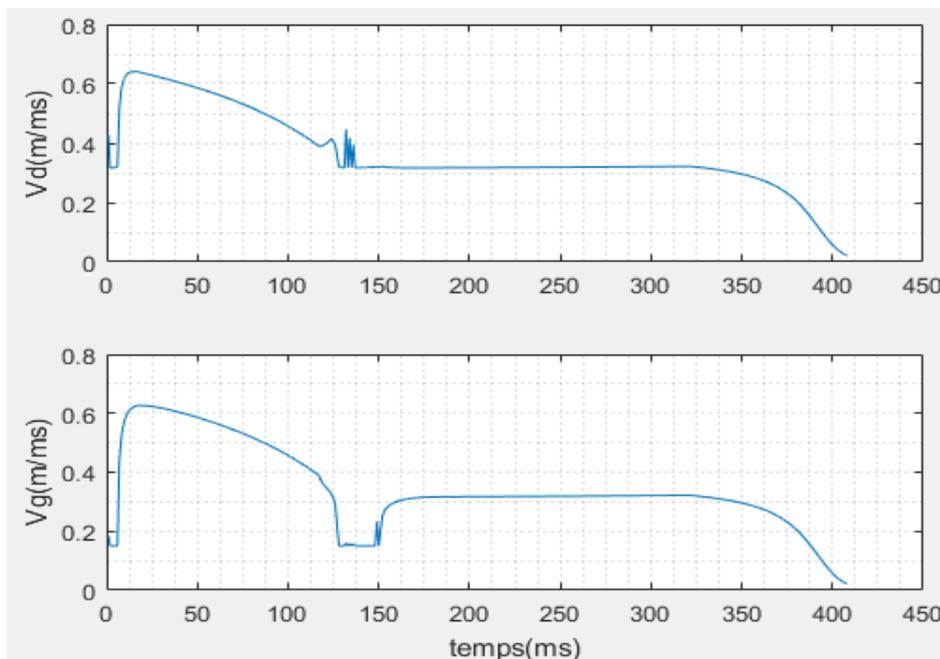


Figure IV.19 Variation de V_d et V_g de R1 de navigation avec évitement d'obstacle (Mamdani).

Le robot faire une orientation vers la droite pour évité l'obstacle puis continu vers la cible

Résultat 2(R2) cas de 2 obstacles.

Le point cible est $(x, y) = (10, 10)$.

Les coordonnées des obstacles (x, y) :

Obstacle 1 = $(5, 5)$.

Obstacle 2 = $(6, 6)$.

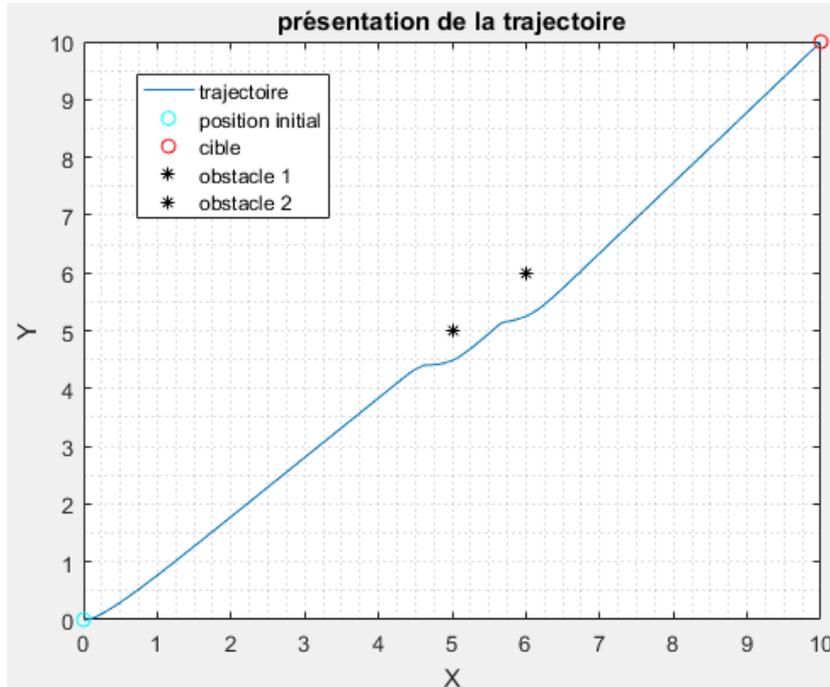


Figure IV.20 Navigation avec évitement d'obstacle R1 (Mamdani).

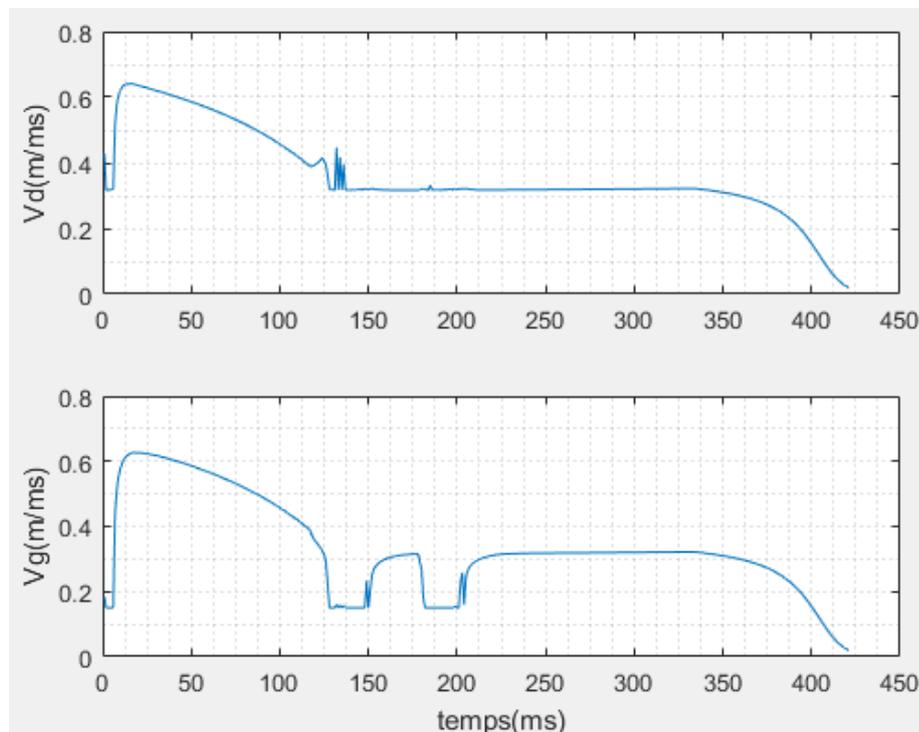


Figure IV.21 Variation de V_d et V_g de R2 de navigation avec évitement d'obstacle (Mamdani).

Résultat 3(R3) cas de 5 obstacles.

Le point cible est $(x, y) = (-8, 5)$.

Les coordonnées des les obstacles (x, y) :

Obstacles 1, 2, 3, 4, 5 = $(-2, 1.5), (-2.5, 2.5), (-4, 2.5), (-6, 3), (-6,4)$.

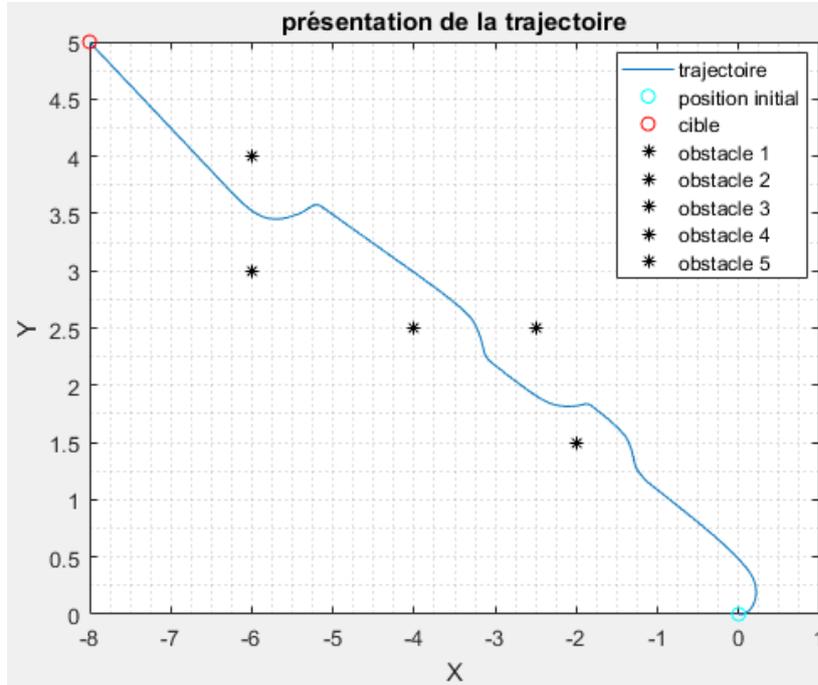


Figure IV.22 Navigation avec évitement d'obstacle R3 (Mamdani).

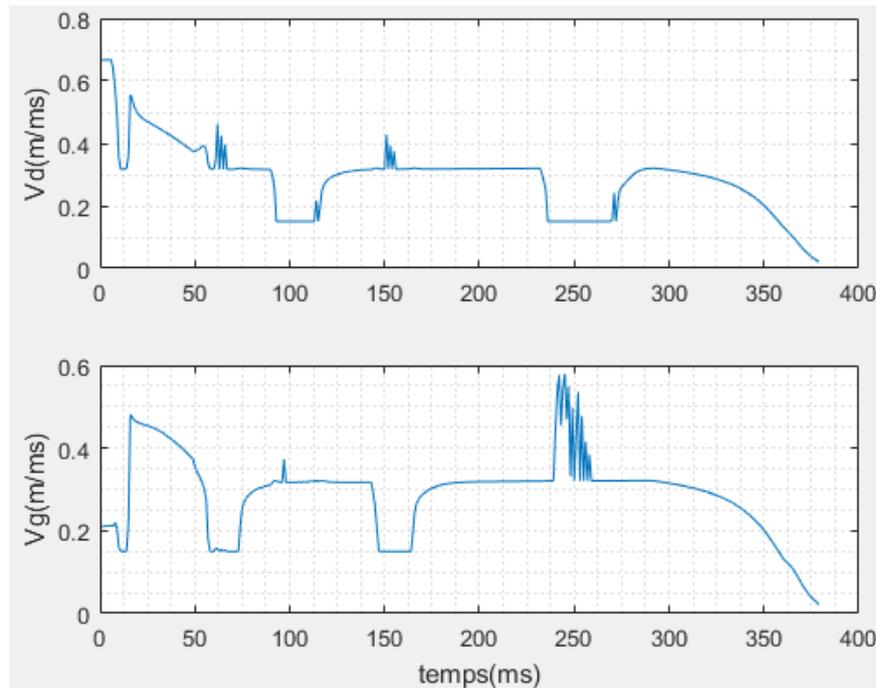


Figure IV.23 Variation de V_d et V_g de R3 de navigation avec évitement d'obstacle (Mamdani).

IV.4. Méthode de Sugeno

IV.4.1. Structures de base de notre contrôleur flou

Dans la commande floue par méthode Sugeno on utilise les deux contrôleurs de navigation, le contrôle de navigation robot-cible et de navigation avec évitement d'obstacle.

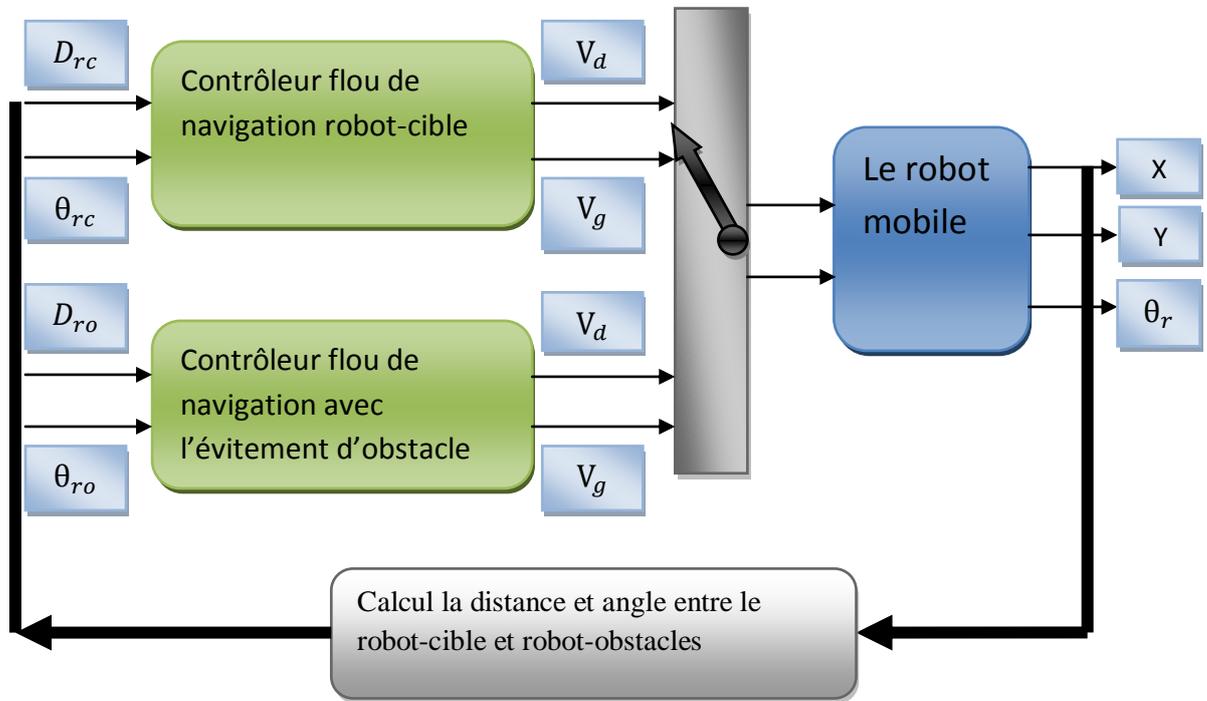


Figure IV.24 La structure de notre contrôleur flou (FLC) (Sugeno).

IV.4.1.1. Fuzzification

La fuzzification pour les deux contrôleurs c'est le même que la commande par méthode Mamdani pour les entrees et défirent pour les sorties. Les sorties dans la commande par méthode Sugeno c'est des valeurs constant.

- Les sorties de contrôleur de navigation robot-cible.
 - La vitesse de la roue droite V_d :

	Z	F	M	R	TR
$V_d(m/ms)$	0	0.2	0.3	0.5	0.7

Tableau IV.3 Les valeurs de V_d de contrôleur de navigation robot-cible (Sugeno).

- La vitesse de la roue gauche V_g :

	Z	F	M	R	TR
$V_g(m/ms)$	0	0.2	0.3	0.5	0.7

Tableau IV.4 Les valeurs de V_g de contrôleur de navigation robot-cible (Sugeno).

- Les sorties de contrôleur de navigation avec évitement d’obstacle.

- La vitesse de la roue droite V_d :

	Z	F	M	R	TR
$V_d(m/ms)$	0	0.3	0.4	0.5	0.7

Tableau IV.5 Les valeurs de V_d de contrôleur de navigation avec évitement d’obstacle (Sugeno).

- La vitesse de la roue gauche V_g :

	Z	F	M	R	TR
$V_d(m/ms)$	0	0.3	0.4	0.5	0.7

Tableau IV.6 Les valeurs de V_g de contrôleur de navigation avec évitement d’obstacle (Sugeno).

IV.4.1.2. Inférence

- Les règles d’inférence du contrôleur de navigation robot-cible.

θ_{rc} \ D_{rc}	GN	N		Z		P		GP		
	V_d	V_g								
P	R	Z	M	Z	F	F	Z	M	Z	R
M	R	F	M	F	M	M	F	M	F	R
G	R	M	R	M	TR	TR	M	R	M	R

Tableau IV.7 Les règles d’Inférence du contrôleur de navigation robot-cible (Sugeno).

➤ Les règles d'Inférence du contrôleur de navigation avec évitement d'obstacles.

D_{ro} \ θ_{ro}	GN		N		Z		P		GP	
	V_d	V_g								
P	F	M	F	TR	F	F	TR	F	M	F
M	R	R	F	R	M	F	R	F	R	R
G	TR	TR	R	R	R	R	R	R	TR	TR

Table IV.8 Les règles d'Inférence du contrôleur de navigation avec évitement d'obstacle (Sugeno).

L'organigramme de simulation de la commande de méthode Sugeno est le même que l'organigramme de la commande de méthode Mamdani dans les deux comportements de navigation.

IV.4.2. Résultats de simulation

La position initiale de robot mobile est : $(x, y, \theta) = (0, 0, 0)$.

Résultat 1(R1)

Le point cible est $(x, y) = (10, 10)$.

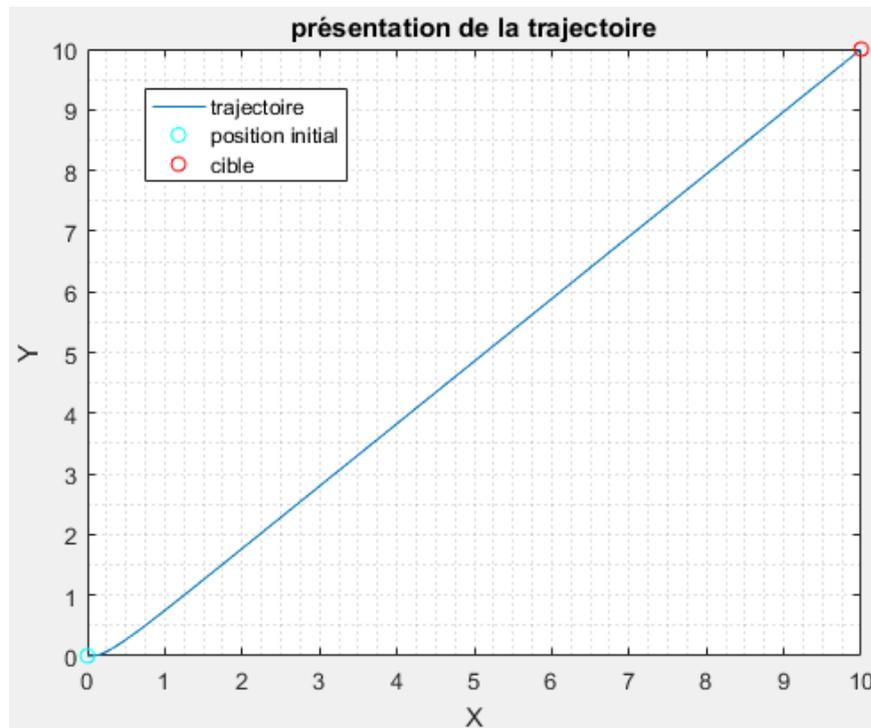


Figure IV.25 Navigation robot-cible R1 (Sugeno).

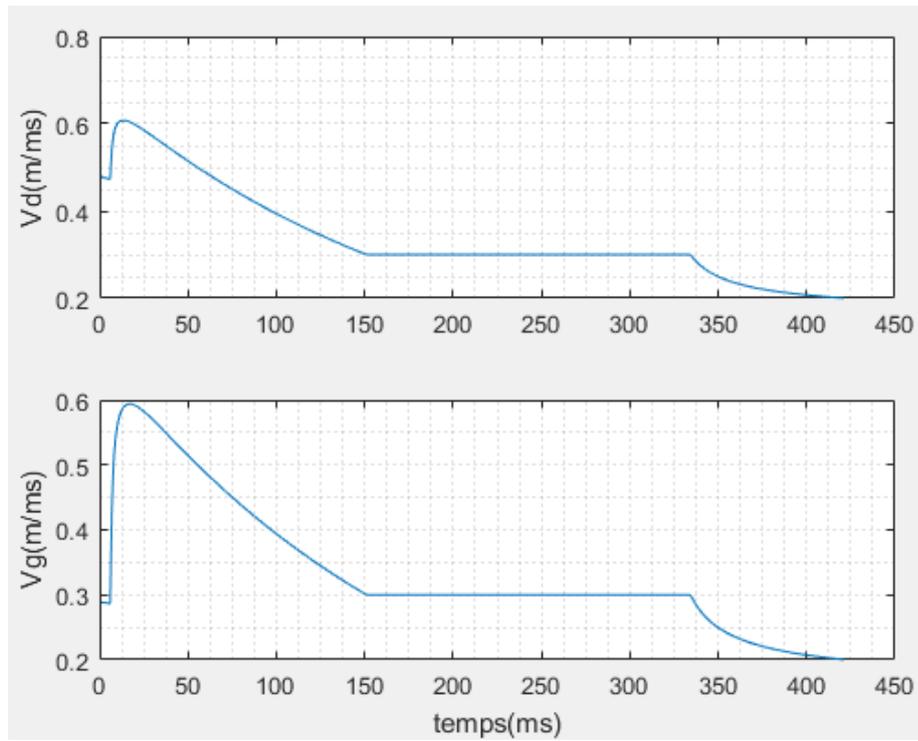


Figure IV.26 Variation de V_d et V_g de R1 de navigation robot-cible(Sugeno).

Résultat 2(R2) cas d'un seul obstacle.

Le point cible est $(x, y) = (10, 10)$.

Les coordonnées de l'obstacle $(x, y) = (5, 5)$.

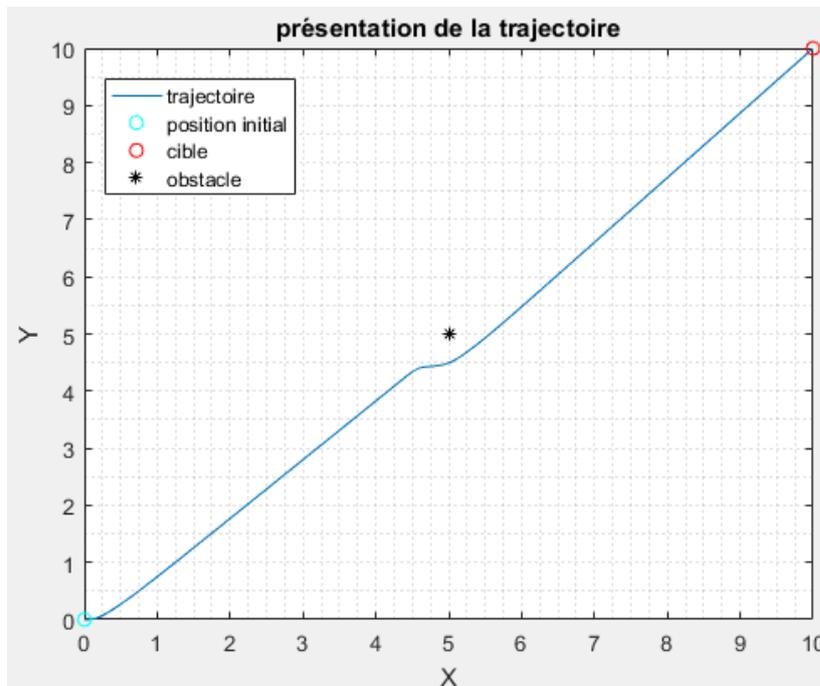


Figure IV.27 Navigation avec évitement d'obstacle R2 (Sugeno).

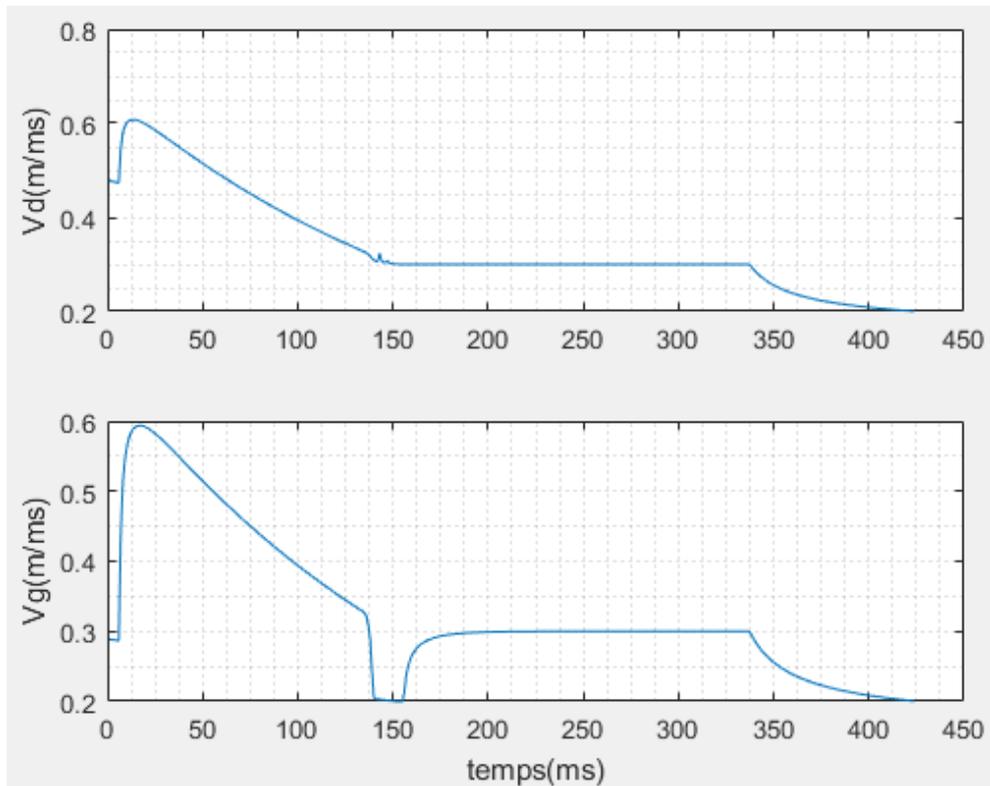


Figure IV.28 Variation de V_d et V_g de R2 de navigation avec évitement d'obstacle (Sugeno).

Résultat 3(R3) cas de 4 obstacles.

Le point cible est $(x, y) = (-5, 8)$.

Les coordonnées de l'obstacle (x, y) :

Obstacles 1, 2, 3, 4 = $(-2, 1.5), (-2.5, 2.5), (-4, 2.5), (-6, 3), (-6,4)$.

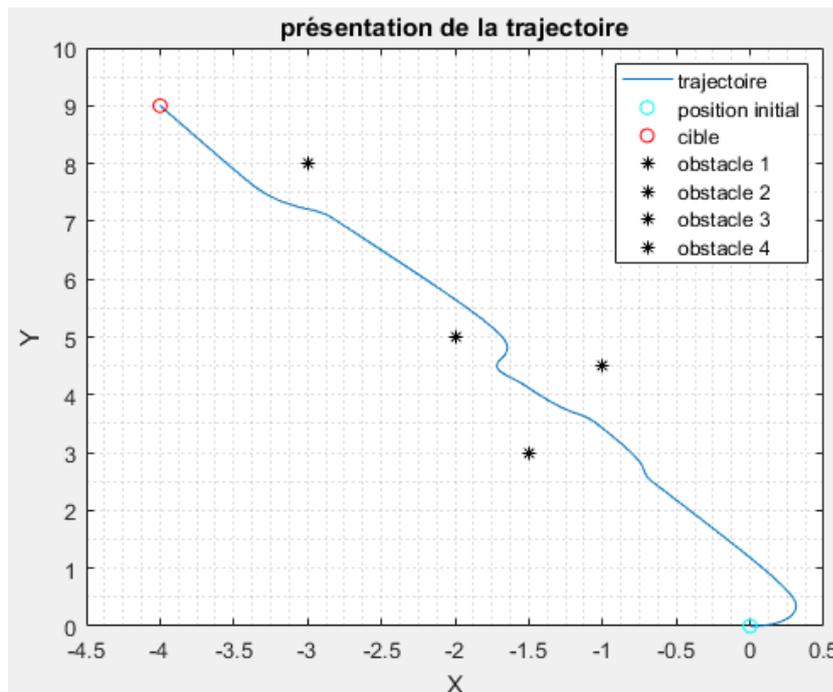


Figure IV.29 Navigation avec évitement d'obstacle R3 (Sugeno).

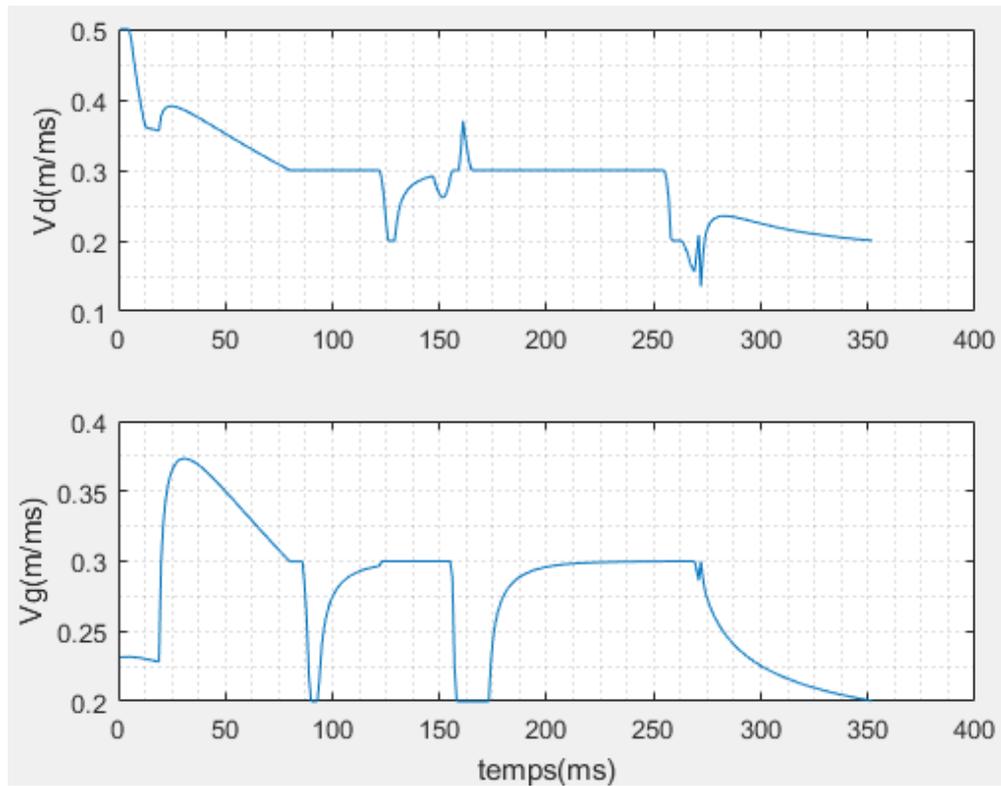


Figure IV.30 Variation de V_d et V_g de R3 de navigation avec évitement d'obstacle (Sugeno).

IV.5. Comparaison entre les deux méthodes

Le point cible est $(x, y) = (10, 10)$.

Les coordonnées de l'obstacle (x, y) :

Obstacles 1, 2, 3, 4, 5 = $(4, 4)$, $(5, 4.5)$, $(6.5, 5.5)$, $(7, 6)$, $(7.5, 7)$.

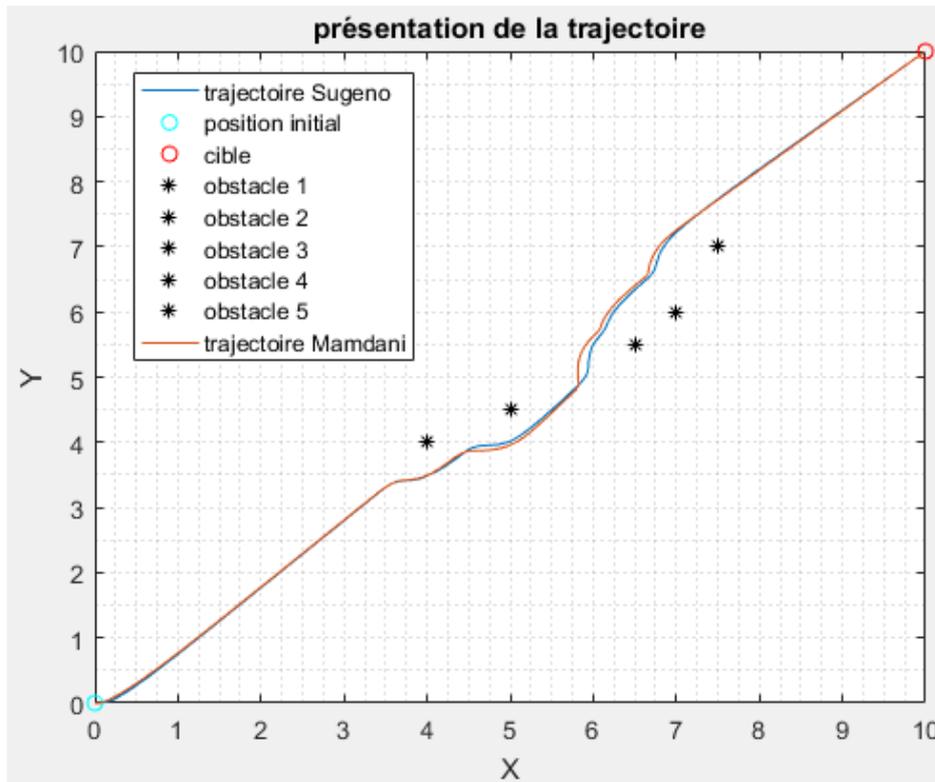


Figure IV.31 Comparaison entre la trajectoire de méthode Sugeno et Mamdani.

Remarque 1_ On remarque que la trajectoire de la méthode Mamdani et Sugeno est le même presque, mais la méthode Sugeno précise dans évitement des obstacles.

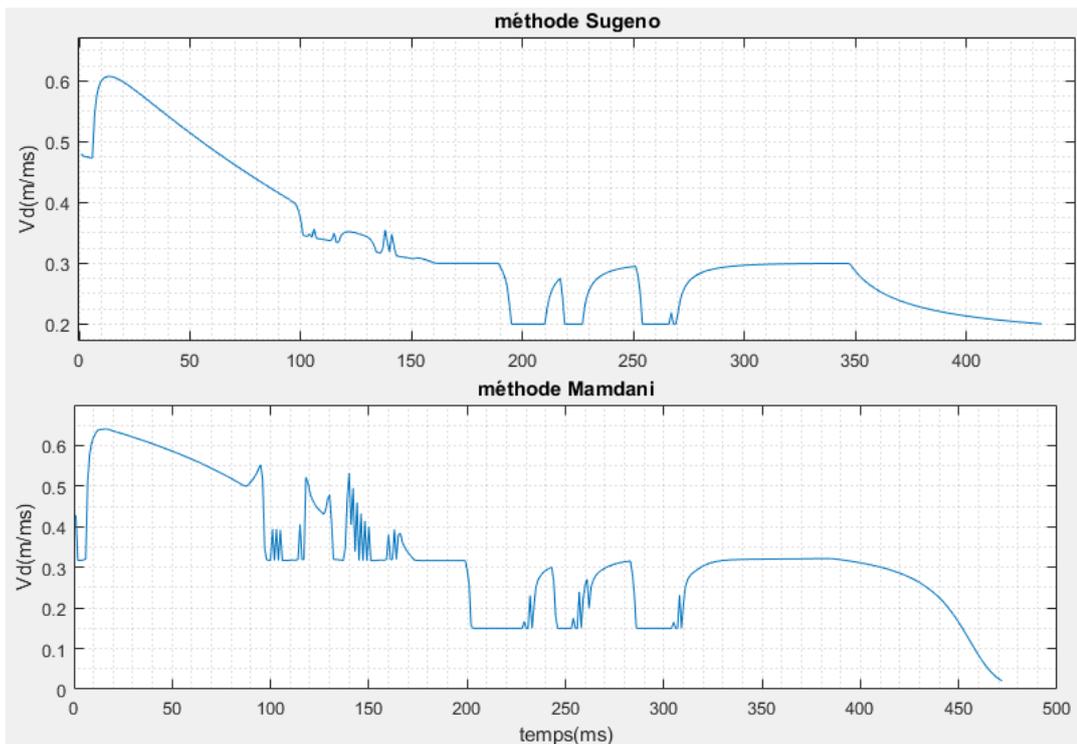


Figure IV.32 Comparaison entre la vitesse V_d de méthode Sugeno et Mamdani.

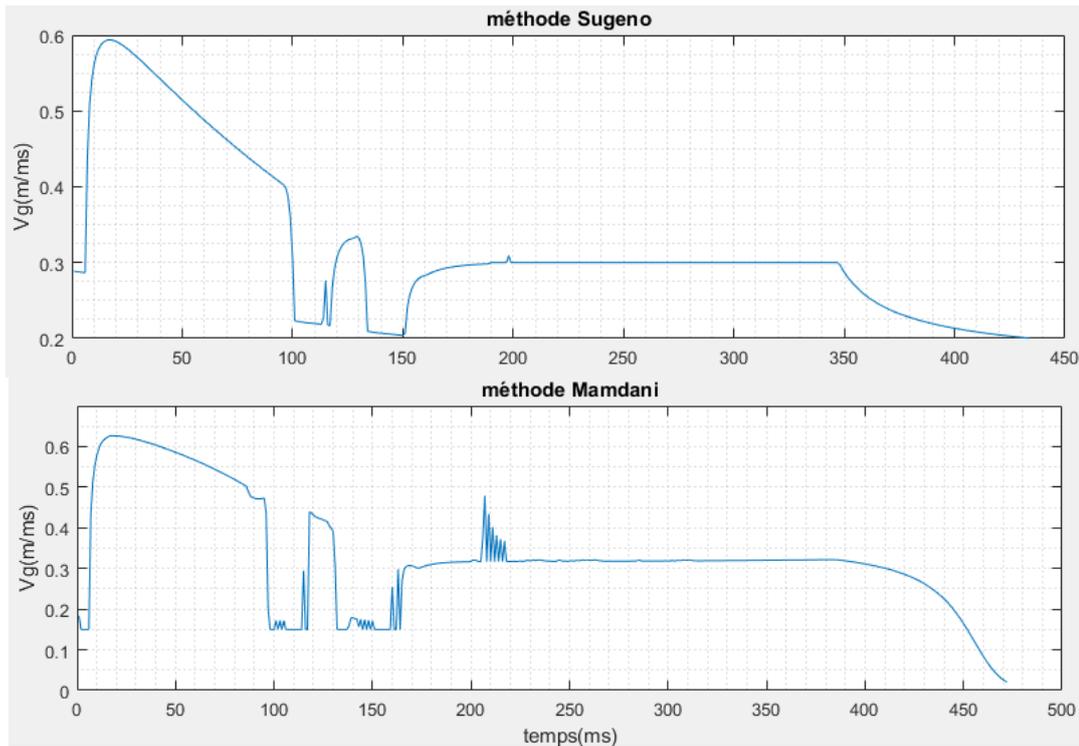


Figure IV.33 Comparaison entre la vitesse V_g de méthode Sugeno et Mamdani.

Remarque 2_ la vitesse de roue droite et gauche est perturbé dans la méthode Mamdani surtout dans le cas d'évitant les obstacles, par contre dans la méthode Sugeno elle est stable ce qui rend cette méthode meilleure que l'autre.

IV.6. Conclusion

Dans ce chapitre on présente notre travaille, où nous avons utilisé deux méthodes de la logique floue Mamadani et Sugeno pour commander un robot mobile de type uni-cycle, nous avons fait un contrôleur intelligent pour une bonne navigation telle que le déplacement de notre robot de la position initial vers une position final avec évitement de les obstacles dans son environnement avec le minimum des capteurs.

Nous présentant les résultats de la simulation de notre système floue en utilisant l'environnement Matlab, puis on comparant les deux méthodes Mamdani et Sugeno, nous avons remarqué que la méthode de Sugeno est plus précise que la méthode de Mamdani.

Conclusion Générale

Conclusion générale :

Dans ce mémoire notre objectif réside dans l'application de la logique floue dans la robotique mobile pour commander un robot mobile de type uni-cycle, pour faire un bon contrôle de navigation.

Nous avons commencé notre travail par des définitions et des concepts généraux sur les robot mobile, on parlé à des choses les plus importantes pour ce domaine, telle que l'autonomie du robot mobile, Les types des robots mobiles, la navigation, la planification, la localisation, et les architectures de contrôles utilisé dans la navigation d'un robot mobile, puis on parle sur la modélisation de robot mobile utilisé sur notre travaille et son étude cinématique et dynamique.

Nous avons fait une présentation des concepts de base et la théorie de la logique floue, telle que la Théorie des ensembles flous, variables linguistiques, fonctions d'appartenance, caractéristiques d'un ensemble flou, opérations sur les ensembles flous, et les deux méthodes de commande Mamdani et Sugeno.

Nous avons axé notre travail à établir une critère de stabilité conduisant à un contrôle simple et efficace d'un robot mobile, puis on a fait une comparaison entre les résultats de simulation obtenus pour les deux méthode de commande; Les résultats de simulation ont montré clairement l'efficacité de la logique floue dans la commande des robots mobiles, a partir des résultats de simulation obtenus Nous avons remarqué que la méthode Sugeno c'est la meilleur méthode qui permet au robot d'exécuter une bonne navigation dans son environnement que la méthode de Mamdani.

Bibliographie

- [1] **Boufera Fatma**, ‘Contribution des outils de l’intelligence artificielle dans la robotique mobile’, Thèse de Doctorat, Université d’Oran, 2014.
- [2] **Cherroun Lakhmissi**, ‘Navigation Autonome d’un Robot Mobile par des Techniques Neuro-Floues’, Thèse de Doctorat, Université de Biskra, 2014.
- [3] **Salah Kermiche**, ‘Modélisation et commande d’un robot par méthodes intelligentes’, Thèse de Doctorat, université badji mokhtar-annaba, 2006.
- [4] <http://marsrovers.jpl.nasa.gov/gallery/artwork/hires/rover3.jpg>.
- [5] **Alexandre Lampe**, ‘Méthodologie d’évaluation du degré d’autonomie d’un robot mobile terrestre’, Automatique/Robotique, Institut National Polytechnique de Toulouse - INPT, 2006, Français.
- [6] **Slimane Noureddine**, ‘système de localisation pour robots mobiles’, Thèse de Doctorat, universite de batna, 2005.
- [7] **Ammar Abdelkader et Serdoun Belkaceme**, ‘Elaboration d’un contrôleur flou pour la navigation d’un robot mobile’, mémoire de Master, Université Djilali Bounaama-Khemis Miliana-algeria, 2016.
- [8] **David Filliat**, ‘Robotique Mobile’, Cours option robotique, École Nationale Supérieure de Techniques Avancées ParisTech, 2016.
- [9] **Éric Beaudry**, Mémoire, ‘Planification de tâches pour un robot mobile autonome’, université de sherbrooke, Canada, 2006.
- [10] **A. Oualid Djekoune**. ‘Localisation Et Guidage Du Robot Mobile Atrv2 Dans Un Environnement Naturel’, Robotique [cs.RO], Université des Sciences et de la Technologie Houari Boumediene, 2010.
- [11] **Olivier Lefebvre**, ‘Navigation autonome sans collision pour robots mobiles non holonomes’, Thèse de Doctorat, l’Institut National Polytechnique de Toulouse, 2006.
- [12] **Houcine Zerfa**, ‘Conception, Réalisation et Commande Floue d’un Robot Mobile’, mémoire magister, Université des Sciences et de la Technologie d’Oran, 2013.
- [13] **B. Bayle**, ‘Robotique mobile’, Cours option ISAV, Télécom Physique Strasbourg 3A, Université de Strasbourg.
- [14] **Zidani Ghania**, ‘Commande Robuste d’un Robot Mobile à Roues’, Thèse de Doctorat, Université de Batna -2-, 2017.
- [15] **F. Morbidi**, ‘perception avancée et robotique mobile’, Cours option robotique, Université de picardie, 2015.

- [16] **Hajer Omrane et al**, 'Fuzzy Logic Based Control for Autonomous Mobile Robot Navigation', <http://dx.doi.org/10.1155/2016/9548482>.
- [17] **Benmechri Ratima**, 'contrôle d'un système flou de type 2 applique a un robot mobile', mémoire magister, université Ferhat Abbas-Setif, 2011.
- [18] **G. Chen et Trung Tat Pham** (2001), 'Introduction to fuzzy sets, fuzzy logic, and fuzzy control systems', Texas, 316 p.
- [19] **V. Peri et D. Simon**, (2005), 'Fuzzy Logic Control for an Autonomous Mobile Robot', North American Fuzzy Information Processing Society Conference, 337-342.