

Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur Et de La Recherche Scientifique



Université de Ghardaïa

Faculté des Sciences et Technologies

Département d'Automatique et Électromécanique

N° d'ordre :

N° de série :

Mémoire présenté en vue de l'obtention du diplôme de

MASTER

Domaine : *Sciences et Technologies*

Filière : *Automatique*

Spécialité : *Automatique et systèmes*

Par : *BAHAZ Radouane*

BAHAZ Abdelbasset

Thème

**Navigation autonome et suivi de trajectoires d'un robot
mobile à entraînement différentiel**

Soutenu publiquement le : 21/10/2020

Devant le jury :

BAHRI Ahmed	MAA	Univ. Ghardaïa	Président
BECHOUAT Mohcene	MCA	Univ. Ghardaïa	Examineur
FIHA KHIR Amine Mehdi	MCB	Univ. Ghardaïa	Examineur
HACENE Nacer	MCB	Univ. Ghardaïa	Encadrant

Année universitaire 2019/2020

Dédicace:

Nous avons le grand plaisir de dédier ce modeste travail à:

 *Nos parents pour leur soutien pendant toutes nos années d'études..*

 *Toutes nos familles.*

 *Tous nos amis, camarades et collègues sans exception.*

Abdelbasset

Radouane

Remerciement:

Avant tous, Nous remercions en premier lieu notre Dieu de nous avoir donné la santé et la patience pour avoir terminé ce travail.

Nous tenons à exprimer notre sincère gratitude à notre superviseur Dr. HACENE Nacer pour son soutien et ses conseils durant ce travail. Ses encouragements et ses conseils ont toujours été une source de motivation. Ce fut un honneur de travailler sous sa supervision.

Nous remercions particulièrement nos professeurs, le personnel académique et le personnel de l'Université de Ghardaïa,

Plus important encore, nous sommes reconnaissants envers nos familles qui ont soutenu notre chaque fois que nous en avons besoin, toujours à nos côtés, guidant notre vie. Nous tenons à les remercier pour leur amour inconditionnel qui nous motive à viser plus haut.

ملخص:

تقترح هذه المذكرة طريقة لحل مشكلة تتبع المسارات من أجل الروبوت ذي القيادة التفاضلية. للقيام بذلك ، يستخدم الروبوت متحكماً غامضاً يمكّنه من تتبع المسار المطلوب.

أولاً ، تم إنشاء النموذج الحركي (سينماتيكي) والنموذج التحريكى (ديناميك) بطريقة مفصلة. ومن ثم بناء النموذجين في الفضاء المحاكاة ماتلاب (Matlab Simulink).

نظراً لأن ديناميك الروبوت يتمتع بدرجة عالية من اللاخطية، فقد تم تصميم وحدة تحكم غامضة للسماح للروبوت باتتباع المسار المطلوب. أظهرت نتائج المحاكاة كفاءة المتحكم المقترح في مشكلة تتبع المسارات، وقد تم دعم ذلك باستخدام مسارات مختلفة.

تم بناء نموذج أولي للروبوت. حيث يتم التحكم فيه بواسطة لوحة (Arduino). وهو مزودة بثلاثة مجسات فوق صوتية لتجنب العوائق.

الكلمات المفتاحية: روبوت متنقل ذو قيادة تفاضلية، ملاحه ذاتية، تتبع المسارات، المنطق الغامض، أردوينو

Résumé

Ce mémoire propose une approche pour résoudre le problème de suivi de trajectoire pour le robot à entraînement différentiel (Differential Drive). Pour ce faire, un contrôleur flou est utilisé pour permettre au robot de suivre la trajectoire souhaitée.

Tout d'abord, le robot a été modélisé, où la cinématique et une dynamique détaillée sont développées. La cinématique et la dynamique du robot ont été réalisées et testées à l'aide de Matlab Simulink.

Étant donné que la dynamique du robot présente un degré élevé de non-linéarité, un contrôleur flou a été conçu pour permettre au robot de suivre une trajectoire souhaitée. Les résultats de la simulation montrent l'efficacité du contrôleur proposé dans le problème de suivi de trajectoire, cela était justifié par l'utilisation de trajectoires différentes.

Un prototype du robot a été construit. Le robot est piloté par une carte Arduino. Et il est équipé de trois capteurs à ultrasons pour éviter les obstacles.

Mots clés: Robot mobile à entraînement différentiel, navigation autonome, suivi de trajectoire. Contrôle flou, Arduino.

Abstract

This thesis proposes an approach to solve the problem of trajectory tracking for the Differential Drive robot. To do this, a fuzzy controller is used to allow the robot to follow the desired trajectory.

First, the robot was modeled, where the kinematics and detailed dynamics are developed. The kinematics and dynamics of the robot have been built and tested using Matlab Simulink.

Since the robot dynamics have a high degree of non-linearity, a fuzzy controller has been designed to allow the robot to follow a desired path. The simulation results show the efficiency of the proposed controller in the problem of trajectory tracking ; this was demonstrated by the use of different trajectories.

A prototype of the robot has been designed. the robot is driven by an Arduino board. And it is equipped with three ultrasonic sensors to avoid obstacles.

Keywords: Differential drive mobile robot, autonomous navigation, trajectory tracking. Fuzzy control, Arduino.

Table des matières

Dédicace:	II
Remerciement:	III
Résumé.....	IV
Table des matières	VI
Liste de Figures.....	X
Liste de Tableaux.....	XIII
Liste des abréviations et symboles.....	XIV
Introduction Générale	1
Chapitre I.....	3
La navigation Autonome des Robots Mobiles: Généralités	3
I.1. Introduction:	3
I.2. La définition d'un robot mobile:	3
I.3. Classification des robots mobiles:	4
I.3.1. Robots mobiles à roues:.....	5
I.3.2. Robots mobiles à chenilles:	5
I.3.3. Robots mobiles à pattes:	5
I.3.4. Robots mobiles volants:.....	5
I.3.5. Robots mobiles sous-marins:.....	6
I.3.6. Robots mobiles hybrides:	6
I.4. L'autonomie d'un robot mobile:.....	6
I.5. Perception:	7
I.6. La navigation autonome d'un robot mobile	12
I.6.1. La navigation:	12
I.6.2. Localisation:	14
I.6.3. La cartographie (mapping)	15

I.7. Conclusion:	16
Chapitre II	18
Modélisation d'un robot à entraînement différentiel	18
II.1. Introduction:	18
II.2. Description du robot à entraînement différentiel:	18
II.3. La modélisation cinématique:	19
II.4. Le modèle de mouvement:	19
II.5. Le modèle cinématique:	21
II.6. La modélisation dynamique:	24
II.7. Simulation	28
▪ Réponse indicielle	30
▪ Réponse indicielle du modèle cinématique	30
▪ Réponse indicielle du modèle dynamique	32
▪ Réponse indicielle du modèle cinématique avec le modèle dynamique	33
II.8. Conclusion:	35
Chapitre III	37
Problème de suivi de trajectoire du robot à entraînement différentiel	37
III.1. Introduction:	37
III.2. Le système de suivi de trajectoires	37
III.3. Les systèmes flous	38
III.3.1. La fuzzification:	39
III.3.2. La base des règles:	39
III.3.3. Le mécanisme d'inférence:	39
III.3.4. La défuzzification:	40
III.4. Conception du contrôleur flou	40
III.4.1. La fuzzification	41
III.4.2. La base de règles floues	42

III.5. Résultats de simulation et discussion.....	44
• Trajectoire circulaire	46
• Trajectoire en forme de huit.....	47
• Trajectoire en forme de filet :.....	48
III.6. Conclusion :	51
Chapitre IV.....	53
Réalisation du robot différentiel.....	53
IV.1. Introduction	53
IV.2. Coté Hardware:.....	53
IV.2.1. La plate-forme:	53
IV.2.2. Les moteurs:.....	54
IV.2.3. Drive of motor (L298N):	55
IV.2.3.1. Contrôle la direction du moteur par H-bridge:	55
IV.2.3.2. Contrôle la vitesse du moteur par PWM:	56
IV.2.4. Les capteurs à ultrasons:.....	59
IV.2.4.1. Le principe de fonctionnement général:	59
IV.2.4.2. Le capteur HC-SR04:	60
IV.2.5. La carte Arduino Uno:	62
IV.2.5.1. Alimentation de la carte Arduino	63
IV.2.5.2. Les entrées et sorties:.....	64
IV.3. Coté Software:	65
IV.3.1. Arduino IDE:	65
IV.4. La réalisation et la programmation :.....	66
IV.4.1. La réalisation :	66
IV.4.2. L'organigramme d'évitement d'obstacle.....	68
IV.5. Implémentation et résultats:.....	69
IV.6. Conclusion :	70

Conclusion générale	72
Références.....	74
Annexe A - Diagramme schématique de l'Arduino Uno	77
Annexe B - Diagramme schématique du Robot	78

Liste de Figures

Chapitre I

Figure I.1. Le robot mobile "Sojourner" de la NASA :	4
Figure I.2. La classification des robots mobiles :	5
Figure I.3. Interaction entre le robot mobiles et l'environnement : a) schéma global, b) paradigme (Perception-Décision-Action)	7
Figure I.4. Exemple de capteurs utilisés par un robot mobile éducatif	8
Figure I.5. Capteurs ultrasons	8
Figure I.6. Capteurs infrarouges	9
Figure I.7. Capteur laser	9
Figure I.8. Les cartes utilisée en robotique mobile: (a) l'environnement réel, (b) carte métrique, (c) carte topologique	16

Chapitre II

Figure II.1. Configuration du robot.....	19
Figure II.2. Le repère global du robot	20
Figure II.3. Les forces sur les roues : une représentation équivalente.....	21
Figure II.4. Schéma des vitesses appliquées au robot	25
Figure II.5. La cinématique du robot sur SIMULINK	29
Figure II.6. La dynamique du robot sur SIMULINK	29
Figure II.7. Modèle de la réponse indicielle de la cinématique du robot sur SIMULINK	30
Figure II.8. La réponse indicielle avec $(x_{G0}, y_{G0}, \varphi_0) = (0m, 0m, 0^\circ)$ et $(V_R = 1 m/s$ et $V_L = 1 m/s)$: a) trajectoire, b) orientation	31
Figure II.9. La réponse indicielle avec $(x_{G0}, y_{G0}, \varphi_0) = (0m, 0m, 60^\circ)$ et $(V_R = 1 m/s$ et $V_L = 1 m/s)$: a) trajectoire, b) orientation	31

Figure II.10. La réponse indicielle avec $(x_{G0}, y_{G0}, \varphi_0) = (0m, 0m, 60^\circ)$ et $(V_R = 1 m/s$ et $V_L = 0.5 m/s)$: a) trajectoire, b) orinetaion	32
Figure II.11. Modèle de la réponse indicielle de la dynamique du robot sur SIMULINK	32
Figure II.12. La réponse indicielle ($E_R = 5 V$ et $E_L = 5 V$)	33
Figure II.13. La réponse indicielle ($E_R = -2V$ et $E_L = 3V$)	33
Figure II.14. Modèle de la réponse indicielle de la dynamique avec la cinématique du robot sur SIMULINK	34
Figure II.15. La réponse indicielle ($E_R = 5V$ et $E_L = 5V$)	34
Figure II.16. La réponse indicielle ($E_R = 5 V$ et $E_L = 2V$)	35

Chapitre III

Figure III.1. Schéma de base du système de suivi de trajectoires	38
Figure III.2. Structure générale d'un contrôleur flou	38
Figure III.3. Le problème de suivi de trajectoire	40
Figure III.4. Architecture du contrôleur flou proposé	41
Figure III.5. Les fonctions d'appartenance des entrées Dis (la distance) et β (l'angle).....	42
Figure III.6. Les fonctions d'appartenance des sorties V_R et V_L	42
Figure III.7. Système générale de contrôle pour le problème du suivi de trajectoires	45
Figure III.8. Suivre une trajectoire circulaire	46
Figure III.9. Suivre une trajectoire circulaire, l'erreur sur la: a) distance (Dis), b) orientation	46
Figure III.10. La commande appliqué pour la trajectoire circulaire : a) Tension du moteur droit E_R , b) Tensien du moteur E_L	47
Figure III.11. Suivre une trajectoire en forme de huit	47
Figure III.12. L'erreur du suivi de trajectoire en forme de huit : a) Distance, b) Bearing :	48

Figure III.13. La commande appliquée pour la trajectoire en forme de huit: a) Tension du moteur droit E_R , b) Tension du moteur gauche E_L	48
Figure III.14. Suivre une trajectoire en forme de filet	49
Figure III.15. L'erreur du suivi d'une trajectoire en forme de filet: a) Distance, b) Bearing	49
Figure III.16. La commande appliquée sur le robot : a) Tension du moteur droit E_R , b) Tension du moteur gauche E_L	50

Chapitre IV

Figure IV.1. La plate-forme du robot à entraînement différentiel :	54
Figure IV.2. Moteur à courant continu avec Gearbox :	55
Figure IV.3. Contrôle du moteur à courant continue par H-bridge :	56
Figure IV.4. PWM (Pulse Width Modulation):	57
Figure IV.5. Les bornes du L298N :	58
Figure IV.6. Fréquence des ultrasons et humaine :	60
Figure IV.7. Le capteur HC-SR04 :	60
Figure IV.8. Le principe de fonctionnement du capteur HC-SR04 :	61
Figure IV.9. La carte Arduino Uno Rev3 :	64
Figure IV.10. Interface logicielle Arduino IDE :	66
Figure IV.11. L'installation d'un circuit électrique entre les éléments électroniques d'un robot mobile pour éviter les obstacles :	67
Figure IV.12 : L'organigramme d'évitement d'obstacles appliqué dans le projet	68
Figure IV.13. Robot mobile à entraînement différentiel d'évitement d'obstacles équipé de trois capteurs à ultrason :	69
Figure IV.14. Test de robot mobile pour éviter les collisions avec des obstacles :	69

Liste de Tableaux

Table II.1. Les paramètres du robot :	30
Tableau III.1. Les variables linguistiques pour les entrées et les sorties	42
Tableau III.2. Règles floues du contrôleur proposé	43
Tableau IV.1. Les caractéristiques du moteur	54
Tableau IV.2. Les spécifications du capteur à ultrason HC-SR04	62
Tableau IV.3. Les spécifications de la carte Arduino Uno Rev3	65

Liste des abréviations et symboles

AUAV: Autonomous Unmanned Aerial Vehicle

NASA: National Aeronautics and Space Administration

GPS: Global Positioning Systems

DDMR: Differential Drive Mobile Robot

ICC: Instantaneous Center of Curvature

MIMO: Multi Input Multi Output

FLC: Fuzzy Logic controller

FAM: Fuzzy Associative Memory

E/S: Entrée (Input) / Sortie (Output)

AC-DC: Direct Current-Alternating-Current

ICSP: In-Circuit Serial Programmer

MOSFET: Metal–Oxide–Semiconductor Field-Effect Transistor

PWM: Pulse Width Modulation

RX/TX: Receive pin/Transmit pin

SPI: Serial Peripheral Interface

USB: Universal Serial Bus

EEPROM: Electrically Erasable Programmable Read-Only Memory

SRAM: Static Random Access Memory

MISO: Master In Slave Out

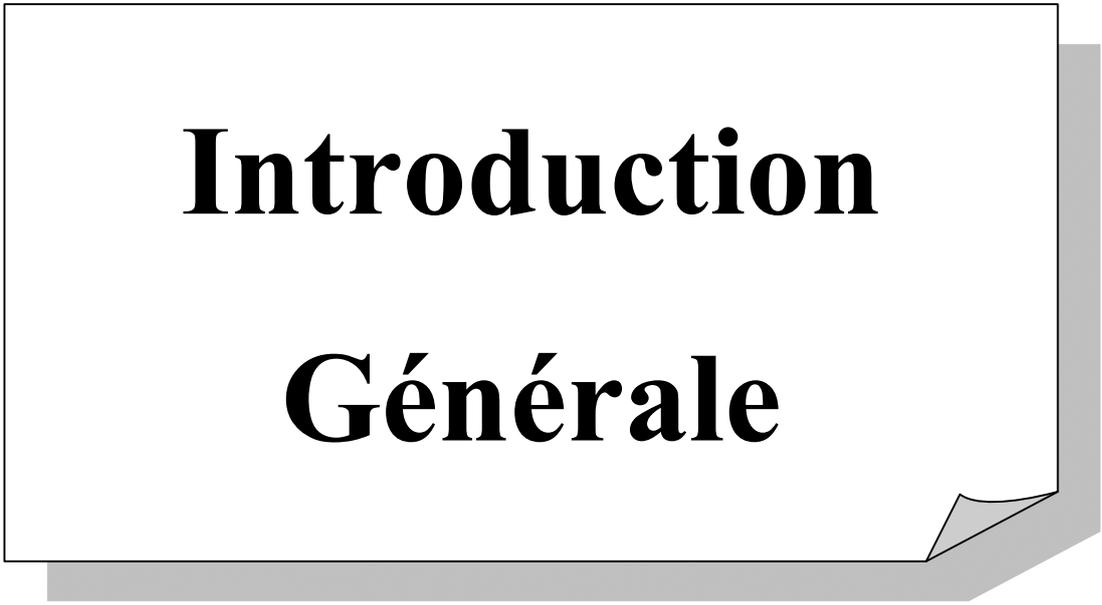
MOSI: Master Out Slave In

SCK: Serial Clock

SS: Slave Select

IDE: Integrated Development Environment

E_R, E_L	[Volts]	Les tensions d'entrée des moteurs
i_a	[A]	Le courant d'induit
L_a	[Henries]	L'inductance d'induit
R_a	[Ω]	La résistance d'induit
k_E	[V_s/rad]	constante de force électromotrice
k_m	[$N.m/A$]	constante de force électromotrice
I	[$kg.m^2$]	L'inertie de rotation du robot
I_w	[$Kg.m^2$]	Le moment d'inertie des roues
I_m	[$Kg.m^2$]	Le moment d'inertie du moteur
f_m	[$N.s.m^{-1}$]	Le frottement visqueux de roue
f_w	[$N.s.m^{-1}$]	Le frottement visqueux du moteur
T_{wR}, T_{wL}	[N.m]	Les couples des roues
T_{mR}, T_{mL}	[N.m]	Les couples des moteurs
l	[m]	Le demi de distance entre la roue droite et la roue gauche
r	[m]	Le rayon de la roue
R	[m]	La distance de l'ICC au point c
V	[m/s]	La vitesse linéaire du robot
ω	[rad/s]	la vitesse angulaire du robot
m	[kg]	La masse du robot
n		Le rapport d'engrenage entre la roue et le moteur
k_v		Le coefficient de résistance au mouvement linéaire
k_ω		Le coefficient de résistance au mouvement rotatif



Introduction Générale

Introduction Générale

Au cours du siècle dernier, grâce à la science de l'automatisation, l'homme a commencé à compter sur la machine pour accomplir des tâches, en particulier complexes et précises. En effet, la machine se caractérise par sa vitesse, sa résistance et sa précision.

Le besoin humain de la machine était l'émergence d'une nouvelle science, la robotique. La science robotique est apparue au début du milieu du siècle dernier et reste une science du futur. La science est un domaine de développement et la croissance est sans précédent, étant donné qu'elle revêt une grande importance sur le marché des technologies.

La robotique s'intéresse à tous les aspects de l'ingénierie, de la construction et de l'exploitation de robots. En général, il s'agit d'entreprendre un système technologique aussi rapidement que possible, avec le moins d'effort et le moins coûteux. Le but est toujours de faciliter l'exécution des différentes tâches de la vie.

Actuellement, le secteur est largement basé sur les robots à bras dans les chaînes de production (soudure, assemblage de composants, peinture, ...), ainsi que sur les robots mobiles qui se déplacent automatiquement dans les usines pour remplir certaines fonctions, telles que le transport d'objets lourds.

Les robots mobiles n'étaient pas limités aux industries, mais étaient utilisés dans la plupart des domaines, en particulier ceux caractérisés par des zones rugueuses et dangereuses telles que les utilisations militaires, les mines, l'exploration spatiale, les missions de sauvetage et les transports.

La navigation autonome est une tâche essentielle pour les robots mobiles. Parce qu'ils seront confrontés à différents types d'obstacles mobiles et statiques. Cela nécessite que ces robots mobiles soient équipés de tous les moyens de perception afin de les contourner en toute sécurité et d'accomplir la tâche donnée.

Dans ce mémoire, le problème de la navigation autonome d'un robot mobile a été abordé en utilisant des techniques floues. La tâche de base que le robot mobile doit faire est de suivre une trajectoire souhaitée, à partir d'une configuration donnée, tout cela sans intervention humaine, et avec le minimum d'erreur.

Le mémoire est organisé comme suit:

Après l'introduction, le premier chapitre présente quelques notions principales au sujet de la robotique mobile.

La modélisation du robot a été faite dans le deuxième chapitre, où un modèle détaillé concernant la partie mécanique et la partie électrique a été développé.

Le troisième chapitre propose une solution au problème de suivi de trajectoire en utilisant un contrôleur flou.

Le dernier chapitre est dédié à la réalisation d'un prototype du robot à base d'une carte Arduino.

Enfin, la conclusion générale présente le bilan de ce travail ainsi que les perspectives envisagées.

Chapitre I:

La navigation Autonome des Robots Mobiles: Généralités

Chapitre I

La navigation Autonome des Robots Mobiles: Généralités

I.1. Introduction:

La robotique est l'union d'un groupe de sciences mécaniques, d'électronique et de programmation. Les roboticiens cherchent à faire fonctionner le robot automatiquement, en lui fournissant un ensemble d'appareils qui lui permettent de percevoir l'environnement dans lequel il se trouve et de prendre la décision appropriée et rationnelle dans le même environnement sans intervention humaine [1].

Initialement, l'utilisation de robots à bras fixe dans l'industrie s'est largement répandue, puis elle a été développée pour devenir portable sur des plates-formes mobiles, d'où l'apparition d'un nouveau type de robot, qui est un robot mobile [1].

En général, nous constatons que les robots mobiles ont une caractéristique fondamentale et sont subjectifs en navigation.

On, grâce à des capteurs et une perception qui sont appropriés à l'environnement dans lequel le robot mobile fonctionne, pour cela la robotique mobile autonome vise plus spécifiquement à concevoir des systèmes capables de se déplacer de façon autonome. Les applications directes se situent notamment dans les domaines de l'automobile, de l'exploration planétaire ou de la robotique de service par exemple [2].

Dans ce chapitre, nous explorerons le domaine des robots mobiles, en particulier les robots à roues. Nous découvrirons également les méthodes et moyens de rendre autonome la navigation des robots.

I.2. La définition d'un robot mobile:

Un robot mobile est une machine mécanique, automatique et programmable capable de se mouvoir dans un environnement donnée.

On regroupe sous cette appellation tous les robots autonomes (c'est-à-dire non télécommandés) capables de se déplacer, par opposition aux robots attachés à un point fixe, comme les robots manipulateurs en industrie [3].

Il existe plusieurs types de robots mobiles et ceux-ci sont, en général, classifiés selon leur type de locomotion c'est-à-dire, selon l'environnement dans lequel ils opèrent, par exemple les robots de recherche sur des planètes dans l'espace ou des robots sous-marins dans les mers pour l'exploration, etc.



Figure I.1: Le robot mobile "Sojourner" de la NASA [26]

I.3. Classification des robots mobiles:

Les robots mobiles sont classés en fonction des tâches qu'ils vont effectuer et des environnements dans lesquels ils sont impliqués. Bien que les robots à roues soient les plus courants [4], il existe d'autres types de robots mobiles. Ainsi, les robots mobiles peuvent être classés comme suit:



Figure I.2: La classification des robots mobiles.

I.3.1. Robots mobiles à roues:

Les roues sont de loin la méthode la plus populaire pour assurer la mobilité des robots et sont utilisées pour propulser de nombreux robots de tailles différentes. Les roues peuvent être à peu près n'importe quelle taille de robots et peuvent avoir à peu près n'importe quel nombre de roues, bien que 3 et 4 soient les plus courantes. Ce type fera l'objet de notre étude au cours du prochain chapitre.

I.3.2. Robots mobiles à chenilles:

Tout comme les chaînes utilisées dans un char militaire, elles réduisent le glissement, ce qui les rend adaptées aux terrains difficiles.

I.3.3. Robots mobiles à pattes:

Les plus couramment utilisés sont les robots à six, quatre ou deux pattes. Il est très adapté aux terrains irréguliers.

I.3.4. Robots mobiles volants:

Appelé AUAV (Autonomous Unmanned Aerial Vehicle), Le plus souvent, l'armée s'en sert pour effectuer des missions de reconnaissance, d'espionnage ou de bombardement.

I.3.5. Robots mobiles sous-marins:

Il se caractérise par la présence de moteurs de poussée, de queue ou d'ailettes, et il est encore en cours de développement car de nombreux problèmes doivent être surmontés.

I.3.6. Robots mobiles hybrides:

Les conceptions diverses peuvent inclure l'aéroglesseur, les conceptions de serpents, les tourelles et plus encore.

I.4. L'autonomie d'un robot mobile:

Comme nous l'avons dit précédemment, le robot mobile est une machine qui a la capacité de percevoir et de prendre des décisions appropriées (rationnellement) dans son environnement sans intervention humaine. Les robots mobiles présentent également une caractéristique très impressionnante qui est sa capacité à effectuer diverses tâches, selon le programme, les actionneurs et les capteurs qui y sont placés.

Une robot autonome peut être définie comme étant l'association des capacités de perception, de modélisation de son environnement et de son propre état, et de capacités d'actions sur son propre état et sur son environnement. Pour cela, le robot doit suivre le schéma correspondant au paradigme (Perception-Décision-Action)[1].

Pour que le robot mobile exécute correctement ses tâches, il doit fournir à son système de contrôle un modèle interne de l'environnement dans lequel il interagit comme la figure I.3 (a). Nous pouvons résumer cela dans la Figure I.3 (b).

- **Perception:** c'est la capacité du robot à obtenir des informations sur l'environnement dans lequel il interagit, grâce à différents capteurs.
- **Décision:** est de définir les procédures que le robot applique après avoir reçu des informations de l'environnement qui l'entoure.
- **Action:** il s'agit de l'application de procédures en envoyant des informations aux les actionneurs par l'intermédiaire des boucles d'asservissements.

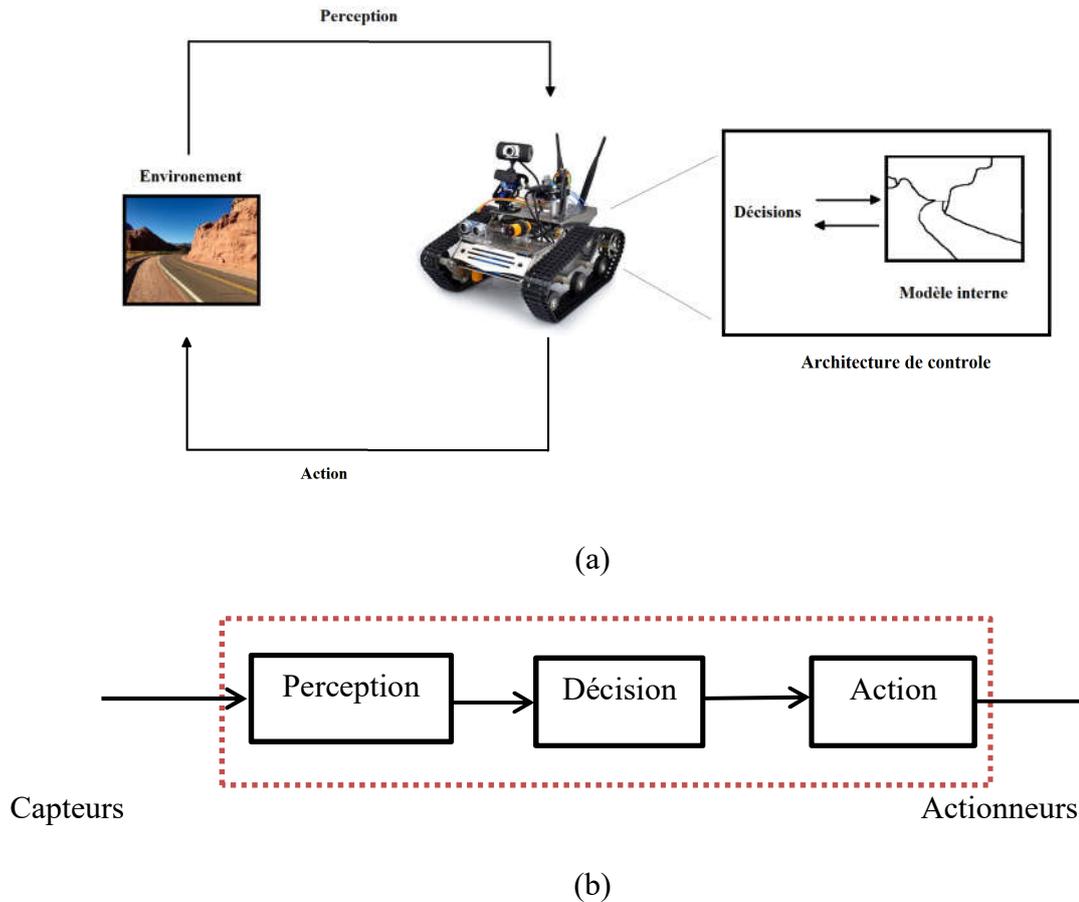


Figure I.3 Interaction entre le robot mobile et l'environnement : a) schéma global, b) paradigme (Perception-Décision-Action)

I.5. Perception:

Pour que les robots mobiles puissent prendre les décisions appropriées, ils doivent avoir une caractéristique essentielle qui est la perception qui les rend plus intelligents dans leur environnement, grâce aux capteurs qu'ils contiennent (Figure I.4).

Le rôle du capteur est d'envoyer au robot des informations sur son environnement de réaction, car il convertit l'état physique en un signal électrique.

On peut distinguer deux catégories de capteurs utilisés dans le robot mobile: le premier est appelé capteurs extéroceptifs, car il fournit des informations sur l'environnement externe du robot mobile, et le second est appelé capteurs proprioceptifs, car il fournit des informations sur l'état interne du robot mobile [7].

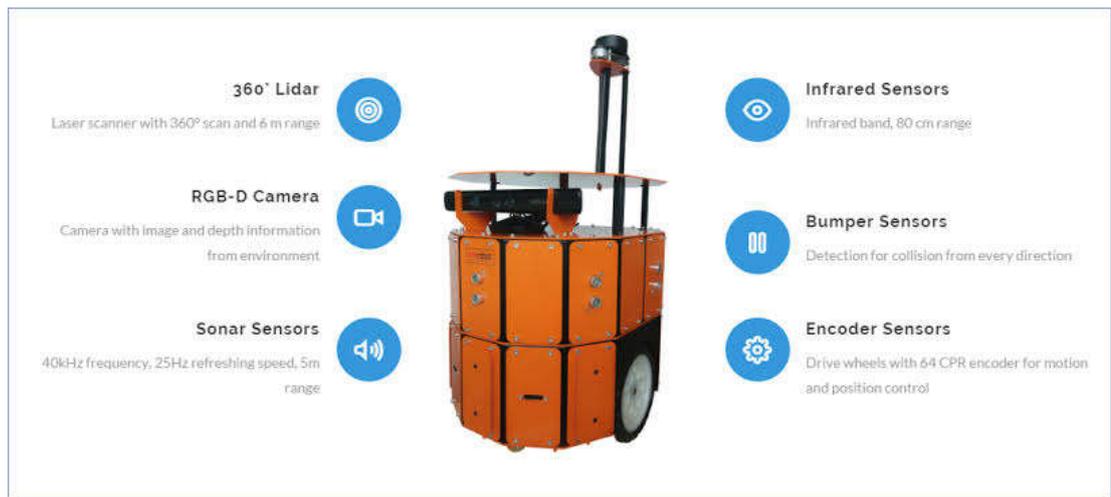


Figure I.4. Exemple de capteurs utilisés par un robot mobile éducatif [27]

I.5.1.1. Les capteurs extéroceptifs:

Comme nous l'avons dit précédemment, cette catégorie de capteur fournit des informations sur l'environnement interactif du robot, par exemple la détection des objets ou des obstacles. Nous mentionnons dans cette catégorie:

- **Les ultrasons:**

Le capteur à ultrason ou télémètre ultrasonores, est considéré comme l'un des capteurs les plus utilisés dans le domaine de la robotique, en raison du prix et de la facilité d'utilisation.

Le capteur à ultrasons est composé d'une paire émetteur-récepteur. Il utilise la notion de temps de vol pour mesurer la distance. Le temps de vol est calculé entre l'instant d'émission et sa réception après réflexion.



Figure I.5. Capteurs ultrasons

- **Les capteurs infrarouges:**

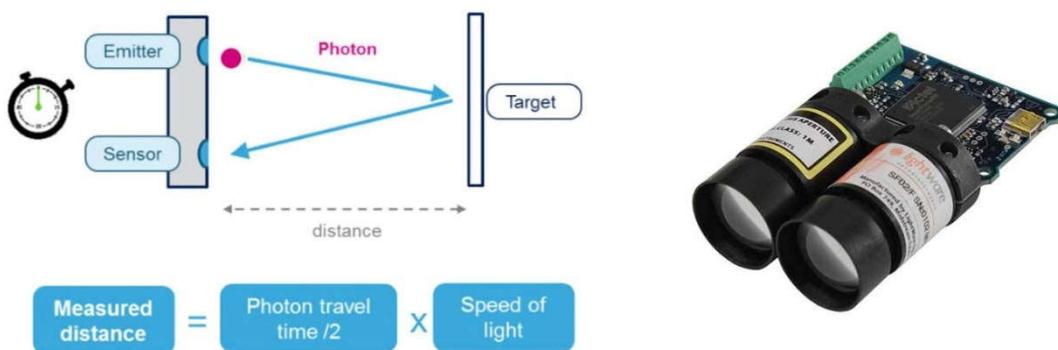
Les capteurs infrarouges fonctionnent sur le même principe que les capteurs à ultrasons. Ils sont constitués d'une paire émetteur-récepteur. L'émission est réalisée par une diode électroluminescente infrarouge. Le récepteur est une photodiode ou un phototransistor, sensible au flux lumineux rétrodiffusé par la cible [7].



Figure I.6. Capteurs infrarouges

- **Les lasers:**

Il existe plusieurs façons de calculer la distance et de détection l'objet par laser, mais le principe de base dépend de l'envoi d'une impulsion laser, puis de sa réception (après sa réflexion sur un objet) par des circuits spéciaux, à partir le temps de vol (du moment de la transmission au moment de la réception), la distance peut être calculée (Figure I.7).



Figures I.7. Capteur laser

- **Les capteurs tactiles:**

Les capteurs tactiles sont simples, peu coûteux et sont utilisés pour fournir plus de protection au robot. Elles sont des détecteurs très sensibles à la touche. Il existe plusieurs types qui peuvent varier en termes de formes et de niveaux de sensibilité [9].

- **Le GPS(Global Positioning Systems):**

Il s'agit d'un système mondial qui permet de déterminer n'importe quel emplacement sur la terre par des satellites en orbite. La position du robot est déterminée par les récepteurs placés dessus.

Le GPS est un moyen efficace de localiser un robot mobile dans un environnement extérieur. La localisation du robot est obtenue en mesurant le temps de vol des signaux émis par au moins quatre satellites séparés [10].

- **Les caméras:**

La caméra a la capacité de donner une image sensorielle similaire à celle de l'environnement interactif du robot. En appliquant des algorithmes de traitement d'image, le robot peut extraire des éléments spécifiques à son environnement.

- **Les capteurs à micro-ondes:**

Le capteur à micro-onde émet des ondes électromagnétiques de l'ordre du GHz. Il peut fonctionner en trois modes différents [7,9].

I.5.1.2. Les capteurs proprioceptifs:

Cette catégorie de capteurs fournit des informations sur l'état du robot mobile et ces informations généralement sont: les vitesses, les accélérations, les angles de rotation, les angles d'altitudes. On peut regrouper les capteurs proprioceptifs en deux familles: les capteurs de déplacement et les capteurs d'attitude.

I.5.1.2.1. Les capteurs de déplacement:

- **L'odomètre:**

L'odomètre fournit la position du robot mobile lors de son déplacement, en intégrant les rotations élémentaires de ses roues. En général, les encodeurs sont souvent utilisés pour mesurer la rotation des roues [9].

Il y a deux types d'encodeurs: les encodeurs incrémentaux et les encodeurs absolus. Les codeurs incrémentaux mesurent la vitesse angulaire et déduisent la position relative, et les encodeurs absolus mesurent directement la position angulaire et déduisent la vitesse [7,9].

- **Les accéléromètres:**

Un accéléromètre est un capteur qui permet de mesurer l'accélération linéaire en un point défini, c'est au moyen d'une masse M placée sur le capteur.

Le principe de ce capteur est de mesurer l'effort massique non gravitationnel qu'on doit appliquer à M pour le maintenir en place dans le boîtier lorsqu'une accélération est appliquée au boîtier [7,9].

- **Le capteur radar Doppler:**

Le capteur radar Doppler a son principe basé sur l'envoi d'un faisceau électromagnétique dirigé vers la Terre à une fréquence F , puis il est reçu après sa réflexion sur le sol.

I.5.1.2.2. Les capteurs d'attitude:

Les capteurs d'attitude sont des capteurs très sensibles et souvent utilisés dans le domaine de l'aviation, fournissant des informations importantes telles que la vitesse angulaire et l'angle de rotation et le cap. Nous les mentionnons:

- **Les gyroscopes:**

Le gyroscope est un capteur qui mesure de l'angle de rotation (position angulaire).

- **Le gyromètre:**

Le gyromètre est un capteur qui mesure la vitesse angulaire.

- **Le gyrocompas:**

Le gyrocompas est un capteur utilisé pour mesurer le cap (heading). Il est composé d'un gyroscope et d'un compas magnétique.

I.6. La navigation autonome d'un robot mobile

Comme nous l'avons dit, la navigation autonome du robot est sa capacité à percevoir son environnement interactif et à prendre les décisions appropriées sans intervention humaine.

Nous pouvons le simplifier facilement, pour que le robot mobile parvienne à une bonne auto-navigation, il doit répondre aux trois questions suivantes:

- “Where am I?” (La localisation)
- “Where am I going?” (Le cartographie ou mapping)
- “How do I get there?” (La navigation).

I.6.1. La navigation:

La navigation d'un robot mobile est une tâche où le robot peut se déplacer librement sur les trajectoires dans l'espace de configuration, sans collision avec des obstacles. Nous appelons ce C_{libre} [1,5].

L'espace de configuration est l'ensemble des paramètres caractérisant la position du robot dans son environnement (position et orientation). Ce mouvement amène le robot d'une configuration initiale $q_0=q(t_0)=(x_0, y_0, \theta_0)$ à une autre finale désirée $q_f=q(t_f)=(x_f, y_f, \theta_f)$.

Nous ne pouvons pas trouver une méthode spécifique pour naviguer sur un robot mobile car il existe de nombreux modes de navigation, mais il existe des stratégies qu'un robot suit Pour exécuter une tâche de navigation autonome.

I.6.1.1. Planification de mouvement:

Parmi les objectifs principaux 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 (point de démarrage)[6].

On a trois approches sont proposées pour la planification de trajectoire. Cependant, les plus utilisées sont la planification globale, locale et hybride.

- **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.

- **Planification locale de trajectoire:**

Le robot mobile ne dispose aucune information sur l'environnement parce que l'environnement du robot est inconnu. 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.

- **Planification hybride de trajectoire:**

Un groupe d'algorithmes suggère une combinaison des méthodes de navigation locale et de planification globale des trajectoires. Ces algorithmes visent à combiner les avantages des méthodes locales et globales, et à éliminer également certaines de leurs faiblesses [7].

I.6.1.2. Le suivi de trajectoire:

Le suivi des lignes ou des trajectoires est une fonction essentielle des robots mobiles, elle consiste à calculer les commandes envoyées aux actionneurs permettant de réaliser le mouvement planifié [1].

I.6.1.3. L'évitement d'obstacles:

Éviter les obstacles est une fonction principale qui doit être disponible dans tous les robots mobiles, et c'est la capacité du robot à éviter les collisions avec des objets ou des obstacles grâce aux capteurs de distance.

I.6.2. Localisation:

Avant de faire la planification de mouvement, le robot mobile doit se localiser dans son environnement. La localisation est l'estimation de la position et de l'orientation du robot mobile par rapport à un référentiel considéré fixe [7].

I.6.2.1. Localisation relative:

C'est la détermination de la position et de l'orientation d'un robot à l'aide des capteurs proprioceptifs (odomètre, vélocimètre, capteurs inertiels) [7].

- **L'odométrie:**

Est la détermination de la position et de l'orientation d'un robot mobile en intégrant ses déplacements élémentaires, par rapport à un référentiel global ou fixe. Cette technique utilise des encodeurs montés sur les roues [7,8]. Ceci est plus utilisé dans la robotique mobile en raison du coût, de la précision et de la facilité de mise en œuvre.

- **Localisation inertielle**

Ce type de localisation utilise des types spécifiques de capteurs: accéléromètres, gyroscopes et compas magnétiques, pour obtenir des informations à partir du mouvement du robot, qui permet de calculer la variation de la position et l'orientation.

- **Localisation basée sur les capteurs d'images**

Basé sur la mesure du mouvement apparent dans l'image, en utilisant une technique de corrélation entre deux images consécutives [7].

- **La vélocimétrie**

Cette méthode permet de mesurer toutes les vitesses dans le robot mobile pour d'obtenir le déplacement. La vitesse de rotation est mesurée par des gyromètres, et la vitesse linéaire est mesurée par le radar Doppler.

I.6.2.2. Localisation absolue

La localisation absolue est une technique qui permet aux robots mobiles de se localiser dans leur environnement, grâce à des capteurs extéroceptifs. Cela nécessite une connaissance

de l'environnement. Le robot mobile doit contenir une base de données qui regroupant des éléments caractéristiques de son environnement par exemple: des balises, à partir desquelles le robot doit reconnaître ces éléments.

- **Localisation par compas magnétique:**

Un compas magnétique permet de déterminer une orientation en mesurant la composante horizontale du champ magnétique terrestre [8]. Le principal inconvénient est de travailler dans un environnement contenant des lignes à haute tension ou des structures métalliques grandes.

- **Localisation par balises actives**

Les balises actives sont ajoutées à l'environnement du robot mobile pour déterminer la position du robot. Ils émettent l'onde captée par les capteurs du robot. Deux principaux types de transmission sont utilisés: les émetteurs de sources lumineuses (comme l'infrarouge) et les antennes microwave [7].

- **Localisation par balises passives**

Ce sont des repères qui ne communiquent pas avec les capteurs du robot, ils peuvent être des codes-barres ou toute marque ou objet identifiable et avoir une position spécifique [7,8].

- **Localisation basée sur un modèle**

Il est basé sur le stockage d'un modèle interne de l'environnement interactif du robot dans sa mémoire. Le robot peut se localiser, s'il trouve une correspondance entre le plan local de l'environnement et la pièce correspondante dans le modèle global [8].

I.6.3. La cartographie (mapping)

La cartographie est le processus de représentation de l'environnement interactif (l'espace de navigation) d'un robot mobile avec sa position dans cet environnement lors de sa navigation, en utilisant les données de ses capteurs [5,11].

Les cartes peuvent être divisées en trois classes :

I.6.3.1. Les cartes métriques

Les objets de la carte ont des positions précises dans un espace métrique, généralement dans deux dimensions 2D. Il se divise en deux familles: représentation purement géométrique qui utilise explicitement les positions «cartésiennes», et représentation par grille d'occupation, qui décrit les propriétés métriques en discrétisant l'environnement en ajoutant des informations d'incertitude.

I.6.3.2. La représentation topologique

Dans cette représentation l'environnement est représenté par des formes de graphe et des nœuds. Les nœuds du graphe correspondent à des positions que le robot peut atteindre. Les arêtes liant les nœuds marquent la possibilité pour le robot de passer directement d'un lieu à un autre et mémorisent en général la manière de réaliser ce passage.

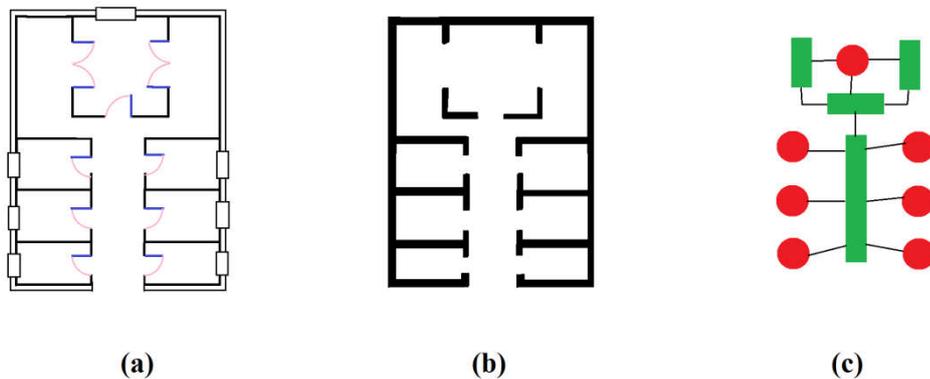


Figure I.8. Les cartes utilisées en robotique mobile: (a) l'environnement réel, (b) carte métrique, (c) carte topologique

I.7. Conclusion:

La robotique est une science qui s'intéresse à l'étude et au développement de systèmes automatisés afin qu'ils puissent interagir idéalement au sein de leur environnement.

Dans ce chapitre, nous avons identifié les robots mobiles et les classer en fonction de leur environnement de travail, et nous avons également introduit la caractéristique fondamentale des robots mobiles, qui est la navigation autonome, c'est-à-dire la capacité de naviguer, localiser et cartographier grâce à un ensemble de capteurs internes et externes qui permettent la perception de cet environnement.

Dans le chapitre suivant, nous établirons le modèle cinématique et dynamique du robot à entraînement différentiel.

Chapitre II:

Modélisation d'un robot à entraînement différentiel

Chapitre II

Modélisation d'un robot à entraînement différentiel

II.1. Introduction:

Les robots mobiles à roues sont les plus largement utilisés dans tous les domaines et sont donc extrêmement importants dans le domaine de la recherche et du développement, nous savons que ces objets sont cinématiques, et lorsque des vitesses élevées ou des charges lourdes sont nécessaires, il est nécessaire de penser également à la dynamique du robot, c'est-à-dire à la modélisation cinématique et dynamique.

La modélisation cinématique est la formulation de contraintes entre les positions, les vitesses et l'accélération des composants du mécanisme tandis que la dynamique du robot liée aux masses / inertie, forces / couple et mouvements, qui fournit des performances précises des opérations du site et le contrôle du chemin cartésien - espace.

Dans ce chapitre, nous présentons la modélisation cinématique et dynamique du robot à entraînement différentiel et nous vérifierons la validité de modélisation par la simulation.

II.2. Description du robot à entraînement différentiel:

Les robots mobiles à roues à entraînement différentiel sont les robots mobiles les plus courants, le robot est équipé de deux roues motrices. Une de chaque côté du robot. Chaque bloc, roue + moteur, est indépendant mécaniquement et électriquement. Il n'y a pas d'axe mécanique reliant les axes moteurs et les moteurs sont alimentés individuellement. La seule liaison mécanique est le châssis du robot (comme support). La plateforme est équipée d'une roue libre pour l'équilibre du robot. Il s'appelle en anglais Differential Drive Mobile Robot (DDMR).

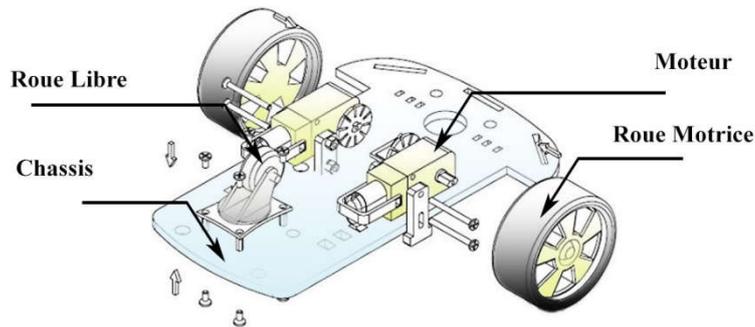


Figure II.1. Configuration du robot

II.3. La modélisation cinématique:

La cinématique est l'étude fondamentale pour comprendre les systèmes mécaniques. Et pour de conception correct des robots mobiles adaptés aux tâches qui leur sont assignées et proportionnels à leur programme de contrôle, nous devons étudier et comprendre le comportement mécanique de ces robots [12].

La modélisation cinématique est l'étude du mouvement des systèmes mécaniques sans tenir compte des forces qui affectent le mouvement. Pour le robot à entraînement différentiel, l'objectif principal de la modélisation cinématique est de représenter les vitesses du robot en fonction des vitesses des roues motrices ainsi que les paramètres géométriques du robot [13].

II.4. Le modèle de mouvement:

Un système de propulsion intérieur très commun utilise deux roues motrices à commande individuelle de chaque côté de la base, avec une roue de stabilité à l'avant ou à l'arrière pour plus de stabilité. Cette configuration permet au robot de tourner en place autour de l'axe vertical pour manœuvrer dans des zones encombrées.

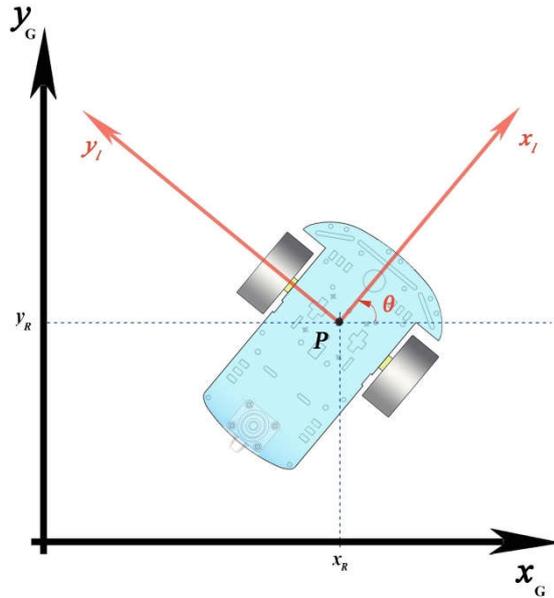


Figure II.2. Le repère global du robot

Supposons un repère global (o, x_G, y_G) qui représente l'environnement du robot à entraînement différentiel (figure II.2). On peut également définir un repère local (o_l, x_l, y_l) associée au centre de masse du robot. L'angle θ de l'axe x_l par rapport à l'axe x peut être défini comme:

$$\theta = \int \dot{\theta} dt \quad (\text{II.1})$$

Où $\omega = \dot{\theta}$ est la vitesse angulaire du robot. La transformation des positions entre le repère global et le repère local est donnée par:

$$X_G = T_G^l X_l \quad (\text{II.2})$$

Où, T_G^l est la matrice de transformation qui peut s'écrire:

$$T_G^l = \begin{bmatrix} \cos \varphi & -\sin \varphi & x_R \\ \sin \varphi & \cos \varphi & y_R \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{II.3})$$

Donc:

$$\begin{bmatrix} x_G \\ y_G \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi & x_R \\ \sin \varphi & \cos \varphi & y_R \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_l \\ y_l \\ 1 \end{bmatrix} \quad (\text{II.4})$$

La transformation des vitesses entre le repère global et le repère local est donnée comme suite:

$$\dot{X}_G = R_G^l \dot{X}_l \quad (\text{II.5})$$

Où, R_G^l est la matrice de transformation qui peut s'écrire:

$$R_G^l = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{II.6})$$

Le mouvement vers l'avant est obtenu lorsque les deux roues sont entraînées à la même vitesse. Tourner à gauche est obtenu en conduisant la roue droite à un taux plus élevé que la roue gauche. Conduire la roue gauche à un taux plus élevé entraîne un virage à droite. Lorsqu'une roue tourne en avant et que l'autre tourne en sens inverse, le robot tourne sur place.

II.5. Le modèle cinématique:

Nous considérons les forces agissant sur la roue droite et gauche, F_R et F_L qui résultent respectivement des deux vitesses linéaires V_R et V_L (figure II.3), la vitesse du robot est déterminée par la vitesse linéaire V et la vitesse angulaire ω .

Si la roue droite tourne alors que la roue gauche ne contribue à rien et est immobile, elle se déplacera instantanément avec la moitié de la vitesse, puisque le point c est à mi-chemin entre les deux roues. Nous pouvons définir cela dans la relation suivante:

$$V_{cR} = \frac{V_R}{2} \quad (\text{II.7})$$

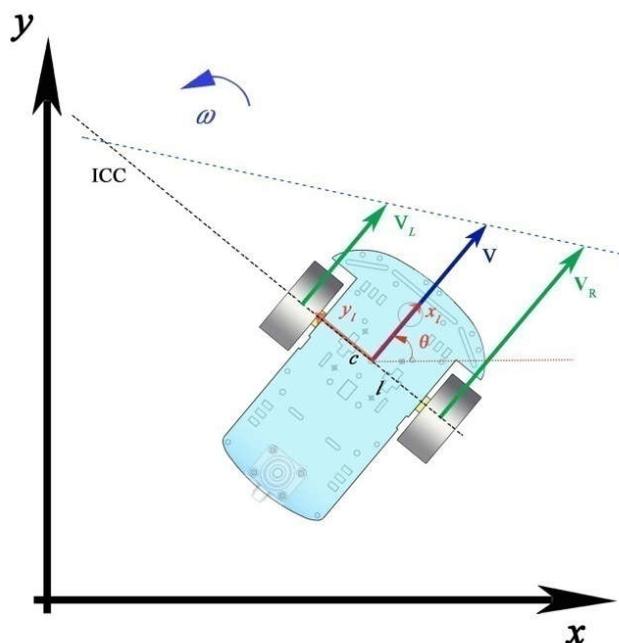


Figure II.3. Les forces sur les roues : une représentation équivalente

Et l'inverse est vrai, si la roue gauche tourne alors que la roue droite ne contribue à rien et est à l'arrêt, elle se déplacera instantanément à la moitié de la vitesse:

$$V_{cL} = \frac{V_L}{2} \quad (\text{II.8})$$

En ajoutant les deux dernières relations, nous pouvons calculer la vitesse V du point c

$$V = V_{cR} + V_{cL} = \frac{V_R + V_L}{2} \quad (\text{II.9})$$

De la même manière, nous pouvons calculer la vitesse angulaire ω , considérons la roue droite. Si la roue droite tourne seule, le robot pivote autour de la gauche. La vitesse de rotation en c peut être calculée car la roue se déplace instantanément le long de l'arc d'un cercle de rayon $2l$:

$$\omega_{cR} = \frac{V_R}{2l} \quad (\text{II.10})$$

De même pour la roue gauche, si la roue gauche tourne seule, le robot pivote autour de la droite:

$$\omega_{cL} = \frac{V_L}{2l} \quad (\text{II.11})$$

En ajoutant les deux dernières relations, nous pouvons calculer la vitesse angulaire ω du point c :

$$\omega = \omega_{cR} + \omega_{cL} = \frac{V_R + V_L}{2l} \quad (\text{II.12})$$

Nous avons obtenu les équations précédentes dans le cas où l'une des roues du robot est arrêtée et l'autre en rotation. Et nous verrons le cas où les deux roues sont en rotation.

En faisant varier la vitesse de chaque roue, le robot peut effectuer un mouvement de roulement; ce mouvement est effectué autour d'un point commun appelé Instantaneous Center of Curvature (ICC) qui se trouve le long de l'axe commun des roues gauche et droite (Figure II.3).

La vitesse angulaire ω autour de l'ICC est la même pour les deux roues, nous pouvons écrire l'équation suivante:

$$\omega = \frac{V_R}{R+l} = \frac{V_L}{R-l} \quad (\text{II.13})$$

De cela nous obtenons:

$$V_R = \omega(R + l) \quad (\text{II.14})$$

$$V_L = \omega(R - l) \quad (\text{II.15})$$

Où, V_R est la vitesse de translation de la roue droite, V_L est la vitesse de translation de la roue gauche, R est la distance de l'ICC au centre du robot et l est le demi de distance entre la roue droite et la roue gauche.

On soustrait l'équation (II.14) de l'équation (II.15), nous obtenons la vitesse angulaire:

$$\omega = \frac{V_R - V_L}{2l} \quad (\text{II.16})$$

En ajoutant les mêmes deux équations, nous obtenons:

$$V_R + V_L = 2\omega R \quad (\text{II.17})$$

Donc à partir l'équation (2.9) la vitesse de translation:

$$V = \frac{V_R + V_L}{2} = \omega R \quad (\text{II.18})$$

À partir de là, nous pouvons calculer R :

$$R = \frac{V}{\omega} = l \frac{V_R + V_L}{V_R - V_L} \quad (\text{II.19})$$

Alors que les coordonnées de l'ICC sont:

$$ICC = \begin{bmatrix} ICC_x \\ ICC_y \end{bmatrix} = \begin{bmatrix} x_R - R \cos \theta \\ y_R - R \sin \theta \end{bmatrix} \quad (\text{II.20})$$

La cinématique avant dans le repère local est donnée par:

$$\begin{bmatrix} V \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2l} & -\frac{1}{2l} \end{bmatrix} \begin{bmatrix} V_R \\ V_L \end{bmatrix} \quad (\text{II.21})$$

La cinématique inverse est donnée par:

$$\begin{bmatrix} V_R \\ V_L \end{bmatrix} = \begin{bmatrix} 1 & l \\ 1 & -l \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \quad (\text{II.22})$$

Puisque la vitesse de translation V est toujours dans la direction x_l , alors, $x_l = V$ et $y_l = 0$. La cinématique avant peut s'écrire:

$$\begin{bmatrix} \dot{x}_l \\ \dot{y}_l \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & 0 \\ \frac{1}{2l} & -\frac{1}{2l} \end{bmatrix} \begin{bmatrix} V_R \\ V_L \end{bmatrix} \quad (\text{II.23})$$

A partir l'équation (II.6), la transformation entre le repère local et le repère global peut être donnée par:

$$\begin{bmatrix} \dot{x}_G^l \\ \dot{y}_G^l \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \cos \theta & \frac{1}{2} \cos \theta \\ \frac{1}{2} \sin \theta & \frac{1}{2} \sin \theta \\ \frac{1}{2l} & -\frac{1}{2l} \end{bmatrix} \begin{bmatrix} V_R \\ V_L \end{bmatrix} \quad (\text{II.24})$$

V_R et V_L sont les entrées du contrôleur, où:

$$V_R = \omega_R r \quad (\text{II.25})$$

$$V_L = \omega_L r \quad (\text{II.26})$$

Donc, l'équation est le modèle cinématique du robot est:

$$\begin{bmatrix} \dot{x}_G^l \\ \dot{y}_G^l \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ \frac{r}{2} \sin \theta & \frac{r}{2} \sin \theta \\ \frac{r}{2l} & -\frac{r}{2l} \end{bmatrix} \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} \quad (\text{II.27})$$

II.6. La modélisation dynamique:

La dynamique est l'étude du mouvement d'un système mécanique en tenant compte des différentes forces qui affectent son mouvement contrairement à la cinématique où les forces ne sont pas prises en compte [13].

Le modèle dynamique du robot à entraînement différentiel est essentiel pour l'analyse de simulation du mouvement et pour la conception de divers algorithmes de contrôle de mouvement (Figure II.4).

En appliquant la deuxième loi de Newton au système, on obtient :

$$\begin{cases} \sum F_i = m \dot{V} \\ \sum M_i = j \dot{\omega} \end{cases} \quad (\text{II.28})$$

$$\begin{cases} F + F_s = m \dot{V} \\ M_r + M_l + M_s = I \dot{\omega} \end{cases} \quad (\text{II.29})$$

Où, m [kg] et I [kg.m^2] sont la masse et l'inertie de rotation du robot.

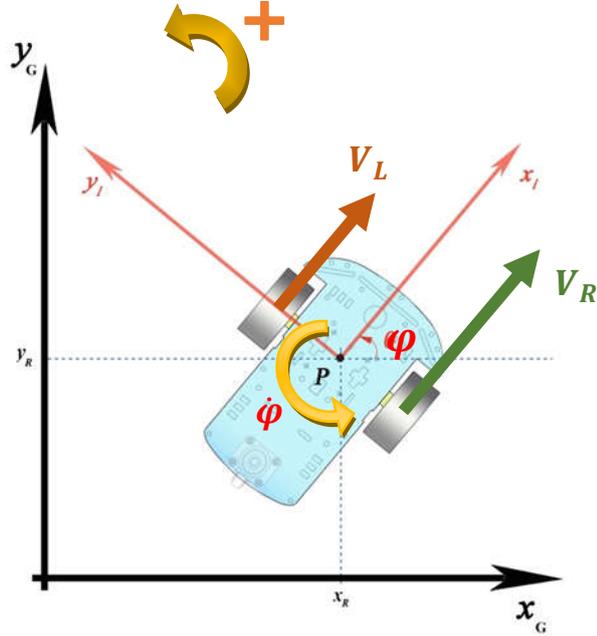


Figure II.4. Schéma des vitesses appliquées au robot

$$\begin{cases} \frac{1}{2} F_R + \frac{1}{2} F_L - k_v V = m \dot{V} \\ F_r l - F_l l - k_\omega \omega = I \dot{\omega} \end{cases} \quad (\text{II.30})$$

k_v est le coefficient de résistance au mouvement linéaire; et k_ω est le coefficient de résistance au mouvement rotatif.

La vitesse linéaire et la vitesse angulaire sont données par:

$$\begin{cases} V = \frac{1}{2} V_R + \frac{1}{2} V_L \\ \omega = \frac{1}{2l} V_R - \frac{1}{2l} V_L \end{cases} \quad (\text{II.31})$$

En prenant la dérivée temporelle de l'équation (II.30), on obtient:

$$\begin{cases} \dot{V} = \frac{1}{2} \dot{V}_R + \frac{1}{2} \dot{V}_L \\ \dot{\omega} = \frac{1}{2l} \dot{V}_R - \frac{1}{2l} \dot{V}_L \end{cases} \quad (\text{II.32})$$

Substituons (II.32) dans (II.30), on obtient :

$$\begin{cases} \frac{1}{2} F_R + \frac{1}{2} F_L - k_v V = \frac{m}{2} \dot{V}_R + \frac{m}{2} \dot{V}_L \\ F_r l - F_l l - k_\omega \omega = \frac{I}{2l} \dot{V}_R - \frac{I}{2l} \dot{V}_L \end{cases} \quad (\text{II.33})$$

Alors ;

$$\begin{cases} F_R + F_L = m \dot{V}_R + m \dot{V}_L + k_v V_R + k_v V_L \\ F_R - F_L = \frac{l}{2l^2} \dot{V}_R - \frac{l}{2l^2} \dot{V}_L + \frac{k_\omega}{2l^2} V_R - \frac{k_\omega}{2l^2} V_L \end{cases} \quad (\text{II.34})$$

On résout l'équation (II.34), on obtient:

$$\begin{cases} F_R = \left(\frac{2l^2 m + l}{4l^2}\right) \dot{V}_R + \left(\frac{2l^2 m - l}{4l^2}\right) \dot{V}_L + \left(\frac{2l^2 k_v + k_\omega}{4l^2}\right) V_R + \left(\frac{2l^2 k_v - k_\omega}{4l^2}\right) V_L \\ F_L = \left(\frac{2l^2 m - l}{4l^2}\right) \dot{V}_R + \left(\frac{2l^2 m + l}{4l^2}\right) \dot{V}_L + \left(\frac{2l^2 k_v - k_\omega}{4l^2}\right) V_R + \left(\frac{2l^2 k_v + k_\omega}{4l^2}\right) V_L \end{cases} \quad (\text{II.35})$$

Maintenant, les relations entre les forces de traction et les couples des roues $[T_{wR}, T_{wL}]$ en [N.m] peuvent être données:

$$\begin{cases} T_{wR} = I_w \left(\frac{\dot{V}_R}{r}\right) + f_w \left(\frac{V_R}{r}\right) + F_R r \\ T_{wL} = I_w \left(\frac{\dot{V}_L}{r}\right) + f_w \left(\frac{V_L}{r}\right) + F_L r \end{cases} \quad (\text{II.36})$$

Où I_w [Kg.m²] est le moment d'inertie des roues; f_w [N.s.m⁻¹] est le coefficient de frottement visqueux, et r [m] est le rayon de la roue. $\left(\frac{V_R}{r}, \frac{V_L}{r}\right)$ et $\left(\frac{\dot{V}_R}{r}, \frac{\dot{V}_L}{r}\right)$ sont respectivement les vitesses angulaires et les accélérations angulaires des roues.

La relation entre le couple de roue et le couple moteur $[T_{mR}, T_{mL}]$ en [N.m] est donnée comme suit:

$$\begin{cases} T_{mR} = I_m \left(n \frac{\dot{V}_R}{r}\right) + f_m \left(n \frac{V_R}{r}\right) + \frac{T_{wR}}{n} \\ T_{mL} = I_m \left(n \frac{\dot{V}_L}{r}\right) + f_m \left(n \frac{V_L}{r}\right) + \frac{T_{wL}}{n} \end{cases} \quad (\text{II.37})$$

Où I_m [Kg.m²] est le moment d'inertie du moteur, f_m [N.s.m⁻¹] est le frottement visqueux du moteur, et n est le rapport d'engrenage entre la roue et le moteur. $\left(n \frac{V_R}{r}, n \frac{V_L}{r}\right)$ et $\left(n \frac{\dot{V}_R}{r}, n \frac{\dot{V}_L}{r}\right)$ sont respectivement les vitesses angulaires et les accélérations angulaires des moteurs.

En remplaçant l'équation (II.36) dans (II.37), nous obtenons:

$$\begin{cases} T_{mR} = \left(nI_m + \frac{I_w}{n}\right) \frac{\dot{V}_R}{r} + \left(nf_m + \frac{f_w}{n}\right) \frac{V_R}{r} + \frac{F_R r}{n} \\ T_{mL} = \left(nI_m + \frac{I_w}{n}\right) \frac{\dot{V}_L}{r} + \left(nf_m + \frac{f_w}{n}\right) \frac{V_L}{r} + \frac{F_L r}{n} \end{cases} \quad (\text{II.38})$$

Substituons les forces de l'équation (II.35) dans l'équation précédente (II.38), on obtient après l'arrangement:

$$\begin{cases} T_{mR} = \left(\frac{n^2 I_m + I_w}{nr} + \frac{2l^2 m r + l r}{4l^2 n}\right) \dot{V}_R + \left(\frac{n^2 f_m + f_w}{nr} + \frac{2l^2 k_v r + k_\omega r}{4l^2 n}\right) V_R + \left(\frac{2l^2 m r - l r}{4l^2 n}\right) \dot{V}_L + \left(\frac{2l^2 k_v r - k_\omega r}{4l^2 n}\right) V_L \\ T_{mL} = \left(\frac{n^2 I_m + I_w}{nr} + \frac{2l^2 m r + l r}{4l^2 n}\right) \dot{V}_L + \left(\frac{n^2 f_m + f_w}{nr} + \frac{2l^2 k_v r + k_\omega r}{4l^2 n}\right) V_L + \left(\frac{2l^2 m r - l r}{4l^2 n}\right) \dot{V}_R + \left(\frac{2l^2 k_v r - k_\omega r}{4l^2 n}\right) V_R \end{cases} \quad (\text{II.39})$$

Nous avons modélisé la partie mécanique du robot décrite par l'équation (II.29) qui est la dynamique du robot qui relie le couple moteur à la vitesse des roues. L'étape suivante consiste à modéliser la partie électrique des moteurs du robot.

Pour un moteur à courant continu, nous avons:

$$\begin{cases} L_a \frac{di_{aR}}{dt} + R_a i_{aR} + k_E \left(n \frac{V_R}{r} \right) = E_R \\ L_a \frac{di_{aL}}{dt} + R_a i_{aL} + k_E \left(n \frac{V_L}{r} \right) = E_L \end{cases} \quad (\text{II.40})$$

E_R et E_L [Volts] sont les tensions d'entrée des moteurs, i_a [A] est le courant d'induit, L_a [Henries] est l'inductance d'induit, R_a [Ω] est la résistance d'induit, est k_E [V_S/rad] la constante de force électromotrice arrière du moteur.

Puisque l'inductance du moteur est omise, car elle est petite et généralement ignorée dans la dynamique du robot, on peut négliger la dynamique du circuit électrique du moteur, $L_a \frac{di_a}{dt} = 0$, ce qui conduit à:

$$\begin{cases} R_a i_{aR} + k_E \left(n \frac{V_R}{r} \right) = E_R \\ R_a i_{aL} + k_E \left(n \frac{V_L}{r} \right) = E_L \end{cases} \quad (\text{II.41})$$

Le couple T_m développé par le moteur est donné par l'équation suivante:

$$\begin{cases} T_{mR} = k_m i_{aR} \\ T_{mL} = k_m i_{aL} \end{cases} \quad (\text{II.42})$$

Où k_m [$N.m/A$] est la constante de couple du moteur.

À partir des équations (II.41) et (II.42), nous obtenons:

$$\begin{cases} T_{mR} = \frac{k_m}{R_a} E_R - \frac{k_m k_E n}{R_a r} V_R \\ T_{mL} = \frac{k_m}{R_a} E_L - \frac{k_m k_E n}{R_a r} V_L \end{cases} \quad (\text{II.43})$$

Substituons cette équation dans (II.39), on trouve :

$$\begin{cases} \left(\frac{n^2 I_m + I_w}{nr} + \frac{2 l^2 m r + l r}{4 l^2 n} \right) \dot{V}_R + \left(\frac{n^2 f_m + f_w}{nr} + \frac{2 l^2 k_v r + k_\omega r}{4 l^2 n} + \frac{k_m k_E n}{R_a r} \right) V_R + \left(\frac{2 l^2 m r - l r}{4 l^2 n} \right) \dot{V}_L + \left(\frac{2 l^2 k_v r - k_\omega r}{4 l^2 n} \right) V_L = \frac{k_m}{R_a} E_R \\ \left(\frac{n^2 I_m + I_w}{nr} + \frac{2 l^2 m r + l r}{4 l^2 n} \right) \dot{V}_L + \left(\frac{n^2 f_m + f_w}{nr} + \frac{2 l^2 k_v r + k_\omega r}{4 l^2 n} + \frac{k_m k_E n}{R_a r} \right) V_L + \left(\frac{2 l^2 m r - l r}{4 l^2 n} \right) \dot{V}_R + \left(\frac{2 l^2 k_v r - k_\omega r}{4 l^2 n} \right) V_R = \frac{k_m}{R_a} E_L \end{cases} \quad (\text{II.44})$$

L'équation (II.44) est la dynamique du robot différentiel, c'est un système de deux équations non linéaires couplées. Où les tensions E_R et E_L sont les variables d'entrée, tandis

que les vitesses des roues E_R et E_L sont les variables de sortie. Cette équation peut s'écrire comme suit :

$$\begin{cases} K_1 \dot{V}_R + K_2 \dot{V}_L + K_3 V_R + K_4 V_L = \frac{k_m}{R_a} E_R \\ K_1 \dot{V}_L + K_2 \dot{V}_R + K_3 V_L + K_4 V_R = \frac{k_m}{R_a} E_L \end{cases} \quad (\text{II.45})$$

Ou encore,

$$\begin{cases} \dot{V}_R = \frac{1}{K_1} [-K_2 \dot{V}_L - K_3 V_R - K_4 V_L + \frac{k_m}{R_a} E_R] \\ \dot{V}_L = \frac{1}{K_1} [-K_2 \dot{V}_R - K_3 V_L - K_4 V_R + \frac{k_m}{R_a} E_L] \end{cases} \quad (\text{II.46})$$

Dont

$$\begin{aligned} K_1 &= \left(\frac{n^2 I_m + I_w}{nr} + \frac{2ml^2 r + Ir}{4l^2 n} \right) \\ K_2 &= \left(\frac{2ml^2 r - Ir}{4l^2 n} \right) \\ K_3 &= \left(\frac{n^2 f_m + f_w}{nr} + \frac{2l^2 k_v r + k_\omega r}{4l^2 n} + \frac{k_m k_E n}{R_a r} \right) \\ K_4 &= \left(\frac{2l^2 k_v r - k_\omega r}{4l^2 n} \right) \end{aligned}$$

II.7. Simulation

Les deux modèles du robot ont été construits sur l'environnement Simulink MATLAB.

La cinématique du robot est donnée par la Figure II.5.

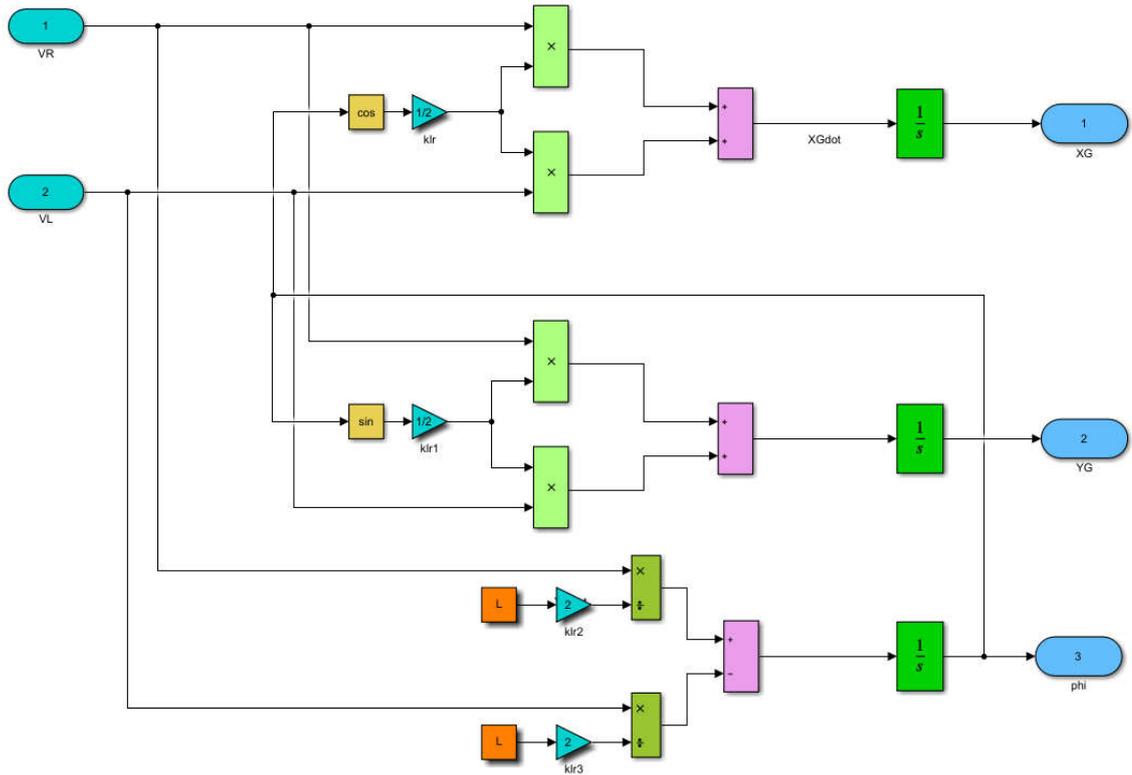


Figure II.5. La cinématique du robot sur SIMULINK

La dynamique du robot est donnée dans la Figure II.6

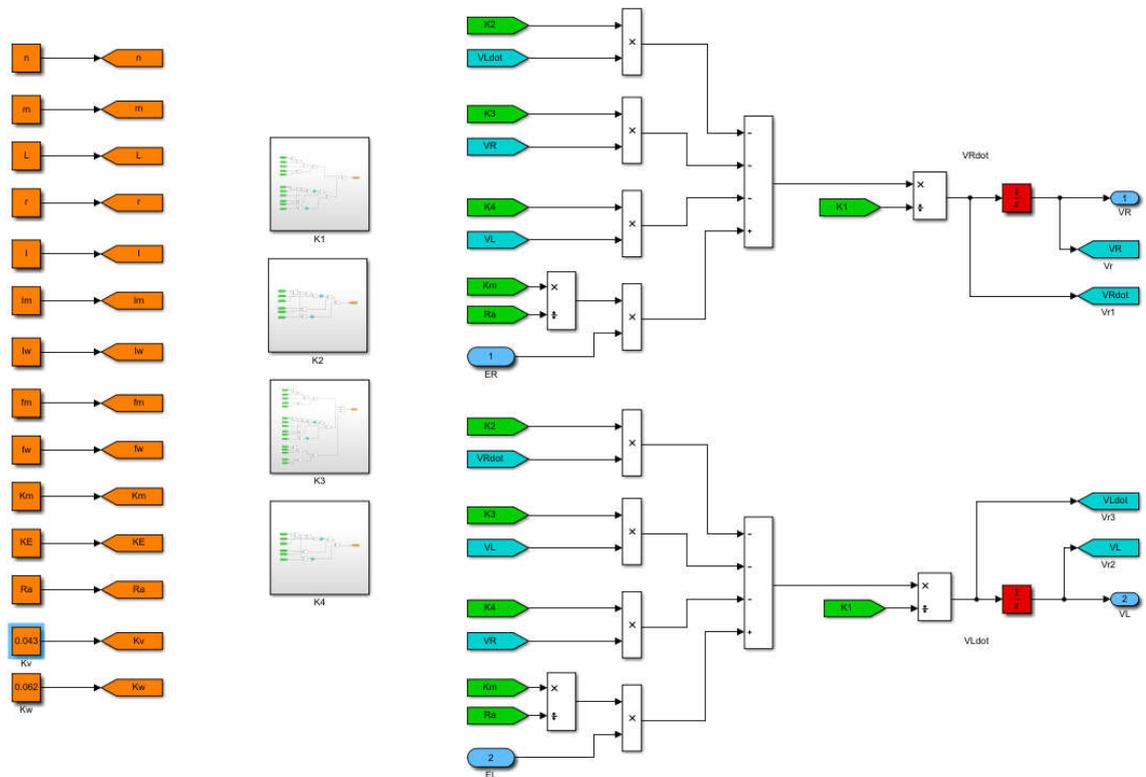


Figure II.6. La dynamique du robot sur SIMULINK

▪ **Réponse indicielle**

Pour tracer la réponse indicielle du système à un échelon, on applique aux deux entrées du modèle cinématique et dynamique deux échelons, et on tracera les sorties. Les paramètres du robot sont donnés dans la Table II.1

Table II.1. Les paramètres du robot

n	m	L	r	I	I_m	I_w	f_m	f_w	k_m	k_E	R_a	k_v	k_w
10	0.8	0.1	0.04	4.6e-3	2.7e-7	8e-5	5e-8	9e-8	0.0144	0.0145	1	0.043	0.062

▪ **Réponse indicielle du modèle cinématique**

Appliquons aux deux entrées du modèle cinématique différents types d'entrées comme il est indiqué à la Figure II.7. Le robot est initialement au point $(x_G, y_G, \varphi) = (0m, 0m, 0^\circ)$.

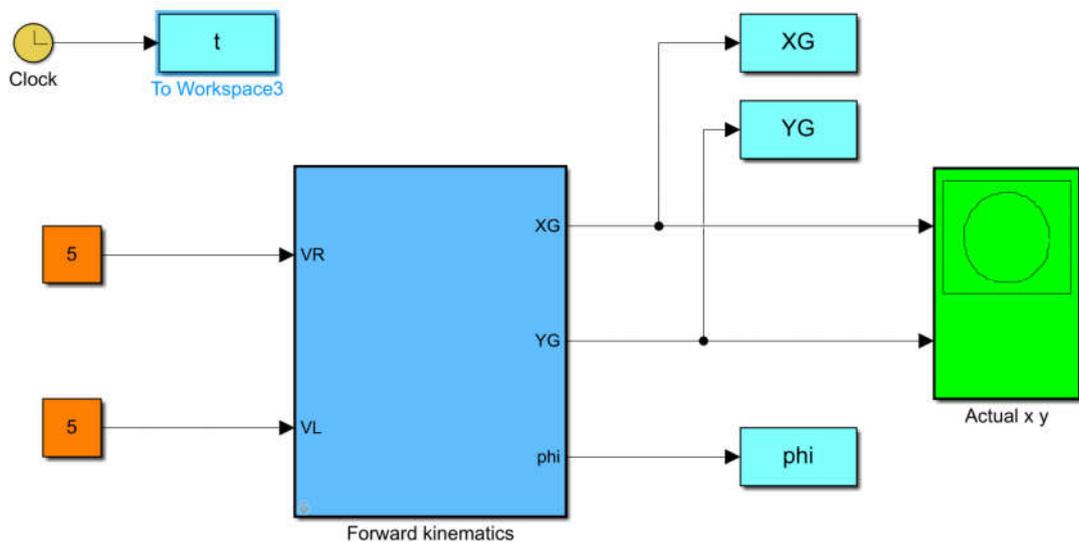


Figure II.7. Modèle de la réponse indicielle de la cinématique du robot sur SIMULINK

On applique la même valeur de vitesse aux deux entrées du robot, les résultats obtenus sont donnés dans la Figure II.8. Le robot marche en ligne droite sans changer son orientation initiale $\varphi_0 = 0$.

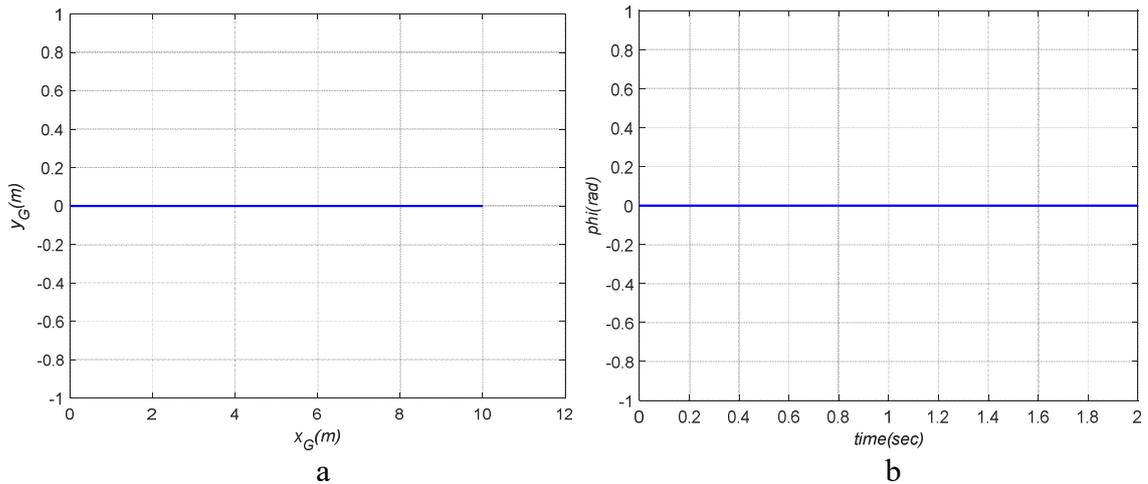


Figure II.8. La réponse indicielle avec $(x_{G0}, y_{G0}, \varphi_0) = (0\text{m}, 0\text{m}, 0^\circ)$ et $(V_R = 1\text{ m/s et } V_L = 1\text{ m/s})$: a) trajectoire, b) orientation

Maintenant, la configuration initiale du robot est $(x_{G0}, y_{G0}, \varphi_0) = (0\text{m}, 0\text{m}, 60^\circ)$. Le robot marche également en ligne droite, en gardant l'orientation initiale $\varphi_0 = 60^\circ = \left(\frac{\pi}{3}\text{ rad}\right)$ (Figure II.9).

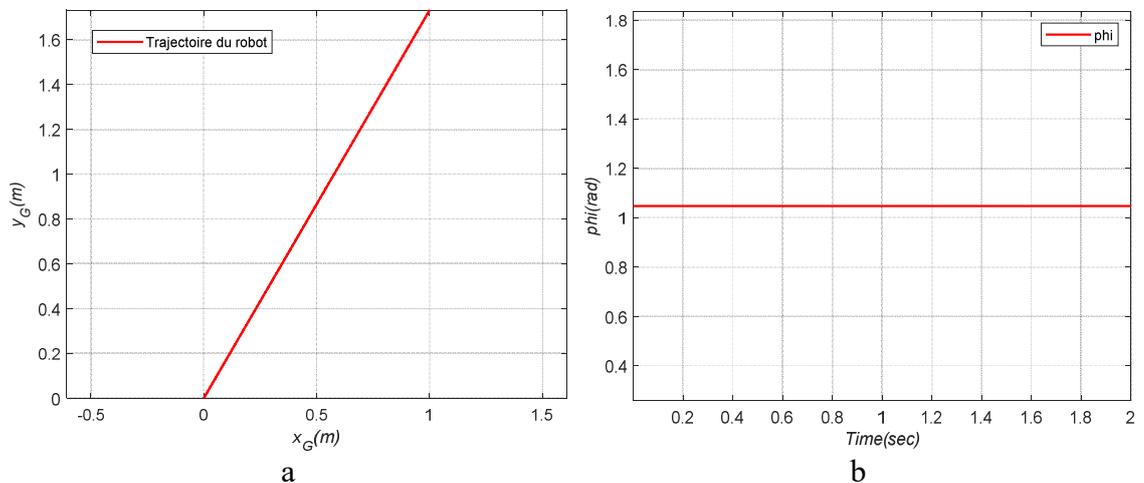


Figure II.9. La réponse indicielle avec $(x_{G0}, y_{G0}, \varphi_0) = \left(0\text{m}, 0\text{m}, \frac{\pi}{3}\text{ rad}\right)$ et $(V_R = 1\text{ m/s et } V_L = 1\text{ m/s})$: a) trajectoire, b) orientation

Si nous appliquons des valeurs différentes aux entrées du modèle, le robot dessinera un cercle et l'orientation augmentera linéairement comme indiqué sur la Figure II.10.

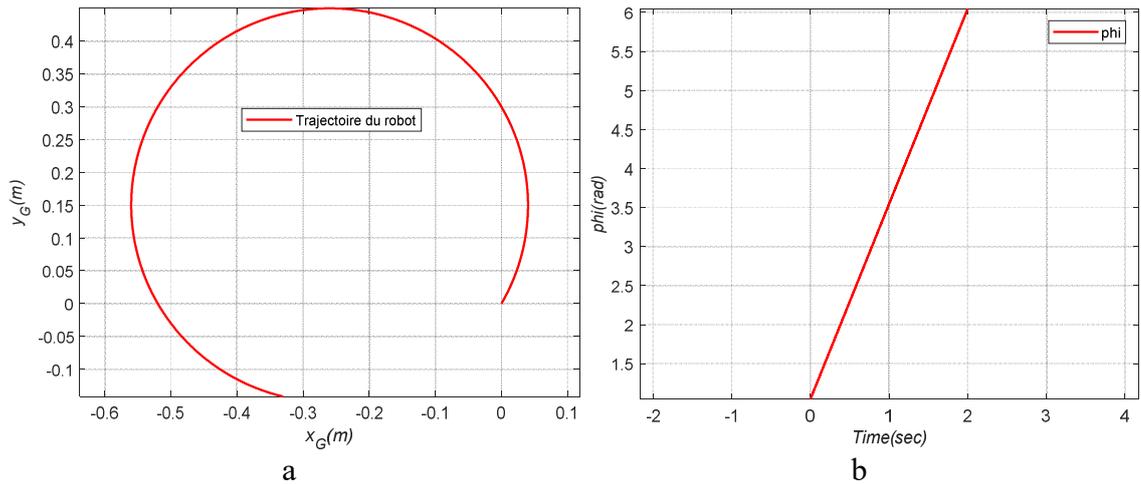


Figure II.10. La réponse indicielle avec $(x_{G0}, y_{G0}, \varphi_0) = (0m, 0m, \frac{\pi}{3} rad)$ et $(V_R = 1 m/s$ et $V_L = 0.5 m/s)$: a) trajectoire, b) orientation

▪ **Réponse indicielle du modèle dynamique**

De la même manière, différents types d'entrées sont appliqués au système dynamique pour le tester (Figure II.11).

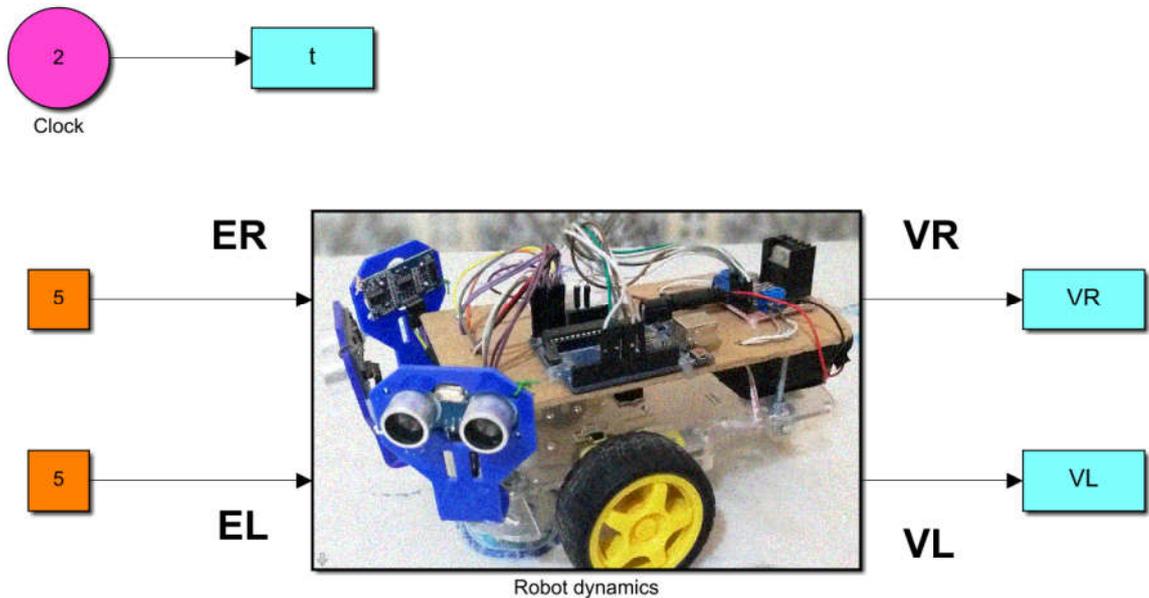


Figure II.11. Modèle de la réponse indicielle de la dynamique du robot sur SIMULINK

Les résultats sont données dans les figures II.12 et II.13.

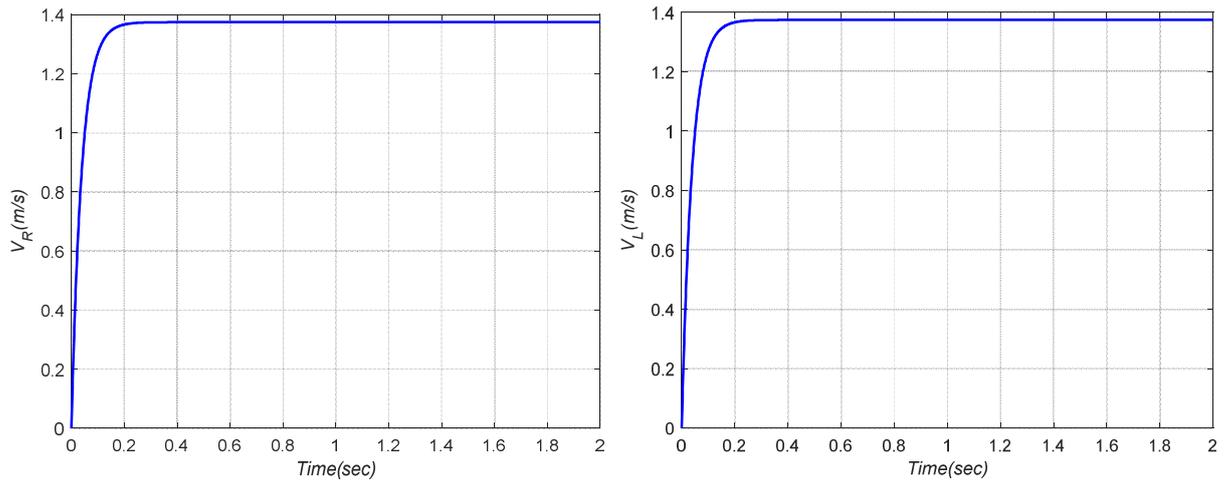


Figure II.12. La réponse indicielle ($E_R = 5\text{ V}$ et $E_L = 5\text{ V}$)

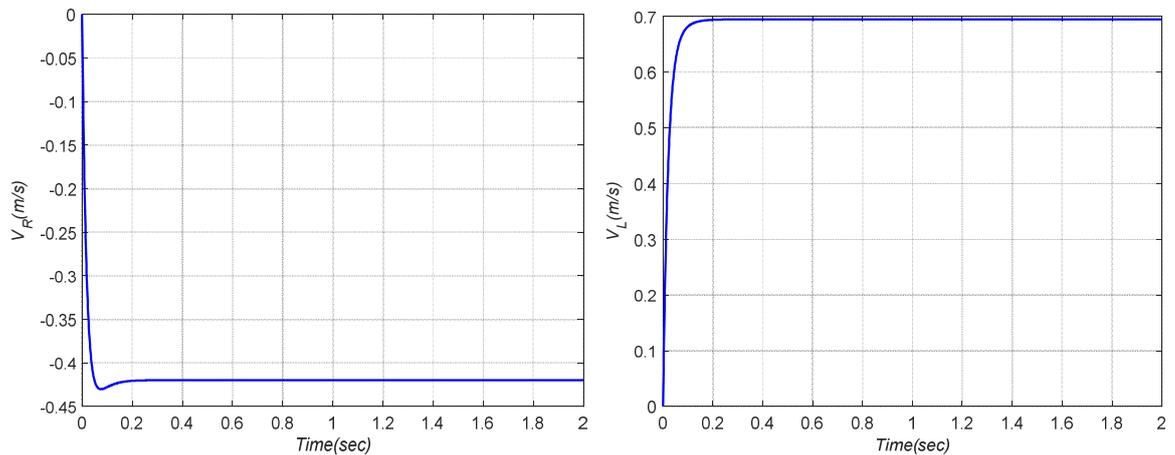


Figure II.13. La réponse indicielle ($E_R = -2\text{ V}$ et $E_L = 3\text{ V}$)

- **Réponse indicielle du modèle cinématique avec le modèle dynamique**

De la même manière, différents types d'entrées sont également appliqués aux deux modèles simultanément: le modèle cinématique et le modèle dynamique (Figure II.14). Les résultats sont présentés à la Figure II.15 et II.16.

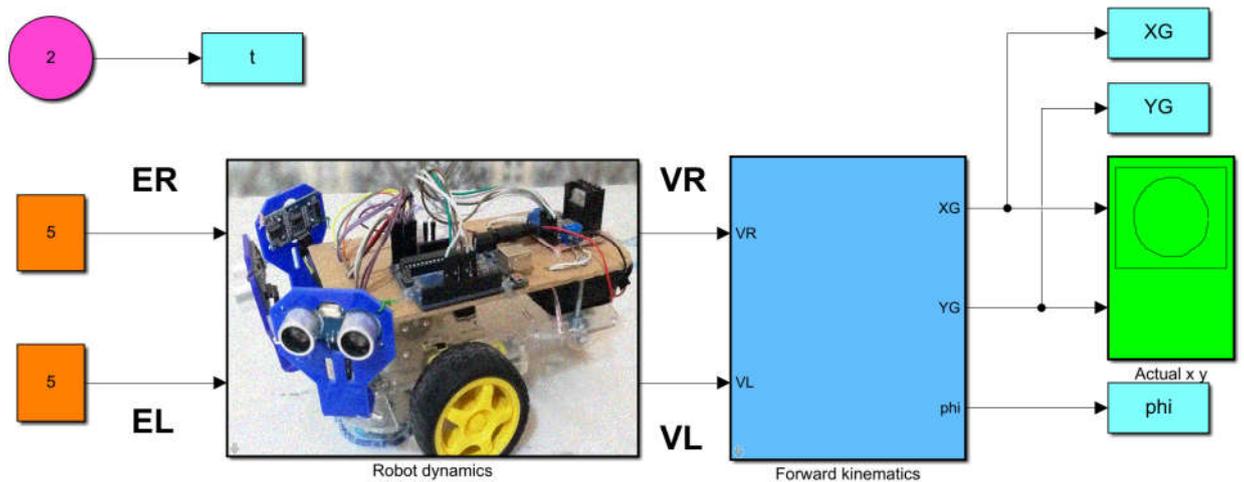


Figure II.14. Modèle de la réponse indicielle de la dynamique avec la cinématique du robot sur SIMULINK

Dans le premier scénario, la même valeur est appliquée aux deux moteurs ($E_R = 5V$ et $E_L = 5V$), dont la configuration initiale du robot est $(x_{G0}, y_{G0}, \varphi_0) = (0\text{ m}, 0\text{ m}, 45^\circ)$.

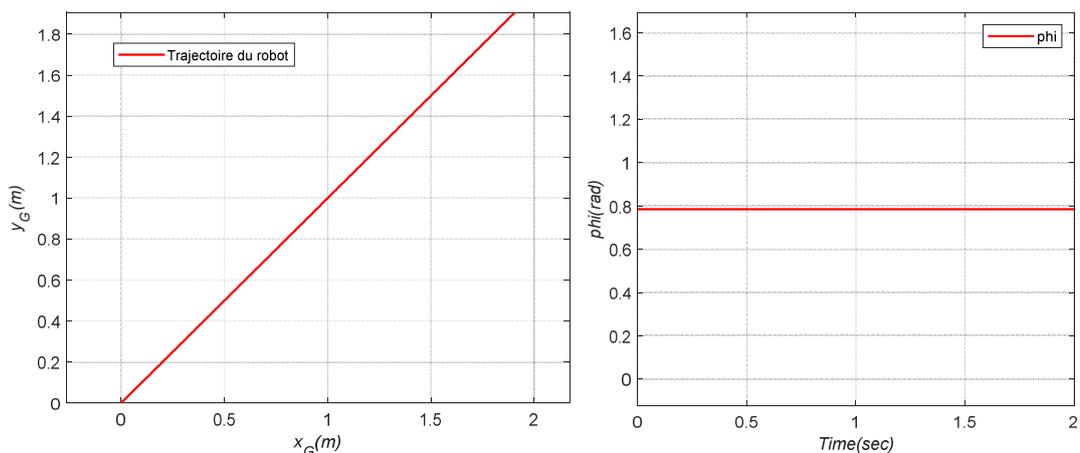


Figure II.15. La réponse indicielle ($E_R = 5V$ et $E_L = 5V$)

Alors, le robot dessine une ligne droite avec l'angle de 45° .

Dans le deuxième scénario, deux valeurs différentes ($E_R = 5V$ et $E_L = 2V$) sont appliquées aux moteurs. Le robot dessine un cercle partant du point $(0,0)$ et se tournant vers la gauche, comme illustré dans la Figure II.16.

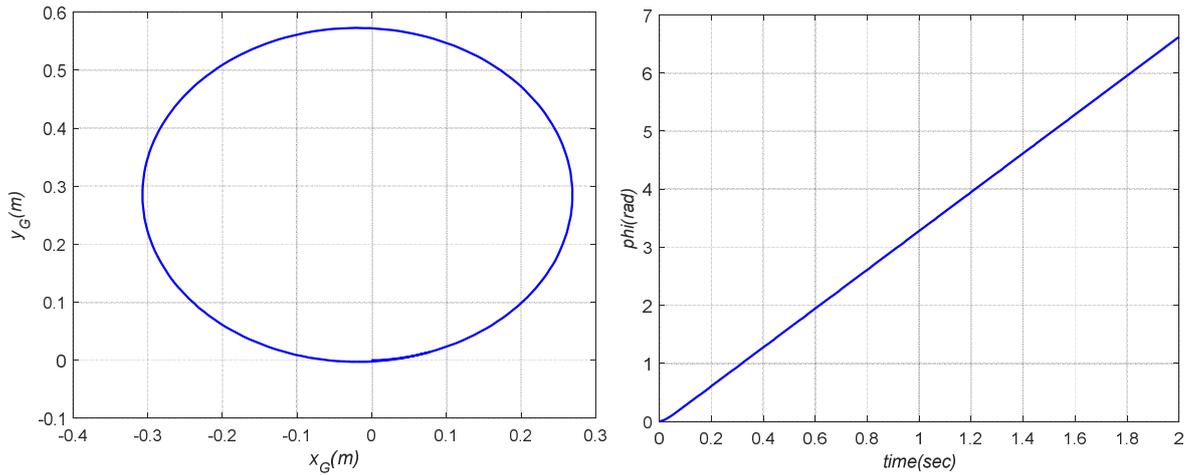


Figure II.16. La réponse indicielle ($E_R = 5\text{ V}$ et $E_L = 2\text{ V}$)

Les résultats de simulation précédents montrent que les réponses des modèles cinématiques et dynamiques sont conformes aux attentes. Ainsi, ce modèle donne une représentation réelle du robot.

II.8. Conclusion:

La modélisation est très importante dans l'étude du mouvement des robots. Comme nous l'avons défini, la modélisation cinématique est l'étude du mouvement des systèmes mécaniques sans prendre en compte les forces qui affectent le mouvement; et la modélisation dynamique est l'étude du mouvement du système mécanique, en tenant compte des forces affectant le mouvement.

Dans ce chapitre, nous nous sommes concentrés sur le calcul du modèle cinématique et du modèle dynamique d'un robot à entraînement différentiel, et nous avons vérifié la validité de des modèles par simulation à l'aide du Simulink Matlab.

Dans le prochain chapitre, nous proposerons un système de contrôle basé sur la logique floue pour le problème de suivi de trajectoire.

Chapitre III:

**Problème de suivi de
trajectoire**

Chapitre III

Problème de suivi de trajectoire du robot à entraînement différentiel

III.1. Introduction:

Après avoir établi la cinématique et la dynamique du robot, nous passons à résoudre le problème du suivi de trajectoire, où le robot doit suivre une trajectoire désirée avec une grande précision.

Généralement, nous ne devons pas utiliser l'intelligence artificielle pour contrôler un système donné avant de tester les choix les plus simples comme le contrôle PID. Mais, malheureusement, après de nombreuses tentatives, les contrôleurs PID n'ont pas réussi à suivre le chemin souhaité. En général, le contrôleur PID est utilisé pour contrôler les systèmes mono variables, le robot est un système multivariable ayant deux équations de mouvement couplées non linéaires, nous ne pouvons donc pas contrôler chaque entrée séparément avec un PID, car chaque sortie est couplée à la seconde. Et aussi, le degré élevé de non-linéarité nécessite un contrôleur qui traite cette non-linéarité comme le contrôle flou. Un contrôleur flou peut contrôler un système multivariable et gère à la fois la non-linéarité et l'incertitude. C'est donc un choix approprié pour notre problème.

III.2. Le système de suivi de trajectoires

Figure III.1 montre la structure de base du système de contrôle du robot mobile pour la poursuite de trajectoires. A partir d'une configuration initiale (position et orientation), le robot doit suivre une trajectoire désirée.

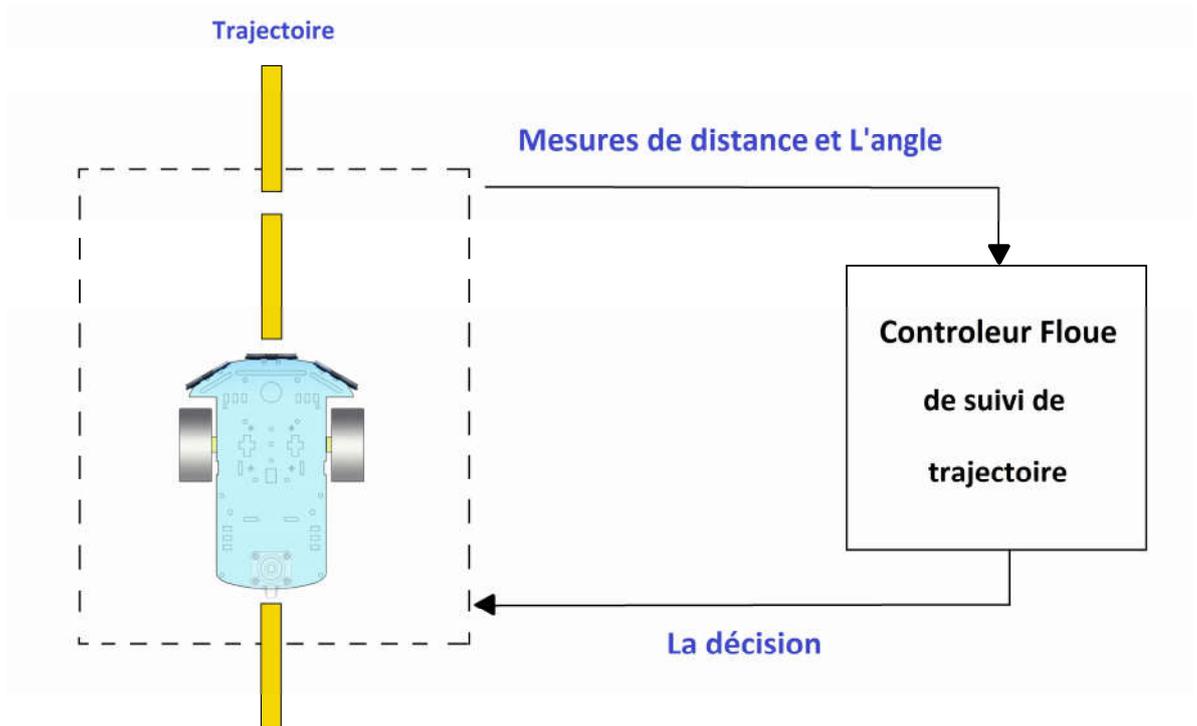


Figure III.1.Schéma de base du système de suivi de trajectoires

III.3. Les systèmes flous

La théorie des systèmes flous est inspirée par la remarquable capacité humaine à raisonner avec des informations basées sur la perception. Le système de logique floue conventionnel comprend principalement la fuzzification, la base des règles, le mécanisme d'inférence et la défuzzification. La fuzzification convertit les variables d'entrée précises en notes d'entrée nommées comme variables floues (linguistiques). La base des règles est utilisée pour stocker les données pertinentes et les règles de contrôle.

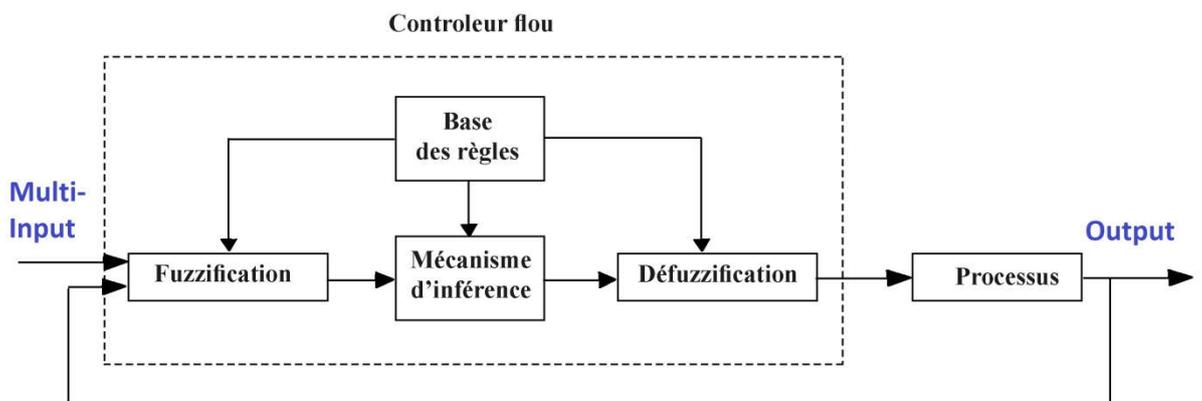


Figure III.2.Structure générale d'un contrôleur flou

Le raisonnement flou génère des résultats flous à partir de la base des règles et du mécanisme d'inférence. La défuzzification convertit les variables floues (linguistiques) aux variables de sortie précises (numériques). Nous pouvons expliquer dans la structure suivant [21].

Donc, le contrôle de la logique floue est basé sur quatre blocs principaux, nous devons connaître la fonction de chaque bloc.

III.3.1. La fuzzification:

Dans cette partie, les variables numériques non floues des entrées sont converties en variables de linguistiques floues. La fuzzification consiste à définir les fonctions d'appartenance pour les différentes variables d'entrées et de sortie. Dans le cas de réglage par la logique floue, on utilise en général des formes trapézoïdales et triangulaires pour les fonctions d'appartenances [22].

III.3.2. La base des règles:

Une base de règles floues lie les variables d'entrées transformées en variables linguistiques grâce à la fuzzification à la variable de sortie fuzzifiée. Ces règles peuvent être utilisées en parallèle ou enchaînées dans certaines applications. D'une manière explicite, l'inférence est décrite à l'aide d'un certains nombres de règles [23]. L'élaboration de règles pour lier les valeurs linguistiques est basée sur une base de connaissances dérivée de l'expérience humaine.

Les règles peuvent être exprimées sous la forme générale par exemple:

Exemple: Si < prédicat 1 > Alors < Opération 1 >, Ou

III.3.3. Le mécanisme d'inférence:

Le mécanisme d'inférence le plus utilisé est celui de Mamdani. La méthode d'inférence Mamdani, utilise l'opérateur «ET» par la formulation du minimum. La conclusion dans chaque règle, introduite par «ALORS», qui est réalisée par la formation du minimum. Enfin l'opérateur «OU» lie les différentes règles, réalisé par la formation du maximum [22][23].

III.3.4. La défuzzification:

Ce bloc est l'inverse de celui de fuzzification, il sert à transformer la décision floue à une valeur numérique afin de l'envoyer au système [23].

III.4. Conception du contrôleur flou

Dans le contrôle automatique robotique, les contrôleurs sont des unités de traitement séquentielles où l'architecture fonctionne à travers un cycle de Sense-Plan-Action [24]. Le problème de suivi de trajectoires est illustré dans la Figure III.3. À tout instant, le robot doit suivre le point actuel désiré, il doit donc minimiser l'erreur de distance Dis et l'erreur d'orientation β .

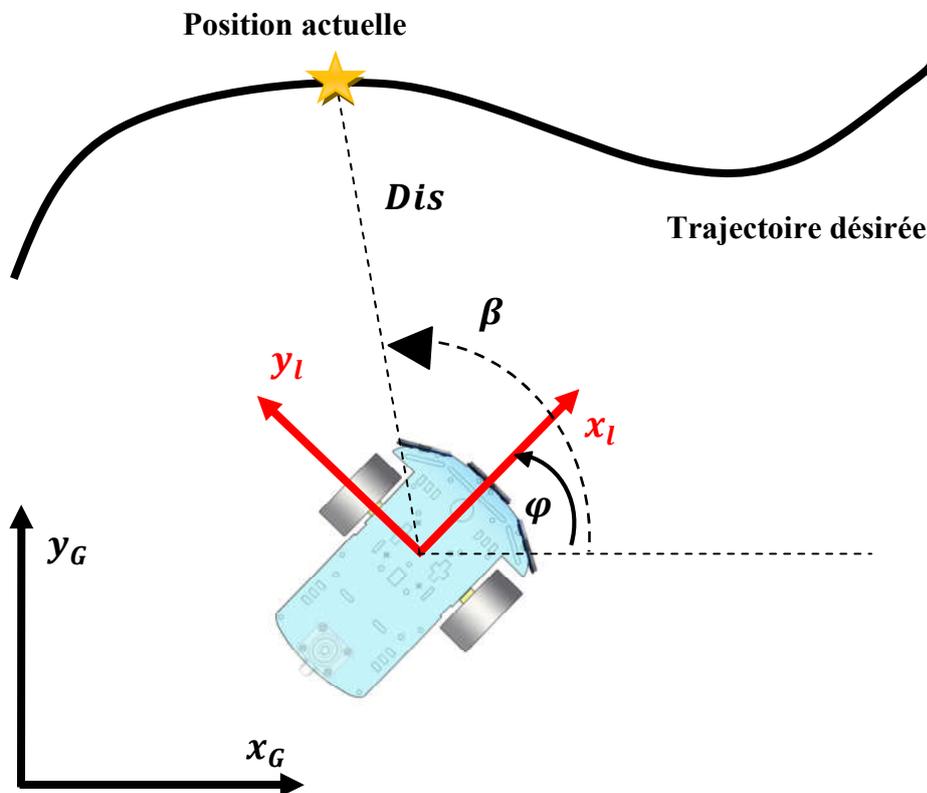


Figure III.3 Le problème de suivi de trajectoire

Dans la théorie de contrôle flou, les commandes inférées sont traités en trois étapes: fuzzification, inférence et défuzzification. Les règles linguistiques peuvent être par exemple:

- R1: Si le but est Loin Alors la vitesse est Grande et le braquage est Zéro,
- R2: Si l'obstacle est à droite Alors la vitesse est Moyenne et le braquage est à Gauche,
- ...
- Rn: Si le but est Proche Alors la vitesse est Faible et le braquage est Zéro.

Le contrôleur flou possède deux entrées et deux sorties comme illustré à la Figure III.4.

Les entrées sont :

- La distance Dis entre le robot et le point actuel de la trajectoire.
- l'angle d'orientation (the bearing angle) du robot par rapport au point actuel désiré.

Les sorties sont :

- La tension du moteur droit V_R
- La tension du moteur gauche V_L

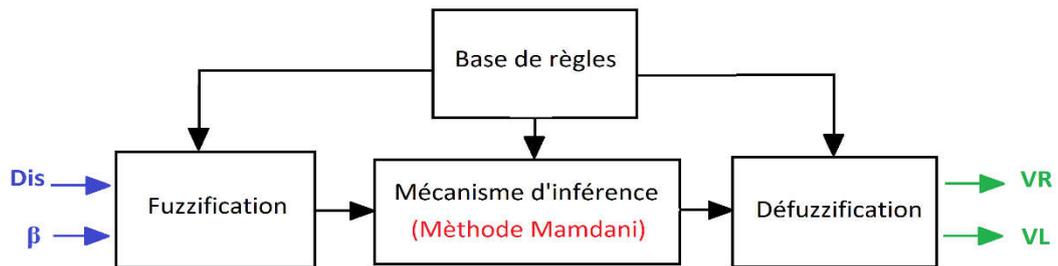


Figure III.4. Architecture du contrôleur flou proposé

III.4.1. La fuzzification

Deux types de fonctions d'appartenance triangulaires et trapézoïdales sont considérées.

Les variables linguistiques sont données dans la Table III.1:

Tableau III.1. Les variables linguistiques pour les entrées et les sorties

Les Variables	La distance Dis (m)	Z	N	F
		Zero	Near	Far
	L'angle β (rad)	N	Z	P
		Negative	Zero	Positive
	La vitesse V_R, V_L (m/s)	N	Z	P
		Negative	Zero	Positive

Les fonctions d'appartenance des entrées du contrôleur sont illustrées à la Figure III.5, tandis que celles des sorties sont indiquées à la Figure III.6.

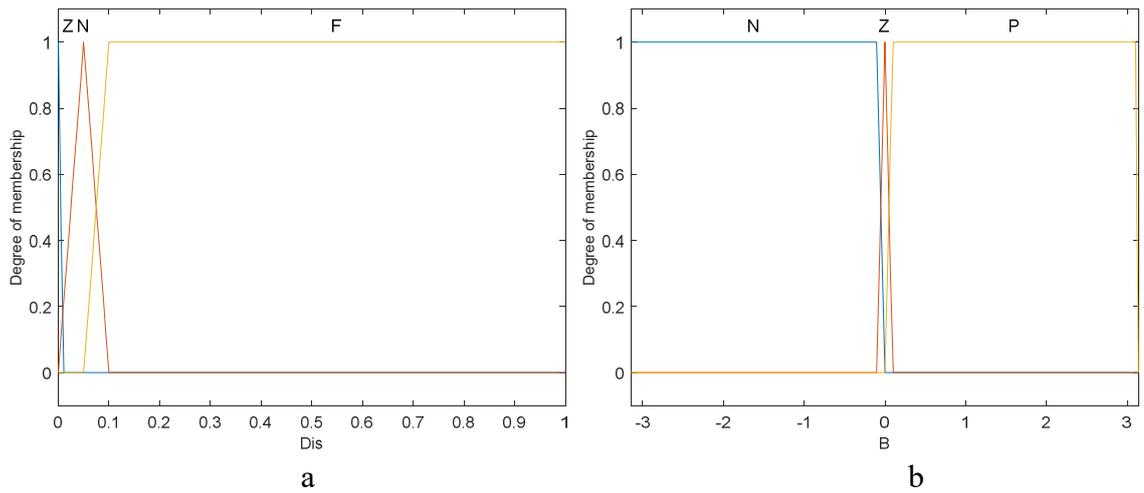


Figure III.5. Les fonctions d'appartenance des entrées Dis (la distance) et β (l'angle)

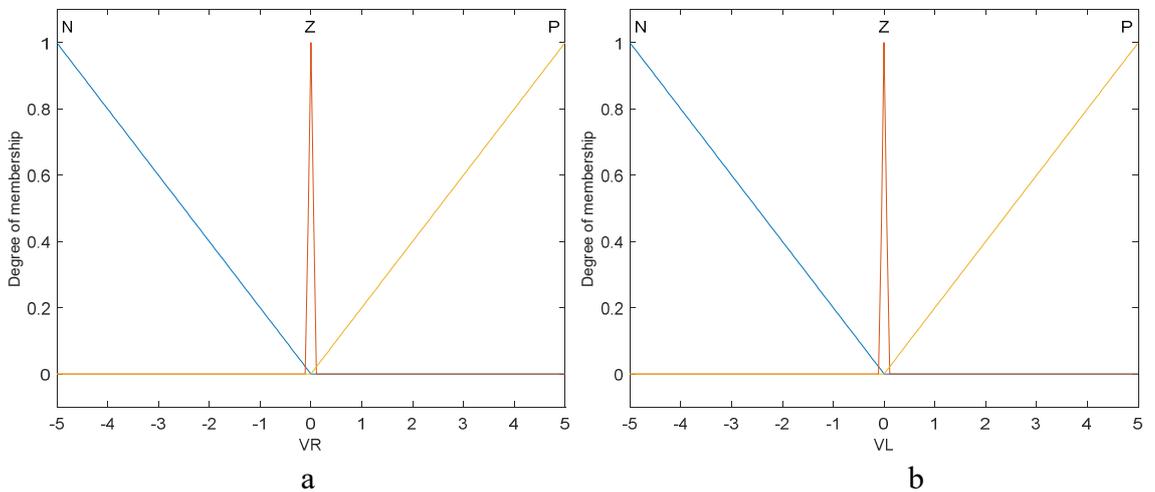


Figure III.6. Les fonctions d'appartenance des sorties V_R et V_L

III.4.2. La base de règles floues

Après l'étape de construction des fonctions d'appartenance, nous construisons la table des règles floues, qui est présentée dans la Table III.2.

Les règles floues IF-THEN ont été formulées sur la base de l'expertise humaine.

Tableau III.2.Règles floues du contrôleur proposé

		Les variables			
		Les entrées		Les sorties	
		Dis	β	VR	VL
	1	Z	N	Z	Z
	2	Z	Z	Z	Z
Les règles (9 règles)	3	Z	P	Z	Z
	4	N	N	N	P
	5	N	Z	P	P
	6	N	P	P	N
	7	F	N	N	P
	8	F	Z	P	P
	9	F	P	P	N

Comme nous avons 3 fonctions d'appartenance pour chaque entrée, nous avons donc (3x3 = 9) règles possibles. Les 9 règles sont données ci-dessous:

1. If (Dis is Z) and (B is N) then (VR is Z) (VL is Z)
2. If (Dis is Z) and (B is Z) then (VR is Z) (VL is Z)
3. If (Dis is Z) and (B is P) then (VR is Z) (VL is Z)
4. If (Dis is N) and (B is N) then (VR is N) (VL is P)
5. If (Dis is N) and (B is Z) then (VR is P) (VL is P)
6. If (Dis is N) and (B is P) then (VR is P) (VL is N)
7. If (Dis is F) and (B is N) then (VR is N) (VL is P)
8. If (Dis is F) and (B is Z) then (VR is P) (VL is P)
9. If (Dis is F) and (B is P) then (VR is P) (VL is N)

III.5. Résultats de simulation et discussion

MATLAB Fuzzy Logic Toolbox est utilisé pour concevoir un contrôleur flou (FLC). La boîte à outils contient des fonctions, des interfaces utilisateur graphiques et des structures de données qui permettent à l'utilisateur de concevoir, tester, simuler et modifier rapidement un système d'inférence floue.

Le système du robot mobile est défini par le modèle cinématique et dynamique. Les valeurs d'orientation et de position du robot sont calculées et utilisées comme signaux de retour pour l'ensemble du système.

Les entrées de contrôleur flou sont la distance entre le robot et le point désiré Dis , et l'angle d'orientation relative du robot par rapport au point désiré β .

$$Dis = \sqrt{(y_{traj} - y_{robot})^2 + (x_{traj} - x_{robot})^2} \quad (III.1)$$

$$\beta = \tan^{-1} \left(\frac{y_{traj} - y_{robot}}{x_{traj} - x_{robot}} \right) \quad (III.2)$$

Le système général de contrôle est illustré dans la Figure III.8

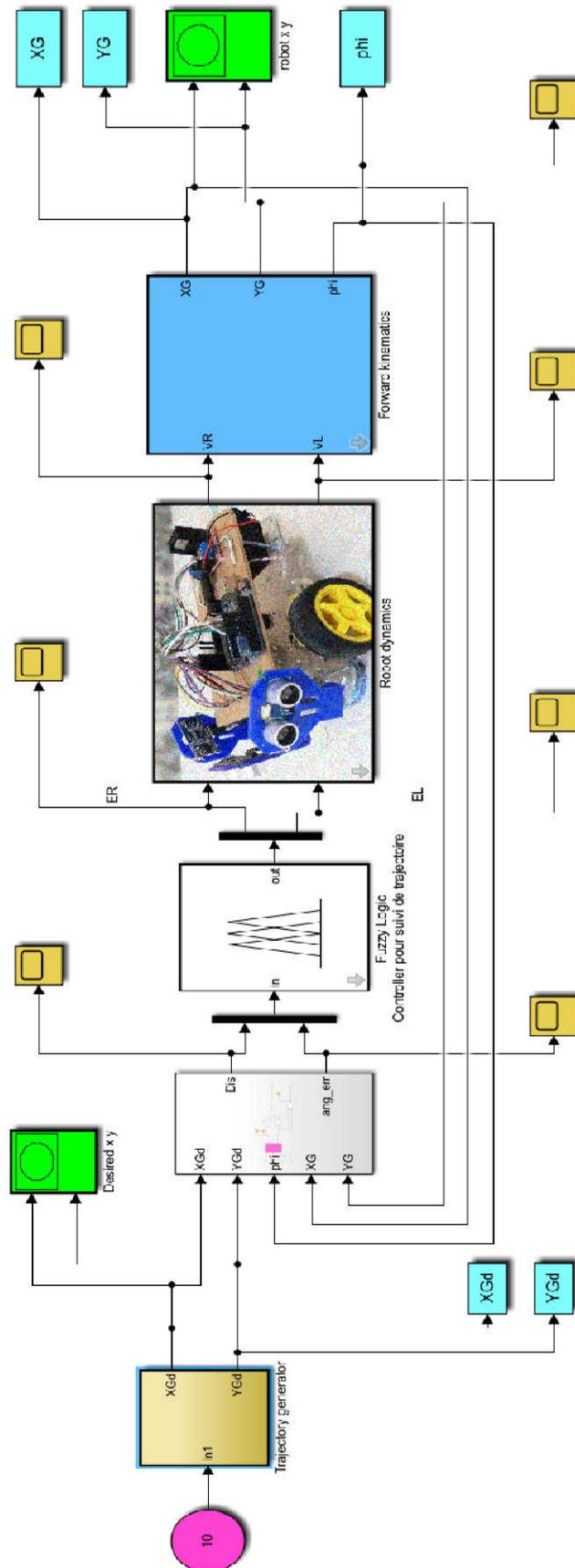


Figure III.7. Système générale de contrôle pour le problème du suivi de trajectoires

Pour vérifier l'efficacité du contrôleur flou dans le problème de suivi de trajectoire, trois trajectoires ont été testées: une trajectoire circulaire, une trajectoire en forme de huit et une trajectoire en forme de filet.

- **Trajectoire circulaire**

Le robot doit suivre une trajectoire circulaire qui a l'origine (0,0) et de rayon de 0,3 m comme illustré sur la Figure III.9. Les erreurs sur la distance entre le robot et le point désiré et l'angle d'orientation sont données dans la Figure III.10. Figure III.11 montre la commande appliquée aux deux moteurs. Le robot est initialement au point (0 m,0 m, 0 rad).

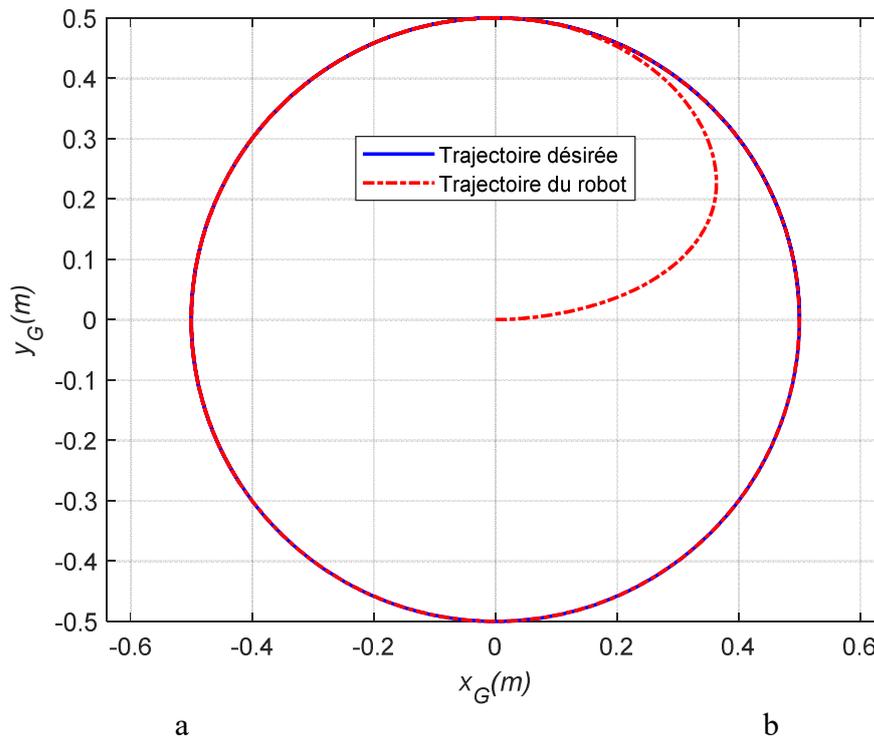


Figure III.8.Suivre une trajectoire circulaire

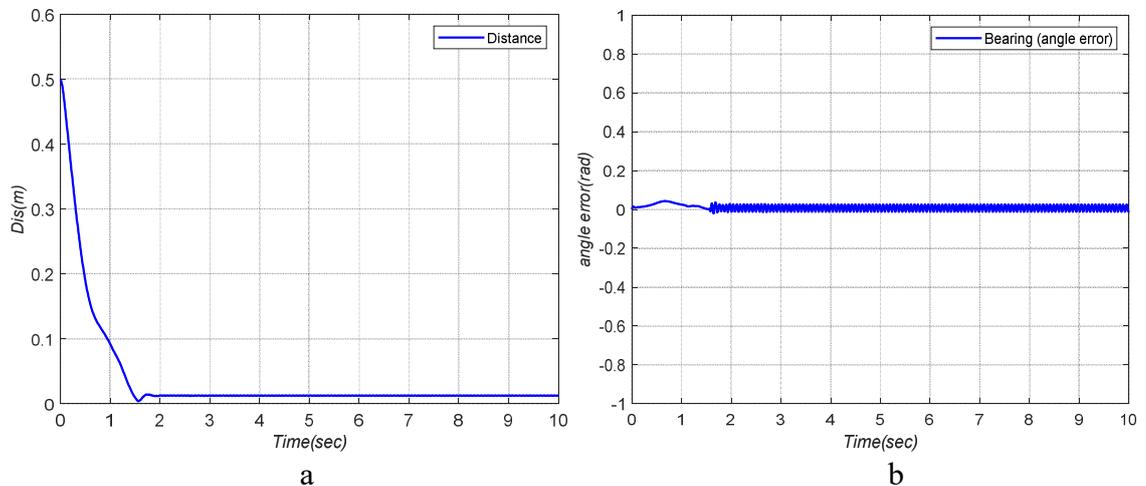


Figure III.9.Suivre une trajectoire circulaire, l'erreur sur la: a) distance (Dis), b) orientation

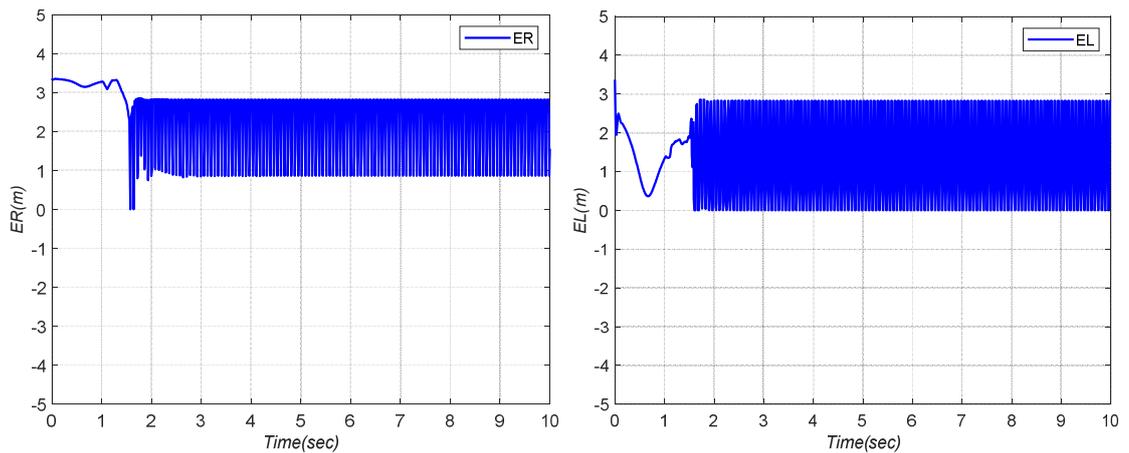


Figure III.10. La commande appliqué pour la trajectoire circulaire : a) Tension du moteur droit E_R , b) Tension du moteur E_L

- **Trajectoire en forme de huit**

Le robot doit suivre une trajectoire en forme de huit (Figure III.12). Les erreurs sur la distance entre le robot et le point désiré et l'angle d'orientation sont données dans la Figure III.13. Figure III.14 montre la commande appliquée aux deux moteurs. Le robot est initialement au point (0.3 m, 0 m, 0 rad).

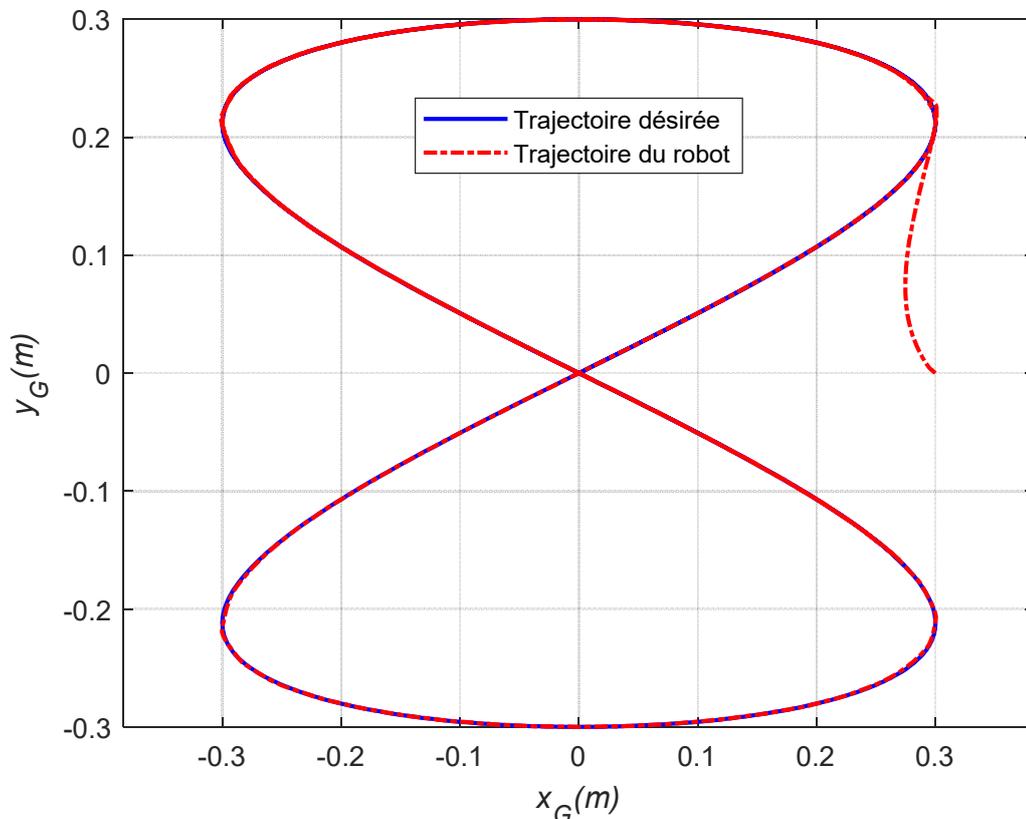


Figure III.11. Suivre une trajectoire en forme de huit

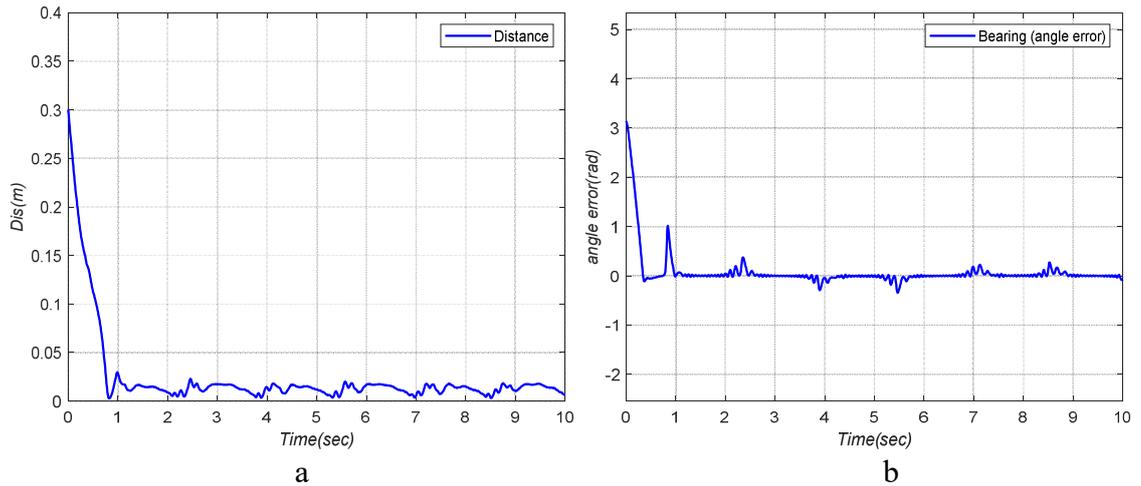


Figure III.12.L'erreur du suivi de trajectoire en forme de huit : a) Distance, b) Bearing

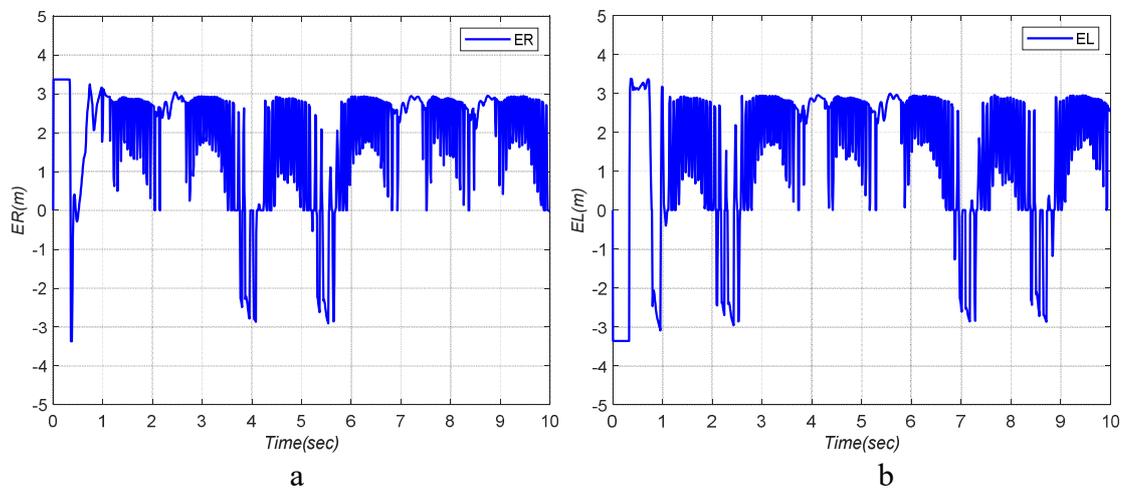


Figure III.13.La commande appliquée pour la trajectoire en forme de huit: a) Tension du moteur droit E_R , b) Tension du moteur gauche E_L

- **Trajectoire en forme de filet :**

Dans ce scénario, le robot doit suivre une trajectoire en forme de filet (Figure III.15). Les erreurs sur la distance entre le robot et le point désiré et l'angle d'orientation sont données dans la Figure III.16. Figure III.17 montre la commande appliquée aux deux moteurs. Le robot est initialement au point (0 m,0 m, 0 rad).

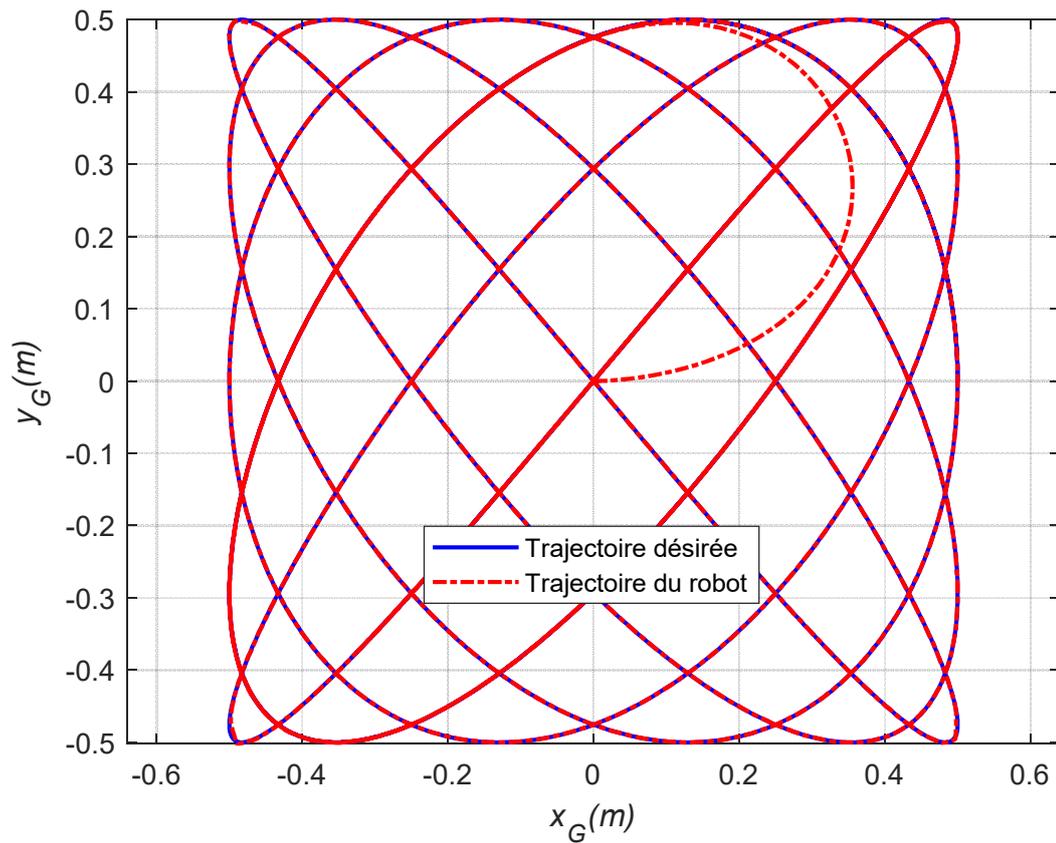


Figure III.14. Suivre une trajectoire en forme de filet

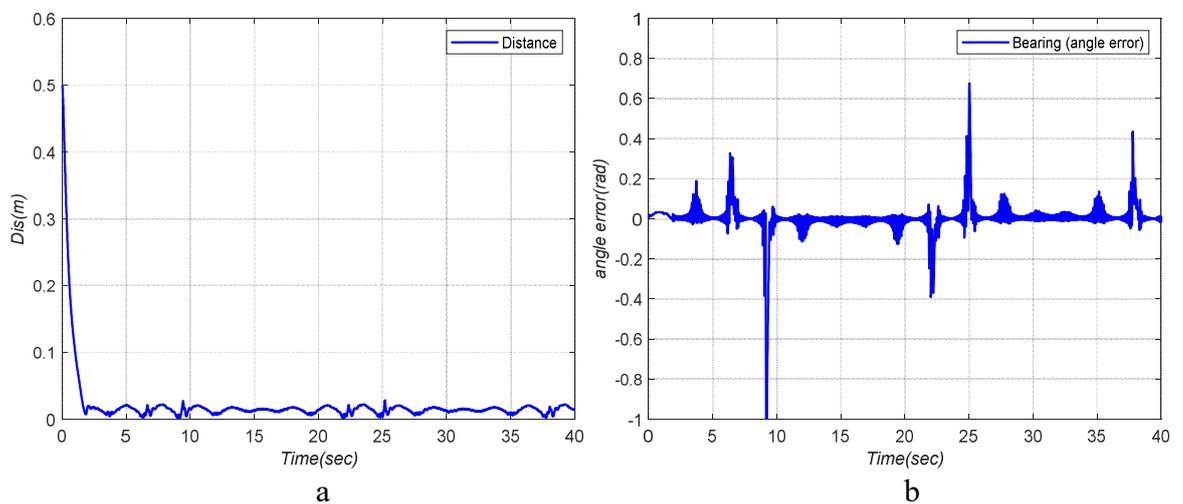


Figure III.15. L'erreur du suivi d'une trajectoire en forme de filet: a) Distance, b) Bearing

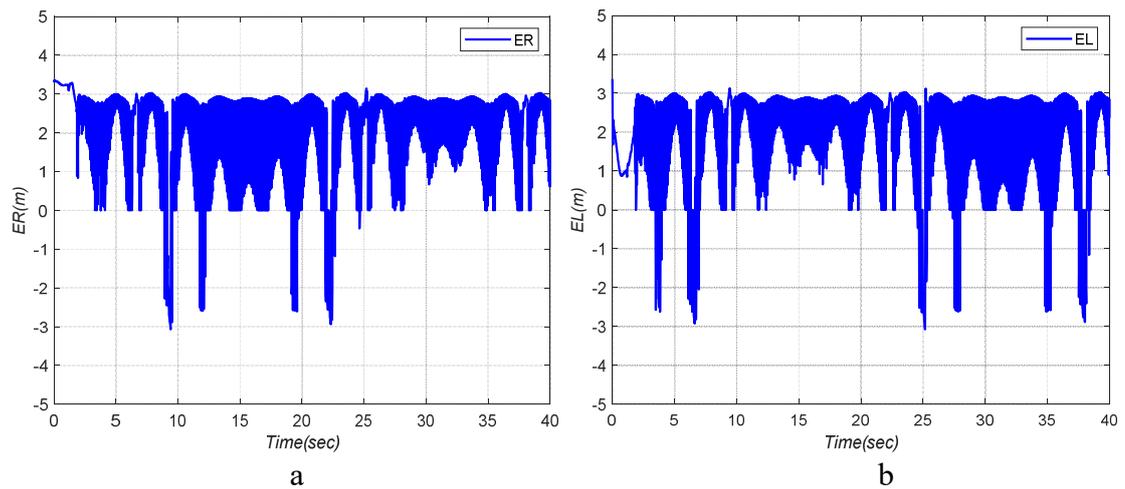


Figure III.16. La commande appliquée sur le robot : a) Tension du moteur droit E_R , b) Tension du moteur gauche E_L

Vous pouvez facilement voir que le robot peut suivre les trajectoires souhaitées avec précision. L'erreur sur la distance est de l'ordre de 2 cm, alors que l'erreur sur l'angle d'orientation (Bearing) est de l'ordre de 1,5 degré pour la trajectoire circulaire. Quand la trajectoire devient plus complexe, nous pouvons voir des impulsions ou des pics, soit dans l'erreur de distance, soit dans l'erreur d'orientation. Et cela est dû aux coins des trajectoires dont la trajectoire changera rapidement sa courbure; ainsi qu'à la base des règles utilisées. Donc, si nous prenons par exemple la règle 4: si (Dis est N) et (B est N) alors (VR est N) (VL est P), lorsque l'orientation est négative, alors le robot tourne rapidement à gauche.

Pour obtenir des courbes plus lisses, plus des règles floues peuvent être utilisées, donc le nombre de règles floues sera augmenté.

III.6. Conclusion :

Dans ce chapitre, nous avons proposé une solution pour la navigation d'un robot mobile en utilisant le contrôle flou.

Le choix le plus simple consiste à utiliser un contrôleur PID. Mais malheureusement, le PID n'a pas réussi à suivre le chemin souhaité après de nombreuses tentatives. De plus, le contrôleur PID est utilisé pour contrôler des systèmes monovariables, la dynamique du robot est un système multivariable ayant deux équations de mouvement couplées non linéaires, alors, nous ne pouvons pas contrôler chaque entrée séparément avec un PID, car chaque sortie est couplée à la seconde. De plus, le degré élevé de non-linéarité nécessite un contrôleur qui traite cette non-linéarité comme le contrôle flou, qui a été utilisé pour le faire.

Les résultats obtenus montrent l'efficacité du contrôleur, qui a permis au robot de suivre la trajectoire souhaitée avec une grande précision.

Chapitre IV:

Réalisation du robot différentiel

Chapitre IV

Réalisation du robot différentiel

IV.1. Introduction

Comme nous l'avons dit plus tôt dans le premier chapitre que la robotique regroupe plusieurs sciences mécaniques, électroniques et de programmation, nous le verrons à travers ce chapitre dans lequel nous allons réaliser un réel robot mobile à entraînement différentiel et cela utilise les connaissances que nous avons acquises grâce à la recherche dans cette mémoire.

Ce chapitre présente la conception matérielle (Hardware) et logicielle (Software) du robot mobile. Il couvre les composants nécessaires à la construction du robot mobile à entraînement différentiel.

IV.2. Côté Hardware:

Les robots sont également des machines, ainsi que du matériel. Selon l'Oxford Dictionary, une machine est un appareil utilisant ou appliquant une puissance mécanique, et ayant plusieurs pièces (matériels) ayant chacune une fonction définie et exécutant ensemble une tâche particulière [14]. Nous découvrirons les différents équipements que nous utiliserons.

IV.2.1. La plate-forme:

C'est la partie principale du corps et de la structure du robot, portable sur deux roues motrices et une roue libre pour l'équilibre du robot (Figure IV.1).

La plate-forme est conçue pour transporter tous les autres composants du robot, tels que les articles électroniques, les marchandises (si le robot est spécial pour le transport), etc. Elle doit être adaptée pour accueillir toutes les composants nécessaires au robot en termes de taille et de poids, ainsi que pour être adaptée aux vibrations et aux chocs auxquels il est exposé dans son environnement de travail.

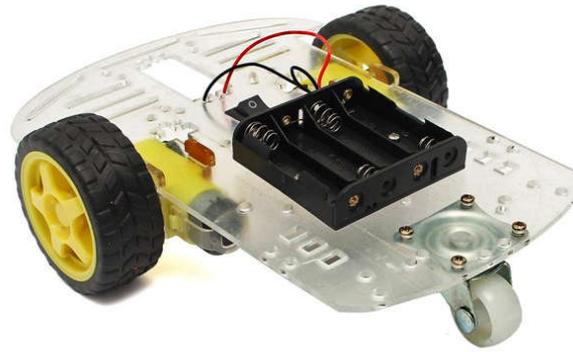


Figure IV.1. La plate-forme du robot à entraînement différentiel

IV.2.2. Les moteurs:

Nous obtenons le mouvement mécanique du robot grâce à deux moteurs à courant continu connectés aux deux roues motrices. Ce type de moteur (Standard 130) est utilisé en robotique et on le retrouve également dans les jouets électriques pour enfants, mais dans ce projet, nous ajouterons une boîte de vitesses (Gearbox) pour augmenter le couple (Figure IV.2). Les caractéristiques du moteur comme suite [15]:

Tableau IV.1. Les caractéristiques du moteur

Modèle	Moteur à courant continu de type 130 standard
Tension de fonctionnement	4,5 V à 9 V
Tension nominale	6 V
Courant à vide	70mA (max)
Vitesse à vide	9000 tr / min
Courant chargé	250mA (environ)
Charge nominale	10g*cm
Taille du moteur	27,5 mm × 20 mm × 15 mm
Poids	20 g(environ)



Figure IV.2. Moteur à courant continu avec Gearbox

IV.2.3. Drive of motor (L298N):

Comme nous l'avons dit, le robot que nous créons, dont la fonction principale est d'éviter les obstacles, et pour cela nous oblige à utiliser la technologie appropriée pour contrôler les moteurs de ce robot en termes de vitesse et de sens de rotation (direction) du moteur, et pour ça on a utilisé le L298N.

Avant d'expliquer le principe de fonctionnement du L298N, nous devons comprendre deux techniques de base, à savoir le PWM et le pont en H (H-bridge).

IV.2.3.1. Contrôle la direction du moteur par H-bridge:

Pour contrôler le sens de rotation du moteur à courant continu, il suffit d'inverser le sens du courant électrique à travers le moteur, la façon la plus courante de le faire est avec un pont en H. Le circuit en pont en H a quatre éléments de commutation, où le moteur au centre forme une configuration similaire à H. En activant deux interrupteurs spécifiques en même temps, nous pouvons changer le sens du courant électrique, changeant ainsi le sens de rotation du moteur. La plupart des applications de ponts H utilisent quatre transistors MOSFET pour agir comme commutateurs de tension [16].

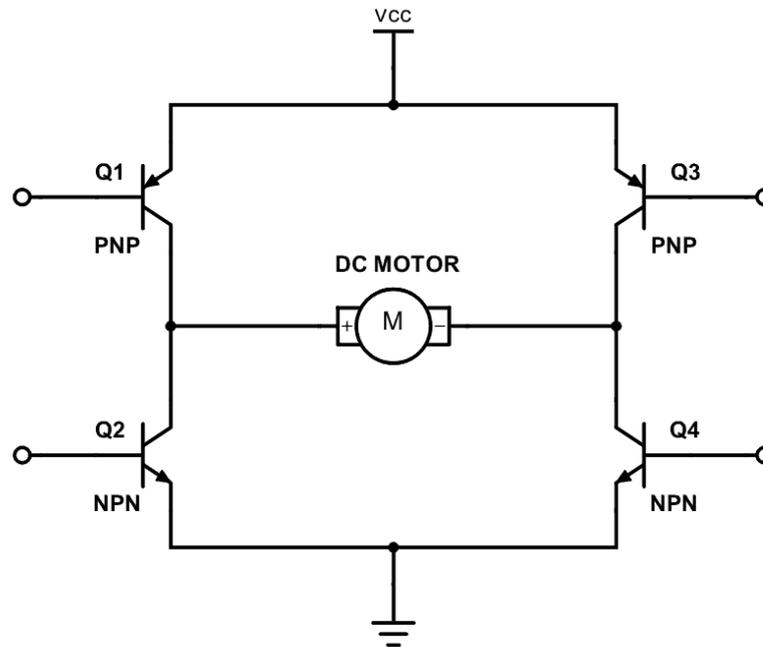


Figure IV.3. Contrôle du moteur à courant continue par H-bridge [28]

IV.2.3.2. Contrôle la vitesse du moteur par PWM:

Cette technologie nous permet de contrôler la vitesse des moteurs de robots en contrôlant la tension électrique du moteur.

PWM, ou (Pulse Width Modulation) est une technologie qui permet d'ajuster la valeur de tension moyenne qui va au moteur (ou à tout appareil électronique) en allumant et en coupant l'alimentation à une période très rapide T . La tension moyenne dépend du Duty cycle ou de la durée de fonctionnement T_{On} du signal par rapport à la durée de fermeture T_{Off} du signal au cours d'une même période T (Figure IV.4).

Nous pouvons expliquer cela avec la relation suivante:

$$V = \frac{T_{On}}{T_{On} + T_{Off}} V_{max} = \frac{T_{On}}{T} V_{max}$$

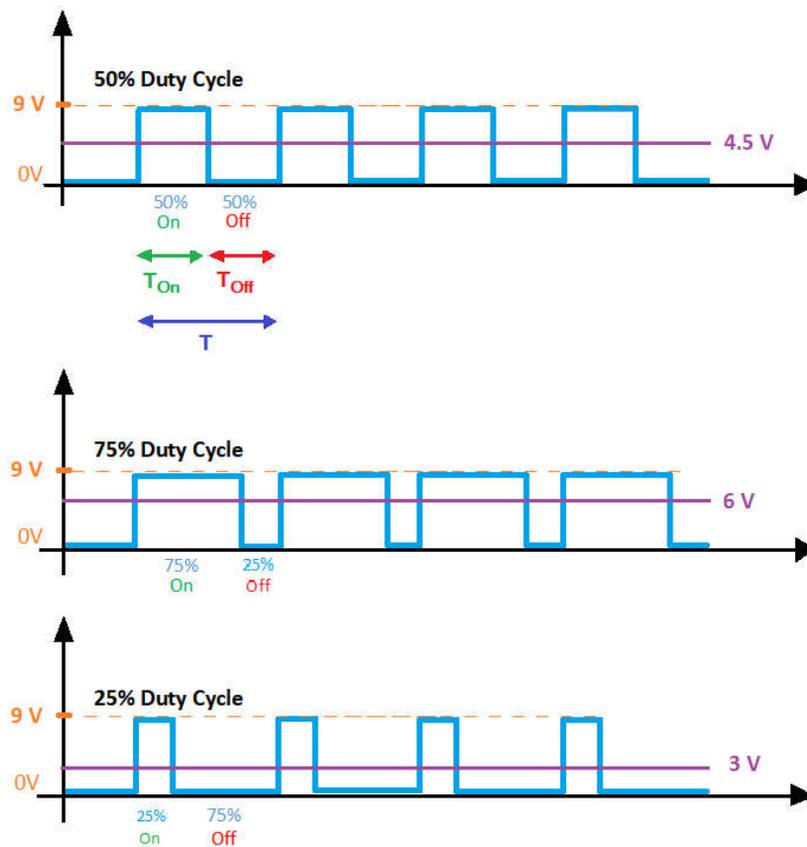


Figure IV.4. PWM (Pulse Width Modulation)

Nous pouvons simplement connecter la sortie Arduino PWM à la base du transistor ou à la porte MOSFET et contrôler la vitesse du moteur en contrôlant la sortie PWM.

Donc, si nous combinons ces deux méthodes, le PWM et l'H-Bridge, nous pouvons avoir un contrôle complet sur le moteur à courant continu. Il existe de nombreux pilotes de moteur à courant continu qui ont ces caractéristiques et le L298N en fait partie.

Le L298N est un pilote de moteur qui permet de contrôler la vitesse et la direction de deux moteurs à courant continu en même temps. Le module peut piloter des moteurs à courant continu ayant des tensions comprises entre 5 et 35 V, avec un courant jusqu'à 2A [16].

Nous allons maintenant connaître les astuces du L298N et expliquer comment il fonctionne. Le L298N contient deux bornes à vis pour les moteurs A et B, et d'autres bornes comme le montre la figure suivante (Figure IV.5), et nous expliquerons la fonction de ces bornes [17].

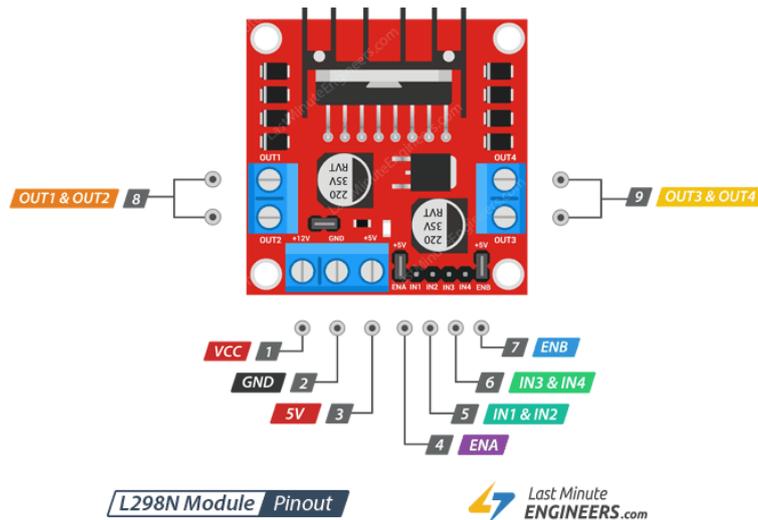


Figure IV.5. Les bornes du L298N [29]

VCC La borne VCC alimente le moteur. Il peut être compris entre 5 et 35V, il vaut mieux être supérieur à 7V pour une vitesse du moteur plus rapide.

5V La borne alimente le circuit logique de commutation à l'intérieur du circuit intégré L298N. Si la borne 5V-EN est en place, cette borne (5V) agit comme une sortie et peut être utilisée pour alimenter notre Arduino. Si la borne 5V-EN est retirée, nous devons le connecter à la broche 5V sur Arduino.

GND Est une borne de masse commune.

ENA Les bornes sont utilisées pour contrôler la vitesse du moteur A.

ENB Les bornes sont utilisées pour contrôler la vitesse du moteur B.

IN1 & IN2 Les broches sont utilisées pour contrôler la direction de rotation du moteur A. Lorsque l'une d'elles est (HIGH) et l'autre est (LOW), le moteur A tourné. Si les deux entrées sont (HIGH) ou (LOW), le moteur A s'arrêtera.

IN3 & IN4 Les broches sont utilisées pour contrôler la direction de rotation du moteur B. Lorsque l'une d'elles est (HIGH) et l'autre est (LOW), le moteur A tourné. Si les deux entrées sont (HIGH) ou (LOW), le moteur B s'arrêtera.

OUT1 & OUT2 Les bornes sont connectées au moteur A.

OUT3 & OUT4 Les bornes sont connectées au moteur B.

Donc, nous pouvons contrôler la vitesse des moteurs par les bornes ENA et ENB ,et le sens de rotation des moteurs par les bornes de commande IN1, IN2, IN3 et IN4.

IV.2.4. Les capteurs à ultrasons:

La caractéristique du robot que nous voulons créer est sa capacité à éviter les obstacles et à ne pas entrer en collision avec eux, ce qui oblige le robot à percevoir le monde qui l'entoure grâce aux capteurs externes dont nous avons parlé dans le premier chapitre.

Le capteur que nous utiliserons dans notre travail est le capteur à ultrasons dont nous avons parlé dans le premier chapitre et nous l'expliquerons plus dans ce chapitre, où nous utiliserons le capteur HC-SR04 qui est plus courant dans le domaine de l'éducation et de la recherche.

IV.2.4.1. Le principe de fonctionnement général:

Avant d'expliquer le principe de fonctionnement, il faut savoir que l'ultrason est une onde sonore haute fréquence (plus 20 KHz) dont la fréquence dépasse la plage audible de l'audition humaine (Figure IV.6).

Généralement, les capteurs à ultrasons fonctionnent en envoyant l'impulsion sonore hors de portée de l'audition humaine. Cette impulsion est transmise sous forme conique à la vitesse du son (340 m/s). Cette impulsion est réfléchié lorsqu'elle entre en collision avec un objet et retourne au récepteur dans le capteur. Le capteur interprète cela comme un écho et calcule l'intervalle de temps entre la transmission du signal et la réception de l'écho. Cet intervalle est ensuite calculé par le contrôleur pour déterminer la distance de l'objet. En relation simple [18]:

$$La\ distance = \frac{Le\ temps\ passé \times La\ vitesse\ du\ son}{2}$$

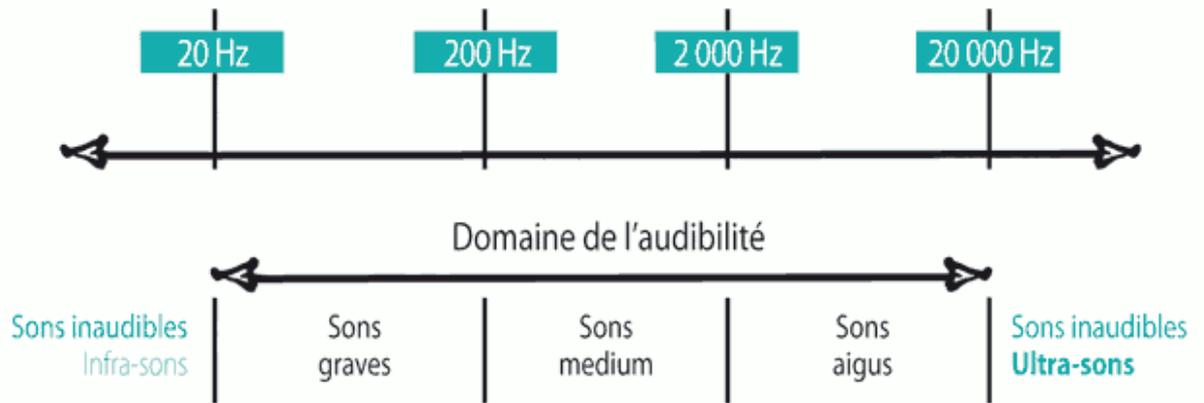


Figure IV.6. Fréquence des ultrasons et humaine [30]

IV.2.4.2. Le capteur HC-SR04:

Le capteur de distance à ultrasons HC-SR04 se compose essentiellement de deux transducteurs à ultrasons (émetteur et récepteur). L'émetteur agit comme un émetteur qui convertit le signal électrique en impulsions ultrasonores à 40 kHz. Le récepteur écoute les impulsions transmises. S'il est reçu, il produit une impulsion de sortie qui peut être utilisée dans sa largeur pour déterminer la distance parcourue par l'impulsion [17].

Le capteur contient à quatre bornes Vcc, Gnd, Echo, et Trig, nous expliquerons la fonction de ces bornes.

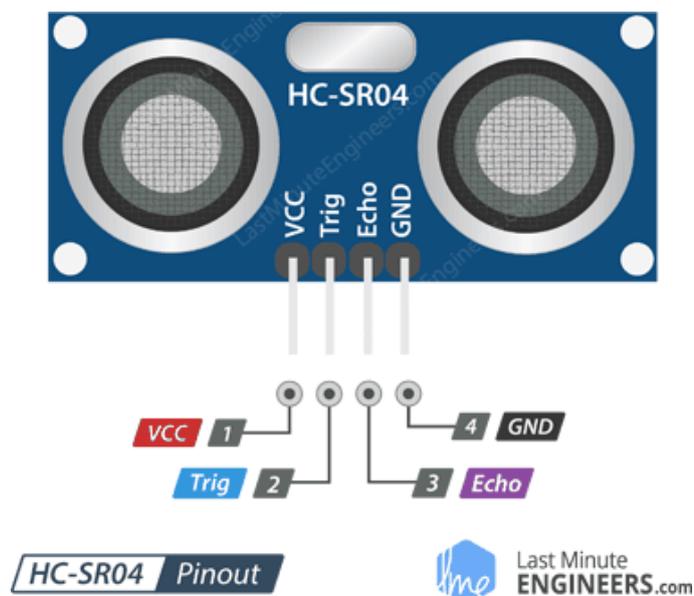


Figure IV.7. Le capteur HC-SR04 [31]

VCC Est la source d'alimentation du capteur que nous branchons sur la broche 5V sur Arduino.

GND Est une borne de masse commune.

Trig La borne Trig (Pin Trigger) est utilisée pour les impulsions sonores ultrasonores.

Echo La borne d'écho produit une impulsion lors de la réception du signal réfléchi. La longueur d'impulsion est proportionnelle au temps mis pour détecter le signal transmis.

Une impulsion d'au moins 10 μ S (10 microsecondes) est appliquée à la borne **Trig**. En réponse à cela, le capteur envoie une explosion sonore à huit impulsions (8 pulse) à 40 kHz. Ce motif à 8 impulsions rend la "signature ultrasonore" de l'appareil unique, permettant au récepteur (borne **Echo**) de distinguer le motif transmis du bruit ultrasonore environnant (Figure IV.8).

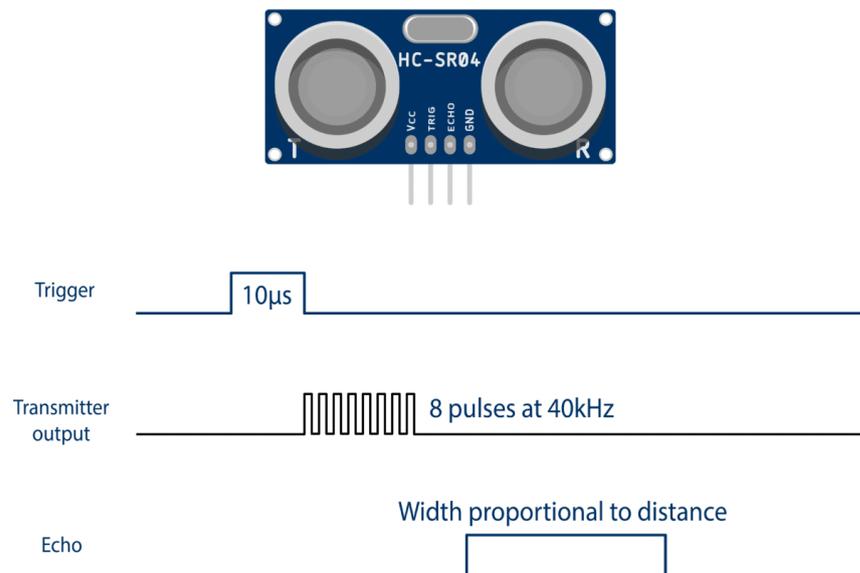


Figure IV.8. Le principe de fonctionnement du capteur HC-SR04 [32]

Les huit impulsions (8 pulse) ultrasonores voyagent dans l'air loin de l'émetteur. Pendant ce temps, la broche d'écho va activer (HIGH) pour commencer à former le début du signal de réception.

Dans le cas, si ces impulsions ne sont pas réfléchies dans un objet, le signal d'écho se terminera après 38 ms (38 ms) et deviendra faible. Ainsi, une impulsion de 38 ms indique qu'il n'y a pas d'obstacle dans la plage du capteur.

Si ces impulsions sont à nouveau réfléchies, la borne **Echo** devient inactive (LOW) et ce dès que le signal est reçu. Il en résulte une impulsion d'une largeur allant de 150 μ s à 25 ms, en fonction du temps nécessaire pour recevoir le signal.

Supposons que nous ayons un objet devant le capteur à une distance inconnue et que nous ayons reçu une impulsion de 500 μ s sur la borne d'écho. Calculons maintenant la distance entre l'objet et le capteur. Nous utiliserons l'équation ci-dessous.

$$Distance = vitesse \times temps$$

Nous avons le temps 500 μ s et une vitesse du son de 340 m/s c'est-à-dire 0,034 cm/ μ s. Mais l'impulsion indique le temps nécessaire pour transmettre le signal et sa réflexion, il faut donc diviser le temps par deux.

$$Distance = (0,034 \text{ cm}/\mu\text{s} \times 500\mu\text{s}) / 2 = 8,5 \text{ cm}$$

Maintenant, nous savons que l'objet est à 8,5 cm du capteur.

Le capteur est petit et facile à utiliser dans n'importe quel projet de robotique et peut fonctionner sur une plage comprise entre 2 cm et 400 cm avec une précision de 3 mm. Puisqu'il fonctionne sur 5V, il peut être connecté directement à Arduino ou à tout microcontrôleur logique 5V. Le tableau suivant montre les spécifications du capteur.

Tableau IV.2. Les spécifications du capteur à ultrason HC-SR04

Modèle	HC-SR04
Tension de fonctionnement	DC 5V
Courant de fonctionnement	15mA
Fréquence de fonctionnement	40 KHz
Plage maximale	400 cm
Plage minimale	2 cm
Précision	3 mm
Angle de mesure	15 degrés
Trigger Input Signal	10 μ S TTL pulse
Dimension	45mm \times 20mm \times 15mm

IV.2.5. La carte Arduino Uno:

Arduino Uno est une carte microcontrôleur basée sur l'ATmega328P. Il dispose de 14 bornes d'entrée / sortie numériques (dont 6 peuvent être utilisées comme sorties PWM), 6 entrées analogiques, un résonateur en céramique 16 MHz (CSTCE16M0V53-R0), une connexion USB, une prise d'alimentation, un en-tête ICSP et un bouton de réinitialisation. Il contient tout le nécessaire pour supporter le microcontrôleur; connectez-le simplement à un ordinateur avec un câble USB ou alimentez-le avec un adaptateur AC-DC ou une batterie pour commencer [19].

"Uno" signifie un en italien et a été choisi pour marquer la sortie du logiciel Arduino (IDE) 1.0, qui étaient les versions de référence d'Arduino, désormais évoluées vers des versions plus récentes [19].

ATmega328 contient 32 Ko et possède 2 Ko de SRAM et 1 Ko d'EEPROM.

IV.2.5.1. Alimentation de la carte Arduino

La carte Arduino Uno peut être alimentée via une connexion USB, en utilisant une source d'alimentation externe, ou via un adaptateur AC- DC ou une batterie. L'adaptateur peut être connecté en branchant une fiche de 2,1 mm dans la prise d'alimentation de la carte. Les câbles de batterie peuvent être insérés dans les têtes GND et Vin du générateur. La carte peut fonctionner sur une source externe de 6 à 20 volts. Cependant, si elle est fournie avec moins de 7V, une borne 5 V peut fournir moins de 5V et la carte peut devenir instable. Si vous utilisez plus de 12 volts, le régulateur de tension peut chauffer et endommager la plaque. La plage recommandée est de 7 à 12 volts. Les bornes d'alimentation sont les suivantes:

Vin Tension d'entrée sur la carte Arduino lors de l'utilisation d'une source d'alimentation externe. Nous pouvons fournir la tension de la carte à travers cette broche et c'est lors de l'utilisation d'un générateur.

5V Cette borne (Output) émet une tension de 5V à partir du régulateur de tension sur la plaque.

3V3 Cette borne (Output) émet une tension de 3.3V à partir du régulateur de tension sur la plaque.

GND Est une borne de masse commune.

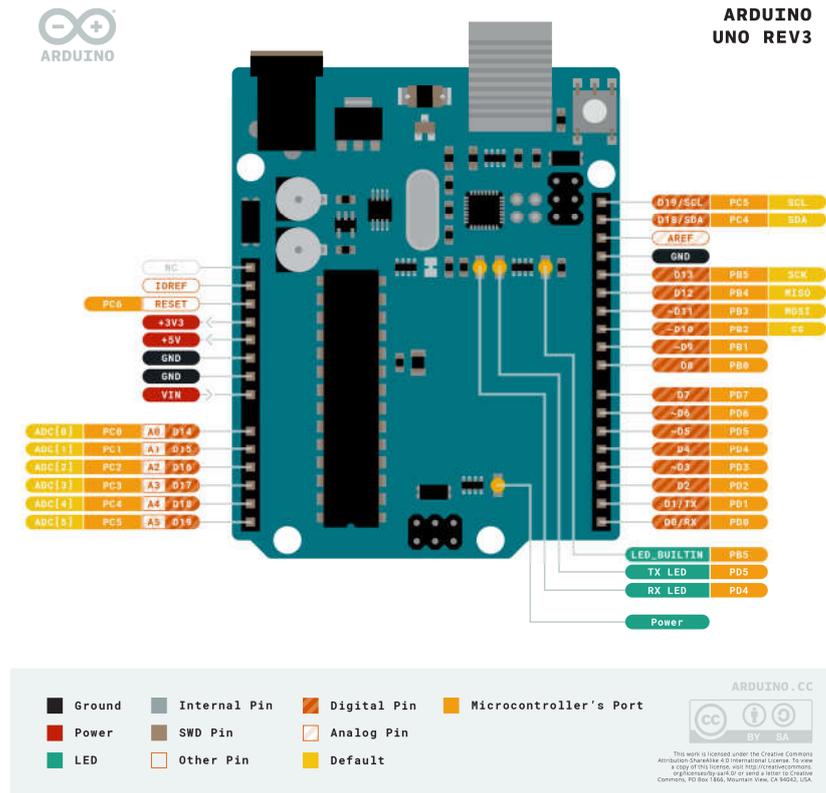


Figure IV.9. La carte Arduino Uno Rev3 [33]

IV.2.5.2. Les entrées et sorties:

La carte Arduino contient 14 entrées et sorties numériques (0 à 13). Nous découvrirons ses fonctions:

Serial: 0 (RX) et 1 (TX), RX est utilisé pour recevoir des données et TX pour envoyer des données. Ils peuvent être utilisés comme entrées et sorties numériques.

Interruptions externes: 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur faible, un front montant ou descendant, ou un changement de valeur.

Les bornes 3, 5, 6, 9, 10 et 11 peuvent être utilisées pour envoyer et recevoir des signaux PWM.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches prennent en charge la communication SPI à l'aide de la bibliothèque SPI.

La carte Arduino Uno possède 6 entrées analogiques, étiquetées de A0 à A5, qui fournissent chacune 10 bits de résolution (soit 1024 valeurs différentes).

Tableau IV.3. Les spécifications de la carte Arduino Uno Rev3

Microcontrôleur	ATmega328P
Tension de fonctionnement	5V
Tension d'entrée (recommandée)	7-12 V
Tension d'entrée (limite)	6-20 V
Broches d'E / S numériques	14 (dont 6 fournissent une sortie PWM)
Broches d'entrée analogique	6 (A0 - A5)
Courant DC pour les broche d'E / S	20 mA
Courant DC pour la broche 3,3 V	50 mA
Mémoire flash	32 Ko (ATmega328P)
SRAM	2 Ko (ATmega328P)
EEPROM	1 Ko (ATmega328P)
Vitesse d'horloge	16 MHz
Dimension	68,6 mm×53,4 mm
Poids	25 g

IV.3. Coté Software:

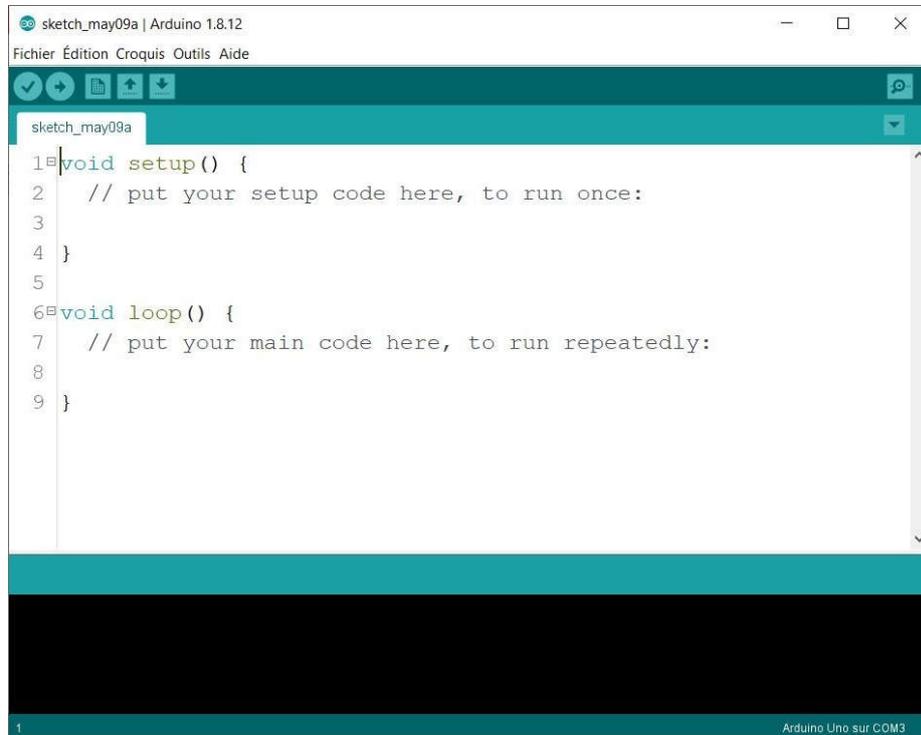
En plus d'être des robots sont des machines, ce des machines programmables. Wikipédia définit le robot comme une machine, en particulier une machine programmable par un ordinateur capable d'exécuter automatiquement une série complexe d'actions. Il est intéressant dans la définition de Wikipédia qu'ils disent spécifiquement une machine programmable par ordinateur [14].

IV.3.1. Arduino IDE:

Arduino IDE est un logiciel open source qui est principalement utilisé pour écrire et compiler le code dans le module Arduino. Arduino IDE est un environnement de développement intégré et application multiplateforme qui est écrite dans des fonctions de C et C++. Il est utilisé pour écrire et télécharger des programmes sur des cartes compatibles Arduino [20].

Le code principal créé sur l'interface IDE générera finalement un fichier hexadécimal qui sera ensuite transféré et téléchargé dans le contrôleur de la carte.

L'environnement IDE contient principalement deux parties de base: l'éditeur et le compilateur où l'ancien est utilisé pour écrire le code requis et plus tard est utilisé pour compiler et télécharger le code dans le module Arduino donné.



```
sketch_may09a | Arduino 1.8.12
Fichier Édition Croquis Outils Aide
sketch_may09a
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
Arduino Uno sur COM3
```

Figure IV.10. Interface logicielle Arduino IDE

Il s'agit d'un programme officiel d'Arduino, simple à utiliser et ne nécessite pas de connaissances techniques préalables de l'utilisateur. Cela le rend approprié même pour les débutants.

IV.4. La réalisation et la programmation :

IV.4.1. La réalisation :

Après avoir fourni les outils et éléments électroniques pour créer un robot mobile pour éviter les obstacles, nous avons préparé la plate-forme mobile pour qu'elle soit adaptée à tous ses équipements, puis nous avons relié les composants électroniques entre eux (figure IV.11).

Les trois capteurs ultrasoniques de distance que nous avons connectés aux terminaux analogiques Arduino, comme suite:

La broche **Trig1** sur le premier capteur avec la broche **7** sur la carte Arduino.

La broche **Echo1** sur le premier capteur avec la broche **6** sur la carte Arduino.

La broche **Trig2** sur le deuxième capteur avec la broche **5** sur la carte Arduino.

La broche **Echo2** sur le deuxième capteur avec la broche **4** sur la carte Arduino.

La broche **Trig3** sur le troisième capteur avec la broche **3** sur la carte Arduino.

La broche **Echo3** sur le troisième capteur avec la broche **2** sur la carte Arduino.

Le L298N (Drive of Motor) nous avons connectés aux terminaux numérique/analogiques Arduino, comme suite:

La broche **IN1** sur le L298N avec la broche **10** sur la carte Arduino.

La broche **IN2** sur le L298N avec la broche **9** sur la carte Arduino.

La broche **IN3** sur le L298N avec la broche **12** sur la carte Arduino.

La broche **IN4** sur le L298N avec la broche **11** sur la carte Arduino.

La broche **OUT1** et **OUT2** sur le L298N avec les bornes du moteur droit.

La broche **OUT3** et **OUT4** sur le L298N avec les bornes du moteur gauche.

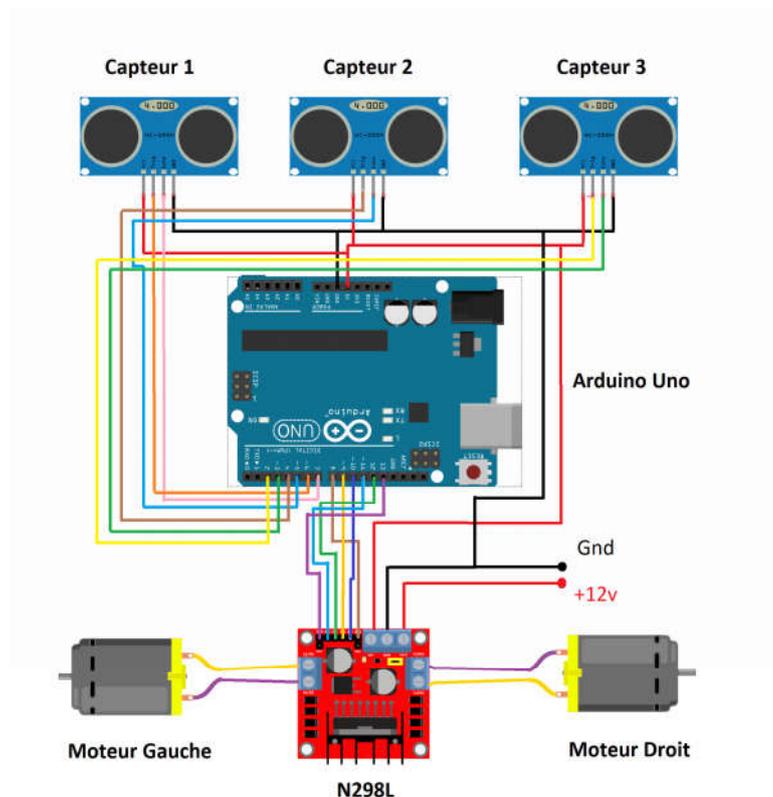


Figure IV.11 : L'installation d'un circuit électrique entre les éléments électroniques d'un robot mobile pour éviter les obstacles

IV.4.2. L'organigramme d'évitement d'obstacle

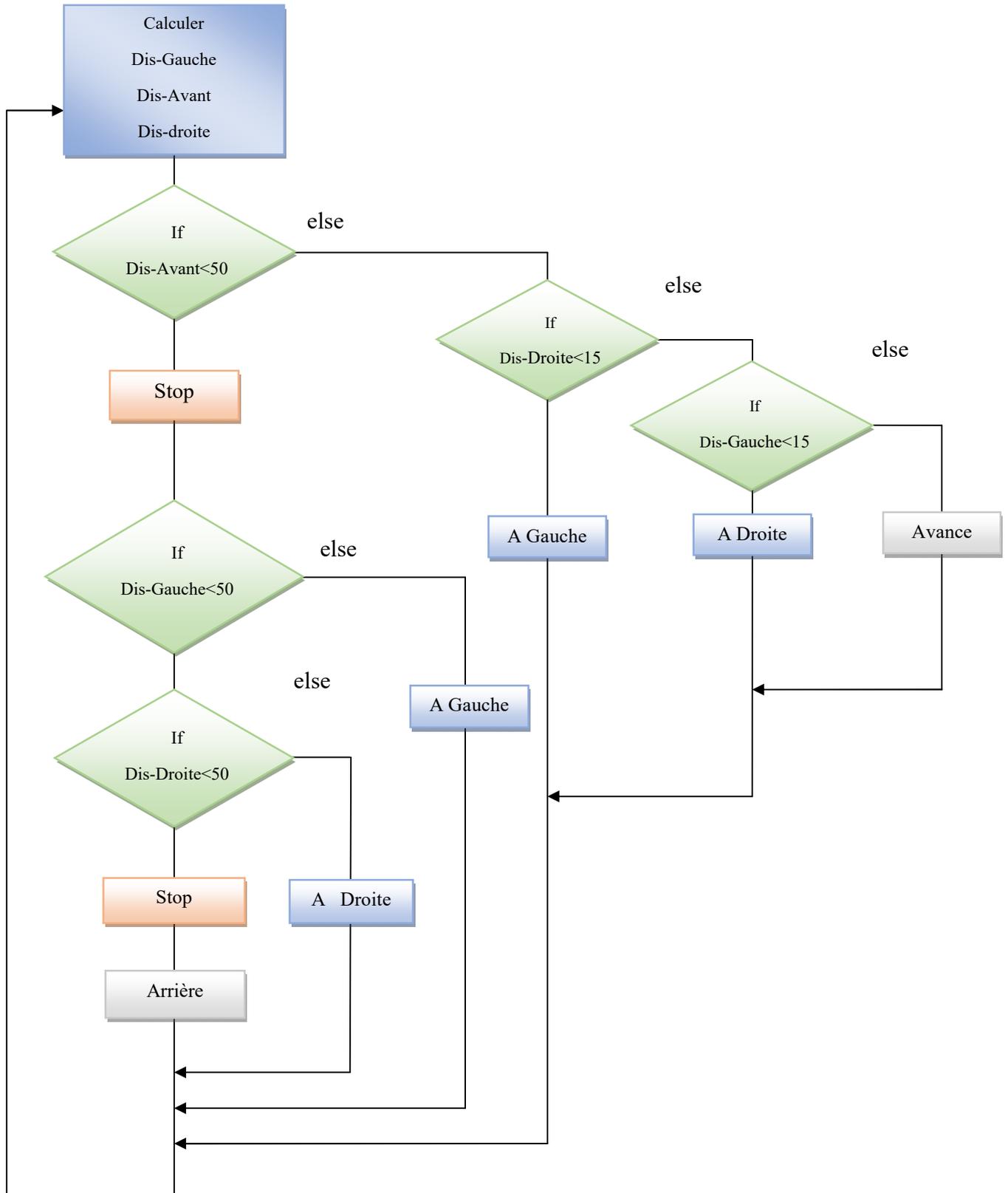


Figure IV.12 : L'organigramme d'évitement d'obstacles appliqué dans le projet.

IV.5. Implémentation et résultats:

Nous avons installé le circuit illustré à la figure IV.11 et inséré le code de l'algorithme illustré à la figure IV.12.

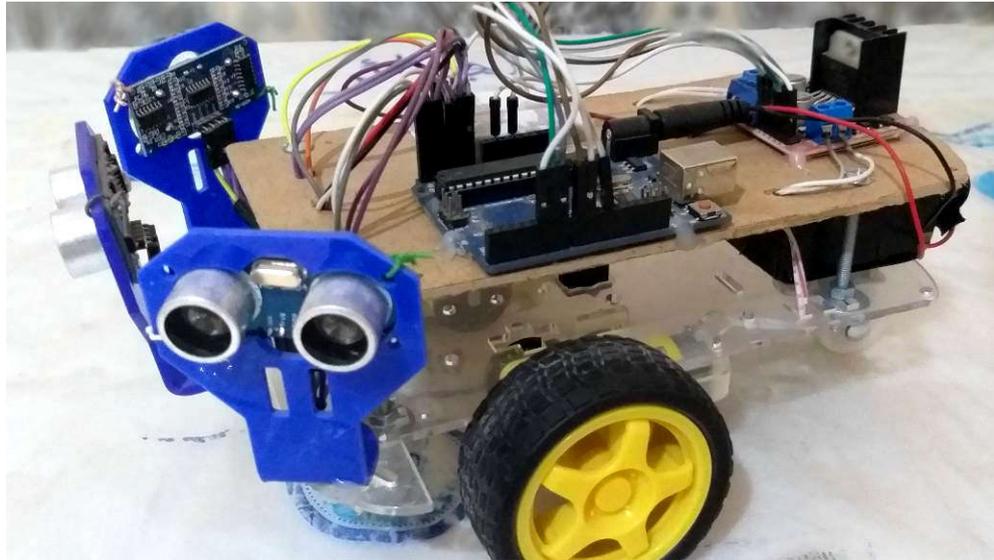


Figure IV.13 : robot mobile à entraînement différentiel d'évitement d'obstacles équipé de trois capteurs à ultrason

Pour valider notre projet en termes d'installation et de programmation, nous avons testé le robot que nous avons créé, afin de pouvoir dire que le robot est capable d'éviter les obstacles.

L'un des inconvénients que nous avons remarqués est que le robot entre en collision avec des obstacles petits, pointus ou bas au niveau des capteurs.



Figure IV.14 : Test de robot mobile pour éviter les collisions avec des obstacles.

IV.6. Conclusion :

Un prototype du robot à entraînement différentiel a été conçu et construit, basé sur une carte Arduino pour le contrôle et des capteurs à ultrasons pour la perception.

Le prototype a été testé pour montrer sa capacité de navigation autonome dans son environnement en évitant les obstacles.

Conclusion Générale

Conclusion générale

Dans ce travail, le problème de la navigation autonome d'un robot mobile a été abordé en utilisant des techniques floues. La tâche de base que le robot mobile doit faire est de suivre une trajectoire souhaitée, à partir d'une configuration donnée, tout cela sans intervention humaine, et avec le minimum d'erreur.

Dans un premier temps, nous avons présenté quelques concepts sur les robots mobiles et leurs types, et donné un aperçu de la fonctionnalité essentielle qui doit être disponible dans tous les robots mobiles, à savoir la navigation autonome. Les éléments de base pour parvenir à une navigation autonome sont la navigation, la localisation, la cartographie et la perception.

Après un aperçu global de la navigation du robot mobile, un modèle de robot mobile à entraînement différentiel a été développé, où la cinématique et la dynamique du robot ont été dérivées en détail. Ce type de robot est largement utilisé, notamment dans le domaine de la recherche en raison de sa simplicité et de sa facilité de travail.

Le problème du suivi de trajectoire a été considéré. Le robot doit suivre une trajectoire souhaitée. Le premier choix est d'utiliser un régulateur PID car c'est le plus simple. Mais malheureusement, le PID n'a pas réussi à suivre la trajectoire souhaitée. et cela est dû au degré élevé de non-linéarité du système, où la dynamique du robot a deux équations de mouvement couplées non linéaires. Alors, la meilleure solution semble être le contrôle flou. La logique floue est un raisonnement approximatif qui peut faire face à l'incertitude de l'information. Il peut produire des contrôleurs robustes à l'incertitude et à l'imprécision basés sur un ensemble de règles IF-THEN dans lesquelles les connaissances d'experts peuvent être utilisées. Un contrôleur flou a été conçu à cet effet. Les résultats de simulation montrent l'efficacité de l'approche proposée basée sur l'utilisation d'un nombre minimum de règles.

Enfin, un prototype du robot a été conçu et construit. Le robot est piloté avec une carte Arduino, et équipé de trois capteurs à ultrasons lui permettant d'éviter les obstacles sur son chemin.

Travaux futurs et perspectives

Ce travail fournit une base pour les orientations futures. Le robot peut être équipé de caméras pour une navigation sûre dans des environnements dynamiques.

Références

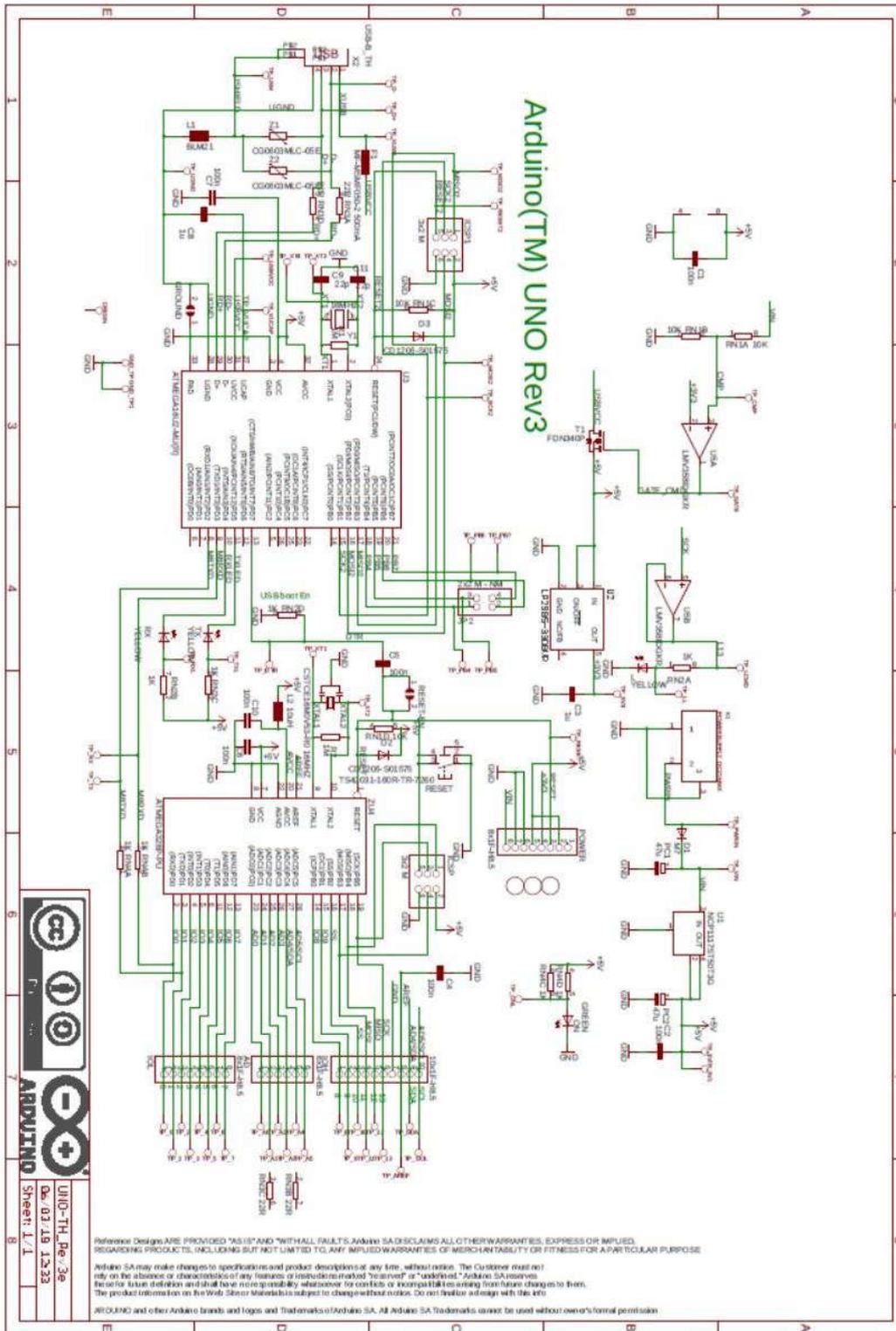
Références

- [01] CHERROUN Lakhmissi. Navigation Autonome d'un Robot Mobile par des Techniques Neuro-Floues. Automatique. Université Mohamed Khider – Biskra, 2014.
- [02] Olivier Lefebvre. Navigation autonome sans collision pour robots mobiles non-holonomes. Automatique/ Robotique. Institut National Polytechnique de Toulouse - INPT, 2006. Français. tel-00134581
- [03] Stéphane Lens. Locomotion d'un robot mobile. 3ELIN. Université de Liège, 2008.
- [04] C.benson. What Types of Mobile Robots are there? <https://www.Robotshop.com/.17/09/2018>
- [05] Bernard BAYLE. Robotique Mobile. Université de Strasbourg, bernard.bayle@unistra.fr
- [06] Abdelmadjid BENMACHICHE. Approche de Navigation Coopérative et Autonome des Robots Mobiles (Application sur un chantier de construction). Reconnaissance des Formes et Intelligence Artificielle. Université Badji Mokhtar -Annaba, 2016.
- [07] A. Oualid Djekoune. Localisation Et Guidage Du Robot Mobile Atrv2 Dans Un Environnement Naturel. Thèse de Doctorat, Université des Sciences et de la Technologie Houari Boumediene, 2010.
- [08] Omar Ait Aider. Localisation référencée modèle d'un robot mobile d'intérieur. Traitement du signal et de l'image. Université d'Evry-Val d'Essonne, 2002.
- [9] N. HACENE, Note de cours : Robotique, Université de Ghardaia, 2018/2019.
- [10] Lionel Geneve. Système de déploiement d'un robot mobile autonome basé sur des balises. Automatique/ Robotique. Université de Strasbourg, 2017. Français. NNT:2017STRAD024. Tel-01713665
- [11] Oussama El Hamzaoui. Localisation et cartographie simultanées pour un robot mobile équipé d'un laser à balayage: Core SLAM. Autre [cs.OH]. Ecole Nationale Supérieure des Mines de Paris, 2012.Français. NNT : 2012ENMP0103. pastel-00935600
- [12] Boru Diriba Hirpo & Prof. Wang Zhongmin. Design and Control for Differential Drive Mobile Robot. School of Mechanical Engineering, Tianjin University of Technology and Education, October 2017. Tianjin 300222, China
- [13] Rached Dhaouadi, and Ahmad Abu Hatab. Dynamic Modeling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework. College of Engineering, American University of Sharjah, Sharjah, UAE. Adv Robot Autonomy 2013
- [14] Paul Heney. Robots are software and software is "eating the world". <https://www.therobotreport.com/>, December 29, 2017

- [15] <https://components101.com/>
- [16] Dejan. <https://howtomechatronics.com/>
- [17] <https://lastminuteengineers.com/>
- [18] Nathan Kelly. Application Note: A Guide to Ultrasonic Sensor Set Up and Testing Instructions, Limitations, and Sample Applications. MICHIGAN State University. kellyna4@gmail.com. ECE 480 – Design Team 5
- [19] Le site officiel d'Arduino: <https://store.arduino.cc/arduino-uno-rev3>
- [20] Le site officiel d'Arduino https://arduino.cc/Arduino_IDE
- [21] Xi Li & Byung-Jae Choi. Design of Obstacle Avoidance System for Mobile Robot using Fuzzy Logic Systems. School of Electronic Engineering, Daegu University Jillyang Gyeongsan-city Gyeonbuk, Korea. li_xi_lixi@hotmail.com, bjchoi@daegu.ac.kr
- [22] Dr. Radhwane SADOUNI. Commande Intelligente. Automatique et systèmes. Université de Ghardaïa – Ghardaïa.
- [23] MECHALIKH Youcef & MILOUDDI Ali. Développement d'algorithmes d'évitement d'obstacles statiques et dynamiques. Génie Electrique/Automatique. Université Kasdi Merbah–Ouargla, 2011/2012.
- [24] Nacer Hacene & Boubekeur Mendil. Fuzzy Behavior-based Control of Three Wheeled Omnidirectional Mobile Robot. International Journal of Automation and Computing, April 2019.
- [25] Thoa T.Mac, Cosmin Copot, Robin DE KEYSER, Trung D.Tran, Thich Vu. MIMO Fuzzy Control for Autonomous Mobile Robot. GhentUniversity Department of Electrical energy, Systems and Automation Sint-Pietersnieuwstraat 41, B-9000 GENT, Belgium, Hanoi University of Science and Technology (HUST) School of Mechanical Engineering Dai Co Viet street 1, Hanoi, Vietnam.

- [26] Figure I.1: <https://www.cite-telecoms.com/>
- [27] Figure I.4: <https://www.robotpark.com/>
- [28] Figure IV.3 : <https://www.build-electronic-circuits.com/h-bridge/>
- [29] Figure IV.5 : <https://lastminuteengineers.com/>
- [30] Figure IV.6 : <https://sites.google.com/>
- [31] Figure IV.7 : <https://lastminuteengineers.com/>
- [32] Figure IV.8. : <https://www.makerguides.com/>
- [33] Figure IV.9 : Le site officiel d'Arduino: <https://store.arduino.cc/arduino-uno-rev3>

Annexe A - Diagramme schématique de l'Arduino Uno



Annexe B - Diagramme schématique du Robot

