الجمهورية الجزائرية الديمقراطية الشعبية
People's Democratic Republic of Algeria
وزارة التعليم الـعـالي و البحـث العـلمـي
Ministry of Higher Education and Scientific Research
جـامعة سعد دحلب البليدة
SAAD DAHLAB University of BLIDA
كلية التكنولوجيا
Faculty of Technology
قسم الأوتوماتيك و ألإلكتروتقني
Automatic and electrical engineering department

Master's thesis

Sector: Electronics

Specialty: Telecommunications systems

Presented by

SAIDANI Aymen

&

BENYAHIA Said

Path following mobile robot

proposed by : Pr. KAZED Boulam

Academic year 2021-2022

Acknowledgements

We thank God the Almighty for having given us the health and the will to begin and finish this memorial.

First of all, this work would not be as rich and could not have been possible without the help of the promoter ***Mr. Kazed Boualem****, we thank him for the quality of his exceptional supervision, for his patience, his rigor and his availability during our preparation of this brief.*

Our thanks also go to all our teachers for their generosity and the great patience they have shown despite their academic and professional responsibilities

We also thank the jury members for their honor in judging our work.

We thank our friends for all their sincere friendship over the last five years of study.

Finally, We thank all those who have contributed directly or indirectly to the development of this work.

Dedication

To our dear mothers:

You have carried for us the care and effort for our education. No dedication can express all the respect and love we have for you.

You have always trusted us. Find in this work the consolation and witness of patience.

To our dear fathers:

Despite the great responsibilities you assume in your work or as fathers of families, you have always been close to us, to listen to us, to support us, to follow us and to encourage us. May this work diminish your suffering and bring you happiness.

To our dear brothers and sisters:

Most of that work is for you. You have always been very helpful to us. We thank you for all the good that each of you has done for us.

To our families.

To all our friends and classmates, may God preserve our friendship.

To all our teachers from Saad Dahleb Blida University

To all those who have trust us.

الملخص: في مشروع نهاية الدراسة هذا ، سنقوم ببناء روبوت ذاتي القيادة وتطوير نظام تحكم في الوقت الفعلي قادر علي التنقل في الاماكن المستوية او اتباع مسار محدد.

اولا سنقوم بتقديم تعريف عام للروبوتيك يليه اعطاء تعريف دقيق للمشوع و الخورزميات الشائعة للتنقل و تحديد المسار ، بعدها ننتقل الى الجانب الرياضي المشروع و المكانيكي ،فيه سنقوم بحساب المعادلات بالاطوال الحقيقية اللازمة للحركة و التحدث عن القطع اللازمة المكانيكية و الكهربائية و اعطاء مخطط الدارة الكهربائية ،في الخطوة التالية سنقوم بالتجارب و عرض النتائج.

في الاخر نخرج باستنتاج عام للمشروع و نقدم فيه ايضا بعض الاقتراحات للتطويرو التحسين مستقبلا .

Abstract: In this end-of-study project, we will build a self-driving robot and develop a real-time control system capable of navigating flat places or following a specific path.

First, we will provide a general definition of the robotic, followed by giving an accurate definition of the project and the common algorithms for movement and determination of the path. Then we will move to the mathematical and mechanical aspect, in which we will calculate the equations with the real lengths necessary for movement. and talk about the necessary mechanical and electrical parts and give the electrical circuit, in The next step we will do the experiments and show the results.

In the end, we come out with a general conclusion of the project and we offer some suggestions for future development and improvement.

Résumé: Dans ce projet de fin d'études, nous allons construire un robot autonome et développer un système de contrôle en temps réel capable de naviguer sur des terrains plats ou de suivre un chemin spécifique.

Tout d'abord, nous fournirons une définition générale de la robotique, suivie d'une définition précise du projet et des algorithmes communs pour le mouvement et la détermination de la trajectoire.

Puis nous passerons à l'aspect mathématique et mécanique, dans lequel nous calculerons l’équation avec les longueurs réelles nécessaires au mouvement et parler des pièces mécaniques et électriques nécessaires et donner le circuit électrique.

Dans la prochaine étape, nous ferons les expériences et montrerons les résultats, à la fin nous sortirons avec une conclusion générale du projet et nous offrons également quelques suggestions pour le développement et l'amélioration futurs.

Table of Contents

General Introduction: 1

1. Chapter 1 2

1.1. Introduction: 2

1.2. History of Robots: 2

1.3. Definition of robotics: 3

1.4. Classification of robots: 3

1.5. Types of mobile robots: 5

1.5.1. Autonomous mobile robots: 5

1.5.2. Industrial Robots: 5

1.5.3. Humanoid Robots: 7

1.5.4. Educational Robots: 8

1.5.5. Unmanned aerial vehicles (UAVs): 8

1.6. Applications: 8

1.7. Conclusion: 9

2. Chapter 2 10

2.1. Introduction: 10

2.2. Differential drive robot: 10

2.3. Autonomous Systems: 11

2.3.1. Sense: 12

2.3.2. Perceive: 12

2.3.1. Decide and Plan: 13

2.3.3. Act: 13

2.4. Path planning approaches: 14

2.4.1. Comparative study on the various trajectory-planning algorithms: 14

2.4.2. Potential Field algorithm: 15

2.5. Path following approaches: 17

2.5.1. Pure Pursuit algorithm: 17

2.6. Conclusion: 20

3. Chapter 3 21

3.1. Introduction: 21

3.2. Definition of modeling: 21

3.3. Differential drive robot modeling: 21

3.3.1. Differential drive robot type of motion: 21

3.3.2. Kinematics: 22

3.3.3. The ‘Unicycle’ model 24

3.3.4. Calculation of change in global coordinates 24

3.3.5. PID regulator: 26

3.3.6. Go to goal: 28

3.4. Tools: 29

3.4.1. Arduino board: 29

3.4.2. Sensors: 31

3.4.3. Actuators 32

3.4.4. Electrical scheme: 32

.3.5 Conclusion: 33

4. Chapter 4 35

4.1. Introduction: 36

4.2. PID Tuning: 36

4.2.1. P regulator: 36

4.2.2. PD regulator: 36

4.2.3. PID regulator: 37

4.3. Conclusion: 40

General Conclusion: 41

Bibliography: 42

List of figures:

Figure 1-1 : Classification of robots by application field 4

Figure 1-2 : Annual installation of industrial robots.. 4

Figure 1-3 : Industrial Robotic Arm.. 6

Figure 1-4 : Humanoid Robot.. 6

Figure 1-5 : Humanoid Robot. 7

Figure 1-6 : Educational Robot.................................**Erreur ! Signet non défini.**

Figure 1-7 : Educational Robot.................................**Erreur ! Signet non défini.**

Figure 2-1: Robot arms during spot welding and surface................................. 10

Figure 2-2: Robot arms during spot welding and surface................................. 10

Figure 2-3: autonomous robot avoide obstacles. 11

Figure 2-4: autonomous robot avoide obstacles. 11

Figure 2-5 : autonomous system scheme... 12

Figure 2-6 : autonomous system scheme... 12

Figure 2-7: robot-sensing environment. 12

Figure 2-8: robot-sensing environment. 12

Figure 2-9: street RAW image after interpreting... 13

Figure 2-10: street RAW image after interpreting... 13

Figure 2-11: Robot plan for the optimal path. 13

Figure 2-12: Robot plan for the optimal path. 13

Figure 2-13 : Potential field with seven obstacles. 15

Figure 2-14 : Potential field with seven obstacles. 15

Figure 2-15 : artificial potential field.. 16

Figure 2-16 : artificial potential field.. 16

Figure 2-17 : robot following trajectory. 18

Figure 2-18 : robot following trajectory. 18

Figure 2-19 : pure pursuit algorithm in action.. 19

Figure 2-20 : pure pursuit algorithm in action.. 19

Figure 2-21 : Small look ahead distance... 19

Figure 2-22 : Large look ahead distance.... 20

Figure 2-23 : Large look ahead distance.... 20

Figure 3-1 : Differential drive robot. 21
Figure 3-2 : Different types of movement. 22
Figure 3-3 : Side view of mobile robot. 22
Figure 3-4 : Top view of a mobile robot. 24
Figure 3-5 : Curve motion of a robot. 25
Figure 3-6 : Top view of a robot. 25
Figure 3-7 : Center point of the robot in global world. 26
Figure 3-8 : Diagram of P regulator. 27
Figure 3-9 : Diagram of I regulator. 27
Figure 3-10 : Diagram of D regulator. 28
Figure 3-11 : Trajectory represented by a series of points. 29
Figure 3-12 : Arduino Mega board. 30
Figure 3-13 : Encoder disk and output signals. 31
Figure 3-14 : DC motor parts. 32
Figure 3-15 : L298N motor driver. 32
Figure 3-16 : L298N motor driver. 32
Figure 3-17 : Mobile robot circuit. 33
Figure 4-1 : P regulator with Kp = 100. 36
Figure 4-2 : PD regulator with Kp = 100, Kd = 100. 37
Figure 4-3 : PID regulator with Kp = 15 , Ki = 0.9 , Kd = 100. 38
Figure 4-4 : PID regulator with Kp= 15 , Ki = 0.4 , Kd = 500. 38

Lists of abbreviations:

AC : alternating current.

AMR : Autonomous Mobile Robots.

DC : Direct Current.

GPS : global positioning system.

IC : Integrated Circuit.

ICSP :Interagency Committee on Standards Policy.

IDE : Integrated Development Environment.

IFR : International Federation of Robotics.

LED : Light Emitting Diode.

LIDAR : Light Detection and Ranging.

PID : proportional integral derivative.

PWM :Pulse-width modulation.

RADAR : Radio Detection and Ranging.

SCARA : Selective Compliance Assembly Robot Arm.

UAV : Unmanned aerial vehicles.

USB : universal serial bus.

General Introduction:

Commercial and industrial robots are now widely used for tasks that are less expensive, more accurate, and more reliable than humans are. They are also used in certain jobs that are too dirty, dangerous, or drab to be adapted to humans. Robots are widely used in manufacturing, assembly, packaging and transportation, ground and space exploration, surgery, weapons, laboratory research, Safety, and consumer and industrial goods mass production.

The goal of this project is to build a mobile autonomous robot and develop a real-time control system for a Differential mobile robot that is expected to be able to navigate a factory or warehouse or track a specific path autonomously.

Our mobile robot is based on a wheel encoder sensor and an Arduino Mega control board, as well as an L298N driver for powering the DC motors that drive the robot and a Li-Po battery as a power source.

This memoir is organized into four chapters divided as follows:

The first chapter will give a general overview of the robots with their history, classifications, types, and uses.

The second chapter will give an in-depth look at the concept of the Differential drive robot as well as give various path-planning algorithms and various path following algorithms. That chapter brings us closer to the concept of self-driving of mobile robot.

The third chapter will give the modeling of Differential drive robot explained possible cases of motion, State-space representation, The 'Unicycle' model representation, deduce the equations of motion and the change in global coordinates. In addition, give used tools: sensors, driver, power supply, controller and circuit of robot

The fourth chapter we talk about the tuning Testing of the robot and the result of our work.

Finally, we will end our memoir with a general conclusion of which we recapitulate what we have done, and we will done ,some suggestions for error-handling and developing to make a more efficient robot.

Chapter 1

Robotics

1.1. Introduction:

Robotics is an important area of research that calls on the cross-knowledge of several disciplines, its objective being to allow the robot to interact rationally with its environment without human intervention. Mobile robots have a special place in robotics. Their interest lies in their mobility, which opens up applications in many areas. Like manipulative robots, they are intended to assist humans in tasks arduous (transporting heavy loads, etc.), monotonous or in a hostile environment (nuclear, marine, space, firefighting, surveillance, etc.).

The autonomy of the mobile robot is a faculty that allows it to adapt or take a decision in order to carry out a task even in an unknown environment [1].

In this chapter, we will provide a comprehensive definition of the robot and a brief overview of its history, as we will classify them according to their working environment.

1.2. History of Robots:

Robotics has passed through several generations as follows:

1921: First fictional automatons called "robots" appear in the play R.U.R (Rossum's Universal Robots).

1930: Humanoid robot exhibited at the 1939 and 1940 World's Fairs.

1948 : Simple robots exhibiting biological behaviors, Elsie and Elmer robot invented by William Grey Walter [2].

1956 : irst commercial robot, from the “Unimation” company founded by George Devol and Joseph Engelberger, based on Devol's patents[3].

1960 : irst installed industrial robot.

1975: Programmable universal manipulation arm, a Unimation product.

1978 : irst object-level robot programming language, allowing robots to handle variations in object position, shape, and sensor noise.

Fully autonomous robots only appeared in the second half of the 20th century. The first digitally operated and programmable robot, the Unimate, was installed in 1961 to lift hot pieces of metal from a die casting machine and stack them. Commercial and industrial robots are widespread today and used to perform jobs more cheaply, more accurately and more reliably than humans.

1.3. Definition of robotics:

Although everyone seems to know what a robot is, it is hard to give a precise definition. The Oxford English Dictionary gives the following definition: “A machine capable of carrying out a complex series of actions automatically, especially one programmable by a computer.” This definition includes some interesting elements:

“Carrying out actions automatically.” This is a key element in robotics, but also in many other simpler machines called automata. The difference between a robot and a simple automaton like a dishwasher is in the definition of what a “complex series of actions” is. Is washing clothes composed of a complex series of actions or not? Is flying a plane on autopilot a complex action? Is cooking bread complex? For all these tasks, there are machines that are at the boundary between automata and robots.

“Programmable by a computer” is another key element of a robot, because some automata are programmed mechanically and are not very flexible. On the other hand computers are found everywhere, so it is hard to use this criterion to distinguish a robot from another machine.

A crucial element of robots that is not mentioned explicitly in the definition is the use of sensors. Most automata do not have sensors and cannot adapt their actions to their environment. Sensors are what enable a robot to carry out complex tasks [4].

1.4. Classification of robots:

Robots can be classified according to the environment in which they operate. The most common distinction is between fixed and mobile robots. These two types of robots have very different working environments and therefore require very different capabilities. Fixed robots are mostly industrial robotic manipulators that work in well defined environments adapted for robots. Industrial robots perform specific repetitive tasks such soldering or painting parts in car manufacturing plants. With the improvement of sensors and devices for human-robot interaction, robotic manipulators are increasingly used in less controlled environment such as high-precision surgery.

By contrast, mobile robots are expected to move around and perform tasks in large, ill-defined and uncertain environments that are not designed specifically for robots. They need to deal with situations that are not precisely known in advance and that change over time. Such environments can include unpredictable entities like humans and animals. Examples of mobile robots are robotic vacuum cleaners and self-driving cars.

There is no clear dividing line between the tasks carried out by fixed robots and mobile robots, humans may interact with industrial robots and mobile robots can be constrained to move on tracks, but it is convenient to consider the two classes as fundamentally different. In particular, fixed robots are attached to a stable mount on the ground, so they can compute their position based on their internal state, while mobile robots need to rely on their perception of the environment in order to compute their location.

There are three main environments for mobile robots that require significantly different design principles because they differ in the mechanism of motion: aquatic (underwater exploration), terrestrial (cars) and aerial (drones). Again, the classification is not strict, for example, there are amphibious robots that move in both water and on the ground. Robots for these three environments can be further divided into subclasses: terrestrial robots can have legs or wheels or tracks, and aerial robots can be lighter-than-air balloons or heavier-than-air aircraft, which are in turn divided into fixed-wing and rotary-wing (helicopters). Robots can be classified by intended application field and the tasks they perform (Figure 1.1). We mentioned industrial robots which work in well-defined environments.

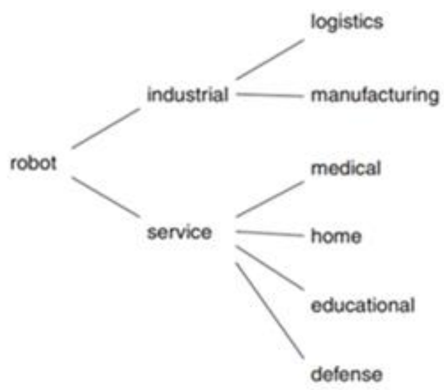

Figure 1-1 : Classification of robots by application field

1.5. Types of mobile robots:

1.5.1. Autonomous mobile robots:

Broadly speaking, an autonomous mobile robot (AMR) is any robot that can understand and move through its environment without being overseen directly by an operator or on a fixed predetermined path. AMRs have an array of sophisticated sensors that enable them to understand and interpret their environment, which helps them to perform their task in the most efficient manner and path possible, navigating around fixed obstructions (building, racks, work stations, etc.) and variable obstructions (such as people, lift trucks, and debris) [5].

Although they are still a relatively young technology, AMRs have already branched off into a number of distinct varieties, each of which is better suited to perform a specific type of action [5].

For this reason, when discussions about AMRs take place, they tend to be focused on the application that the technology is meant to perform, rather than a particular name or model [5].

Typically, AMRs can be split into three (3) broad buckets:

AMRs that move inventory within a facility.

AMRs that assist in the picking process.

AMRs that are a flexible sortation solution.

1.5.2. Industrial Robots:

An industrial robot is a robot system used for manufacturing. Industrial robots are automated, programmable and capable of movement on three or more axes [6].

Design must ensure that moving parts are not a danger to the user. The advantage of humans working with robots is that each can perform what they do best: the robots perform repetitive or dangerous tasks, while humans perform more complex steps and define the overall tasks of the robot, since they are quick to recognize errors and opportunities for optimization [7].

In the year 2020, an estimated 1.64 million industrial robots were in operation worldwide according to the International Federation of Robotics (IFR) (Figure 1.2) [8].

There are six main types of industrial robots: Cartesian, SCARA, cylindrical, delta, polar and vertically articulated. However, there are several additional types of robot configurations. Each of these types offers a different joint configuration. The joints in the arm are referred to as axes [9].

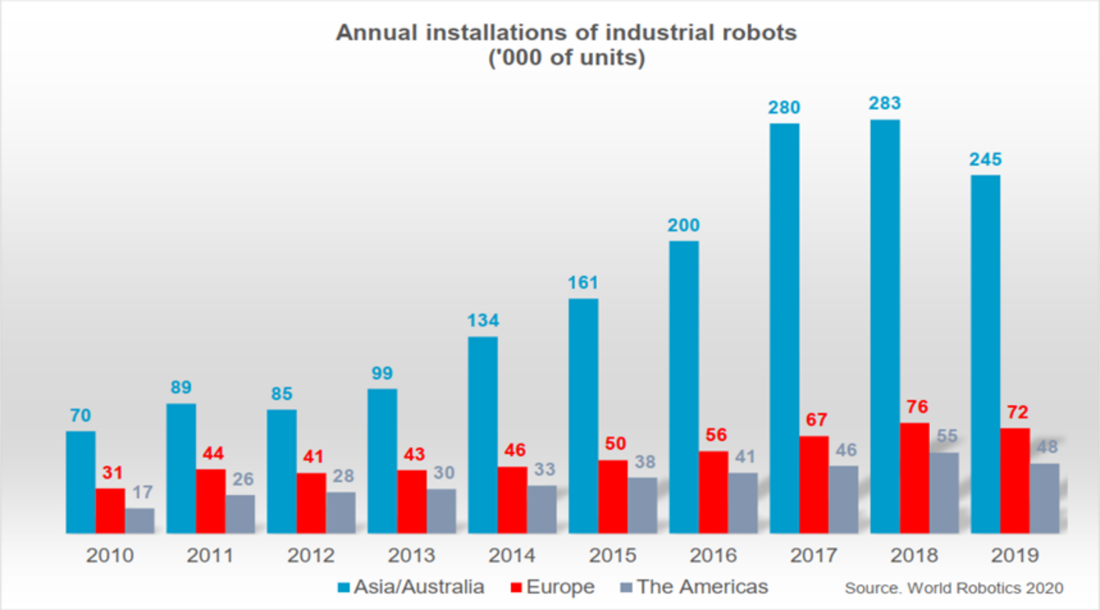

Figure 1-2: Annual installation of industrial robots.

1.5.2.a. Robotic Arm:

Robot manipulators are a very common and familiar type of robot. We are used to seeing pictures or video of them at work in factories doing jobs such as assembly, welding and handling tasks, or even in operating rooms doing surgery. The first robot manipulators started work nearly 60 years ago and have been enormously successful in practice – many millions of robot manipulators are working in the world today. Many products we buy have been assembled, packed or handled by a robot. Robot manipulators do not move through the world. They have a static base and therefore operate within a limited workspace (Figure 1.3) [10].

Figure 1-3: Industrial Robotic Arm.

1.5.3. Humanoid Robots:

A humanoid robot is a robot resembling the human body in shape. The design may be for functional purposes, such as interacting with human tools and environments, for experimental purposes, such as the study of bipedal locomotion, or for other purposes. In general, humanoid robots have a torso, a head, two arms, and two legs, though some humanoid robots may replicate only part of the body, for example, from the waist up. Some humanoid robots also have heads designed to replicate human facial features such as eyes and mouths. Androids are humanoid robots built to aesthetically resemble humans (Figure 1.4) [11].

Figure 1-4: Humanoid Robot.

Humanoid robots are now used as research tools in several scientific areas. Researchers study the human body structure and behavior (biomechanics) to build humanoid robots. On the other side, the attempt to simulate the human body leads to a better understanding of it. Human cognition is a field of study which is focused on how humans learn from sensory information in order to acquire perceptual and motor skills. This knowledge is used to develop computational models of human behavior, and it has been improving over time [12].

It has been suggested that very advanced robotics will facilitate the enhancement of ordinary humans [12].

1.5.4. Educational Robots:

Advances in the electronics and mechanics have made it possible to construct robots that are relatively inexpensive. Educational robots are used extensively in schools, both in classrooms and in extracurricular activities. The large number of educational robots makes it impossible to give a complete overview [13].

Educational robots enable students of all ages to become familiar with and deepen their knowledge of robotics and programming, while at the same time learning other cognitive skills (Figure 1.5) [13].

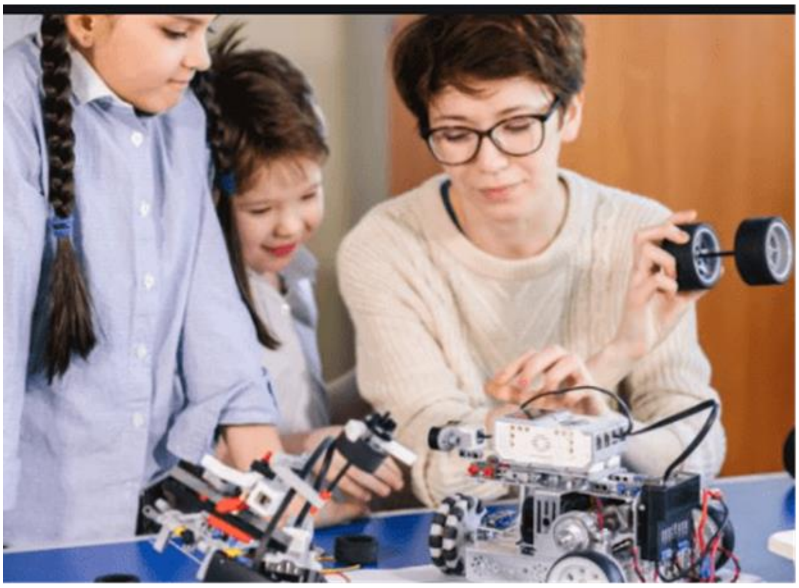

Figure 1-6 : Educational Robot.

1.5.5. Unmanned aerial vehicles (UAVs):

In recent years, there has been rapid development of autonomous unmanned aircraft equipped with autonomous control devices called unmanned aerial vehicles (UAVs) [14].

These have become known as “robotic aircraft,” and their use has become wide spread. They can be classified according to their application for military or civil use. There has been remarkable development of UAVs for military use. However, it can be said that the infinite possibilities of utilizing their outstanding characteristics for civil applications remain hidden [14].

1.6. Applications:

As more and more robots are designed for specific tasks, this method of classification becomes more relevant. For example, many robots are designed for assembly work, which may not be readily adaptable for other applications. They are termed as "assembly robots". For seam welding, some suppliers provide complete welding systems with the robot i.e. the welding equipment along with other material handling facilities like turntables, etc. as an integrated unit. Such an integrated robotic system is called a "welding robot" even though its discrete manipulator unit could be adapted to a variety of tasks. Some robots are specifically designed for heavy load manipulation, and are labeled as "heavy-duty robots" [15].

1.7. Conclusion:

In this chapter, we have summarized all the basic concepts necessary to understand the robotics as tools that replace humans in demanding and dangerous tasks.

We see that robots will have a significant place in the near future and that the links between the Man-Machine will be woven more and more easily, but to what extent ?

Robots have many advantages, they truly enhance human life, and everything becomes possible with a robot.

Chapter 2

Path planning and following

For mobile robots

2.1. Introduction:

Robotics has been a major success so far in the world of industrial manufacturing. Robot arms or manipulators are widely used in the industrial field with an estimated value of $2 billion dollar [16], this robot can move with great speed and precision allowing it to perform several varied tasks such as welding , painting and cutting(Figure 2.1).

There is a fundamental lack of mobility for these kinds of robots, Because his range of motion depends on where he is set up, On the other hand the mobile robot is able to move freely in its environment allow him to cover much bigger working area .

Figure 2-1: Robot arms during spot welding and surface.

2.2. Differential drive robot:

The differential drive is a two-wheeled drive system with independent actuators for each wheel. The name refers to the fact that the motion vector of the robot is sum of the independent wheel motions. The drive wheels are usually placed on each side of the robot and toward the front [17].

The mobile robotic control system can be divided into three categories. The first category is namely the sensor-based control-based approach. Such control system is emphasized on how to model the motion of a robot in a dynamic environment [18]. The control process to produce estimation and predictions of the mobile robot movement is based on information from sensor detection [19]. The intelligent control scheme is an approach that is most widely used. However, its results produce sub-optimal response, because the motion is only around the trajectory detection [20].

The second category is the approach of decomposed execution process using a path planning [21]. The control system regulates the movement of the mobile robot through the planned path, therefore it can move according to the target that has been set up. The environmental mapping is created for producing collision-free path. Such scheme of control is based on minimal distance, energy and time.

The third category presenting the optimization algorithm is developed for controlling the mobile robot with accurate trajectory. The controller design is based on mathematical model of mobile robot. The approach is for tracking the mobile robot errors between reference and actual trajectory .

However, the whole categories of the control system only operate when the condition of linear velocity is not zero. Therefore, a mobile robot is difficult to control, especially in the case of following the reference of trajectory in a short time with minimal errors. Nonlinear control approaches have been employed to solve this problem. Although the regulation problem is solved to track the mobile robot move to desired trajectory, but it found to yield slow asymptotic convergence. In order to obtain faster convergence, an alternative approach must be proposed .

2.3. Autonomous Systems:

The goal of an autonomous system is to operate within an environment without human interaction. The system needs to be able to understand itself and the world around it in order to determine which path to take and what the right commands are to get the system to follow that path (Figure 2.2).

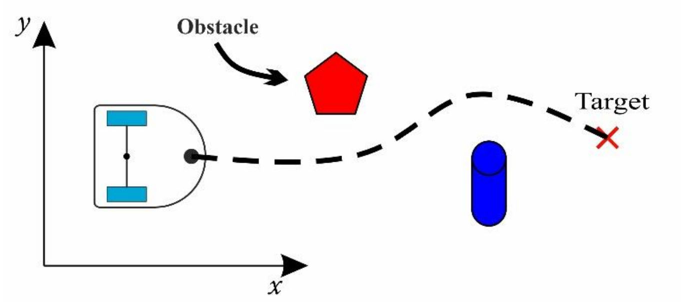

Figure 2-3: autonomous robot avoide obstacles.

We can divide these autonomous capacities into four main areas: sense, perception, planning and action (Figure 2.3).

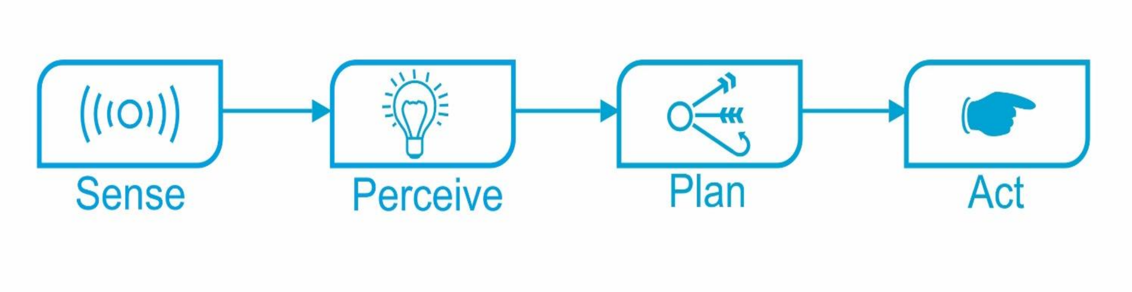

Figure 2-5 : autonomous system scheme.

2.3.1. **Sense:**

The sense step is when sensors are used to gather information about the state of the system and the state of the outer world. An automated driving system may measure its own state—for example, determining its position using GPS—but it also may measure the state of the environment with externally facing sensors such as radar, LIDAR, and vision cameras (Figure 2.4).

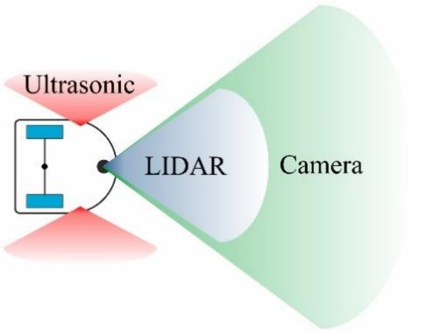

Figure 2-7: robot-sensing environment.

The purpose of this step is always the same to gather enough information so that the system can correctly interpret its situation and ultimately act on it, but simply gathering raw sensor data is not immediately useful. The system also has to make sense of the data.

2.3.2. **Perceive:**

Maybe the best way to understand the perception step is to think of a raw picture from the camera sensor of a vehicle. A picture can have several million pixels, each with 8 bits or more information in three bands of different colors. That is a large amount of data! In order to use that huge array of numbers, an algorithm or a human needs to come up with something of value. In this way, perception is more than the acquisition of information it is the act of interpreting it in a helpful quantity (Figure 2.5).

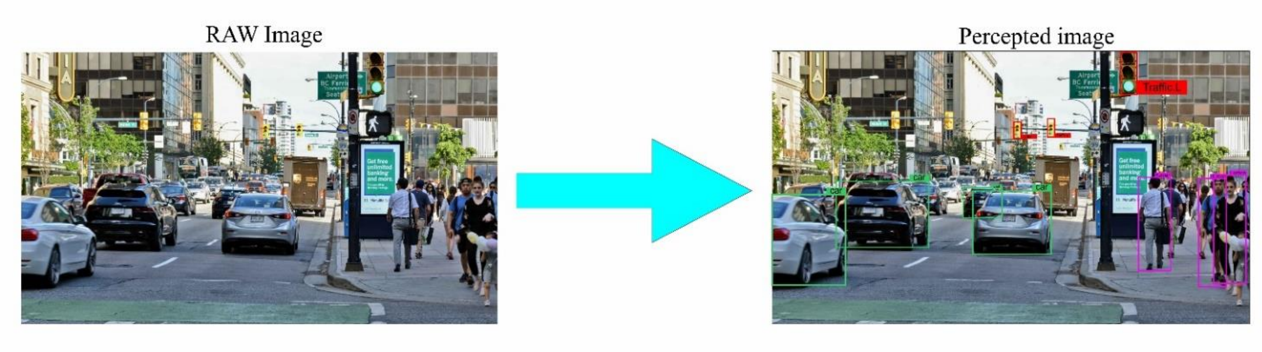

Figure 2-9: street RAW image after interpreting.

There are two different but equally important responsibilities for perception. It is responsible for self-awareness, which is perceiving your own state, and for situational awareness, which is perceiving other objects in the environment and tracking them.

A system must be capable of understanding itself and its environment in order to have sufficient information to make decisions. And the first step in deciding is to create a plan.

2.3.1. Decide and Plan:

The planning stage is when the stand-alone system determines what it would like to do and plots a course to get there (Figure 2.6).

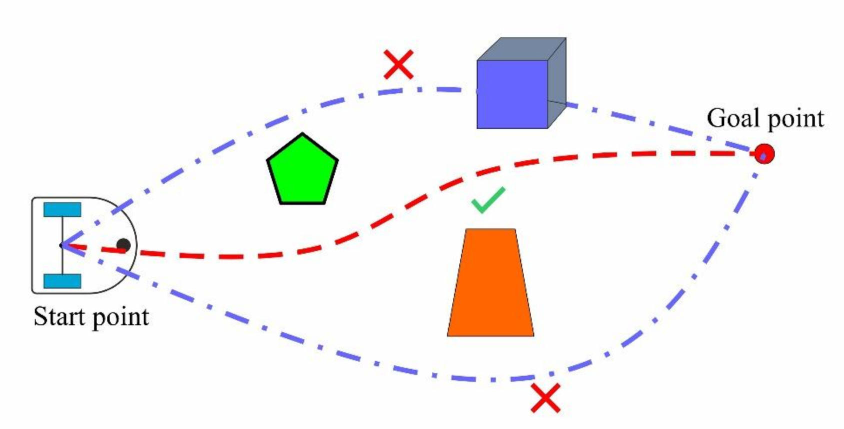

Figure 2-11: Robot plan for the optimal path.

2.3.3. Act:

This brings us to the last step: calculating and executing the actions necessary for the system to follow the plan. This is the job of the controller and the control system.

So, we can think of the plan as the reference signal that the controller uses to command the actuators and other control effectors in a way that manipulates the system to move along that path.

This controller is sensing the actual vehicle speed, comparing it with the reference speed, and then commanding the engine to produce the power necessary to drive the error to zero.

2.4. Path planning approaches:

Concepts for planning:

- Local planning and Global planning.

- planning based on information only on the sensing range of the robot.

- planning based on global map of the environment.

- **reacting and planning.**

- planning in response to sensory inputs (in action).

- planning for an entire event (start to goal).

2.4.1. Comparative study on the various trajectory-planning algorithms:

Several scientists have recently studied the optimization of trajectory planning and the problems of detecting obstacles.

A number of algorithms can be used and manipulated in several ways in order to use them for path planning of robots. Here is an overview of their characteristics:

2.4.1.a. Potential field:

- Difficult to implement for a real-world application.

- Poor performance in narrow passages.

- Poor performance in a dynamic environment.

- Prone to get stuck in local minima situations.

- Problems in dealing with the symmetrical obstacles.

2.4.1.b. Floyd–Warshall:

- It is a subset of Dynamic Programming with optimal substructures

- It can lead to an optimal answer in a grid-based map.

- Capable of solving hard path-planning problem.

- Due to high computational complexity, not a suggested method for on board implementation.

2.4.1.c. Genetic algorithm:

- Highly capable heuristic method.

- The computation time and the complexity of the algorithm are highly dependent on the implementation of the algorithm.

2.4.1.d. A* algorithm:

- It can provide a general heuristic approach for the search of an optimal path.

- It can potentially search a huge area of the map.

- The solution time would benefit from a static environment.

- Based on the scenarios, may lead to optimal or near-optimal solutions.

2.4.2. Potential Field algorithm:

A potential field is a physical field that respects the Laplace formula. Electric, magnetic and gravitational fields are a few common examples of possible fields. A potential field algorithm uses the artificial potential field to control a robot in a given space. For our convenience, we consider a space to divide in a grid of cells with obstacles and a goal node (Figure 2.7).

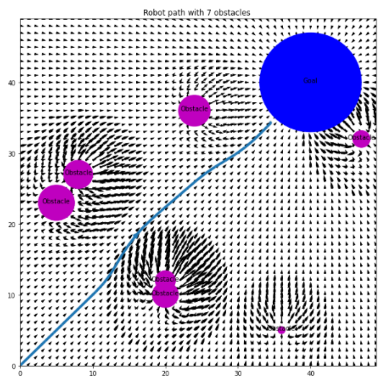

Figure 2-13 : Potential field with seven obstacles.

The algorithm assigns an artificial potential field to every point in the world using the potential field functions. The robot simulates from the highest potential to the lowest potential. In this case, the goal node has the lowest potential while the starting node will have the highest potential.

Two kinds of artificial potential fields are generated within the system: Attractive field and Repulsive fields (Figure 2.8).

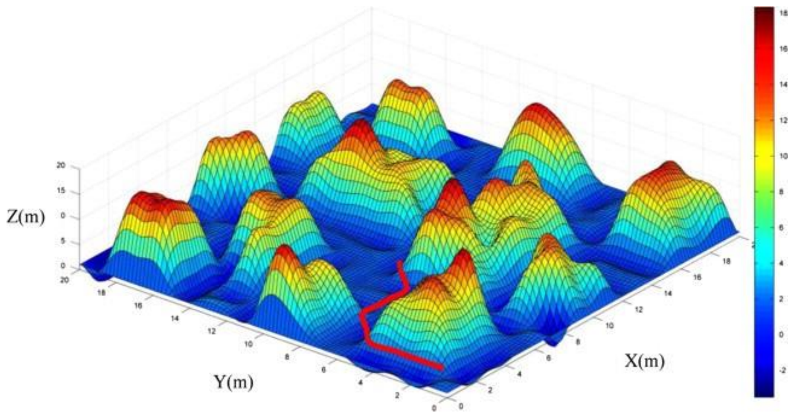

Figure 2-15 : artificial potential field.

The goal node exhibits an attractive field while the obstacles in the system produce repulsive fields.

Force of Attraction:

The following function can be used for generating the force generated by the goal node. Here, x and y are the coordinates of the starting node, (Xgoal, Ygoal) are the coordinates of the goal node and C is a constant.

$$P_g = C\sqrt{|x - x_{Goal}|^2 + |y - y_{Goal}|^2}$$

Force of repulsion:

Two kinds of repulsive forces are produced within the system.

- Repulsive forces by the boundaries:

This force remains uniform throughout the system and hence, does not affect the calculations; this force is useful in keeping the robot away from the boundaries. The following equation can be used to find the repulsive forces exhibited by the boundaries:

$$P_{HA} = \frac{1}{\delta + \sum_{i=1}^{s}(g_i + |g_i|)}$$

where g_i is a linear function that represents the boundary of the convex region, δ is a constant number with a small value and s is the number of boundary face segments.

-Repulsive forces by the obstacles:

The force of repulsion from obstacles can be calculated through the formula given below. Here, Pmax is the highest potential, (x0,y0) are the coordinates of the center of an obstacle and l is the side length of the obstacle:

$$P_{i.j} = \frac{P_{max}}{1+g}$$

Where

$$\begin{aligned} g(x,y) = {} & \left(x_0 - {}^{l}/_{2} - x\right) + \left|x_0 - {}^{l}/_{2} - x\right| \\ & + \left(x - x_0 - {}^{l}/_{2} + 1\right) + \left|x - x_0 - {}^{l}/_{2} + 1\right| \\ & + \left(y_0 - {}^{l}/_{2} - y\right) + \left|y_0 - {}^{l}/_{2} - y\right| \\ & + \left(y - y_0 - {}^{l}/_{2} + 1\right) + \left|y - y_0 - {}^{l}/_{2} + 1\right| \end{aligned}$$

The resultant force on the environment is:

$$P = P_0 + P_{Gol}$$

While

$$P_0 = max\{P_i\}$$

2.5. Path following approaches:

2.5.1. Pure Pursuit algorithm:

Pure pursuit is a path tracking algorithm. It computes the angular velocity command that moves the robot from its current position to reach some look-ahead point in front of the robot. The linear velocity is assumed constant, you can change the linear velocity of the robot at any point. The algorithm then moves the look-ahead point on the path based on the current position of the robot until the last point of the path. You can think of this as the robot constantly chasing a point in front of it. The property Look Ahead Distance decides how far the look-ahead point is placed.

2.5.1.a.Reference Coordinate System:

It is important to understand the reference coordinate frame used by the pure pursuit algorithm for its inputs and outputs. The figure below shows the reference coordinate system. The input waypoints are [x y] coordinates, which are used to compute the robot velocity commands. The robot's pose is input as a pose and orientation (theta) list of points as [x y theta]. The theta value is the angular orientation of the robot measured counterclockwise in radians from the x-axis (Figure 2.9).

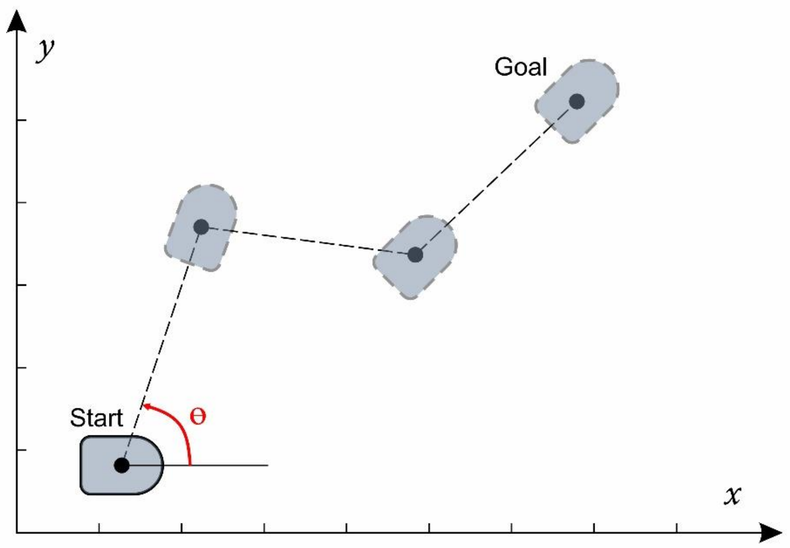

Figure 2-17 : robot following trajectory.

2.5.1.b.Look Ahead Distance:

The look ahead distance is how far along the path the robot should look from the current location to compute the angular velocity commands. The figure below shows the robot and the look-ahead point. As displayed in this image, note that the actual path does not match the direct line between waypoints (Figure 2.10).

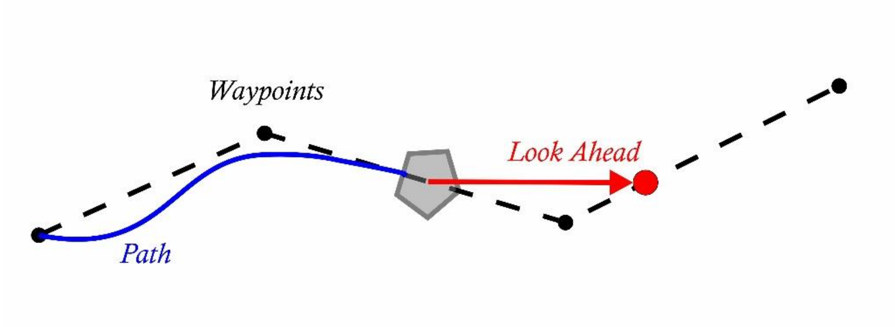

Figure 2-19 : pure pursuit algorithm in action.

Figure 2-20 : pure pursuit algorithm in action.

The effect of changing this parameter can change how your robot tracks the path and there are two major goals: regaining the path and maintaining the path. In order to quickly regain the path between waypoints, a small Look Ahead Distance will cause your robot to move quickly towards the path. However (Figure 2.11), the robot overshoots the path and oscillates along the desired path. In order to reduce the oscillations along the path, a larger look ahead distance can be chosen, however, it might result in larger curvatures near the corners (Figure 2.12).

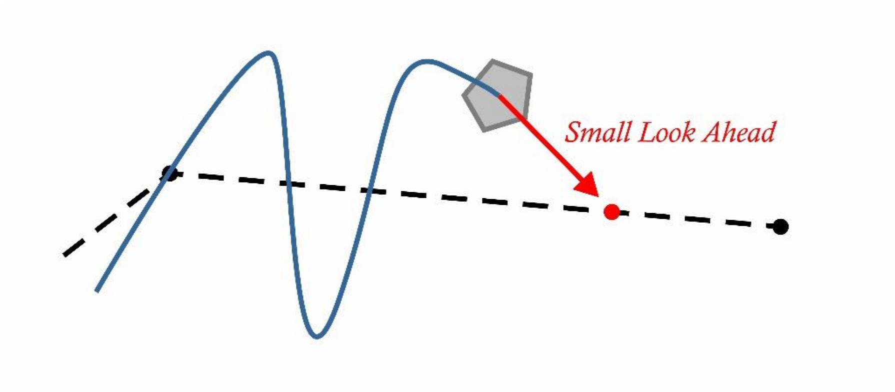

Figure 2-21 : Small look ahead distance.

The Look Ahead Distance property should be tuned for your application and robot system. Different linear and angular velocities will affect this response as well and should be considered for the path following controller.

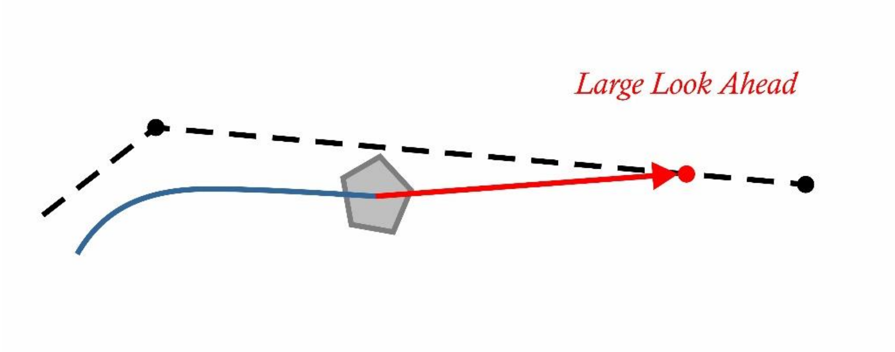

Figure 2-22 : Large look ahead distance.

2.5.1.c.Limitations:

There are a few limitations to note about this pure pursuit algorithm:

- As shown above, the controller cannot exactly follow direct paths between waypoints. Parameters must be tuned to optimize the performance and to converge to the path over time.

- This pure pursuit algorithm does not stabilize the robot at a point. In your application, a distance threshold for a goal location should be applied to stop the robot near the desired goal.

2.6. Conclusion:

In this chapter we mention the steps to make an autonomous system and the principles needed to accomplish such task.

The algorithm used in the robot is selected based on the field of application and the nature of the environment in which it operates.

Chapter 3

Mobile robot modeling

3.1. Introduction:

In this chapter we will give a detailed explanation about the modeling process of a differential drive mobile robot.

3.2. Definition of modeling:

The generation of a physical, conceptual, or mathematical representation of a real phenomenon that is difficult to observe directly. Scientific models are used to explain and predict the behavior of real objects or systems and are used in a variety of scientific disciplines, ranging from physics and chemistry to ecology and the Earth sciences. Although modeling is a central component of modern science, scientific models at best are approximations of the objects and systems that they represent they are not exact replicas. Thus, scientists constantly are working to improve and refine models.

3.3. Differential drive robot modeling:

The differential drive robot consists of two individually propelled wheels and a third wheel called castor wheel that can move freely in space (Figure 3.1). By adjusting the power [22].

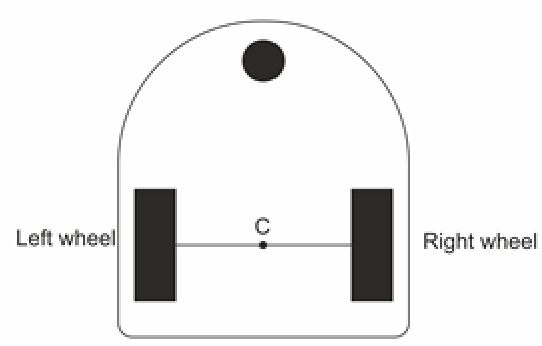

Figure 3-1 : Differential drive robot.

3.3.1. Differential drive robot type of motion:

The differential drive robot consists of a platform equipped with a front castor and a pair of rear co-axial drive wheels for isostatic equilibrium. Each of these drive wheels are independently driven by a DC motor, which is in turn energized by a control voltage.

By varying the power applied to the motors the differential wheeled mobile robot can be made to move in straight line or trace different trajectories like curves, circles etc.

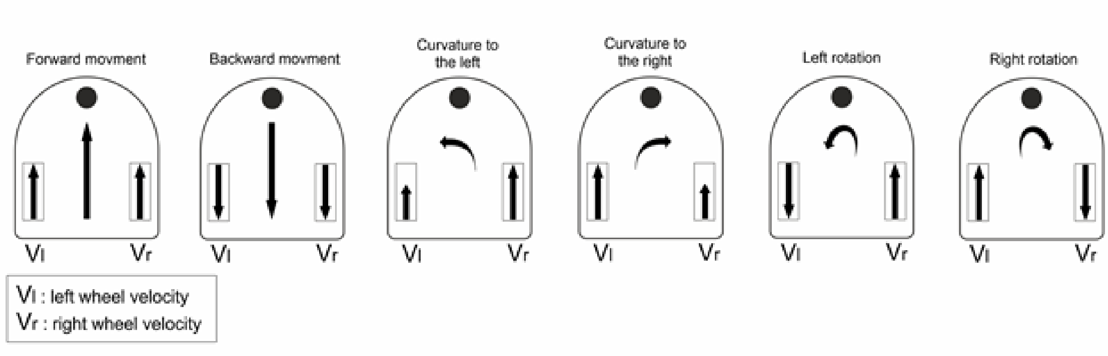

Figure 3-2 : Different types of movement.

3.3.2. Kinematics:

Kinematics is the basic study of how mechanical systems behave. In mobile robotics, we need to understand the mechanical behavior of the robot both in order to design appropriate mobile robots for tasks and to understand how to create control software for an instance of mobile robot hardware.

Calculation of the displacement and wheels velocity (Figure 3.3).

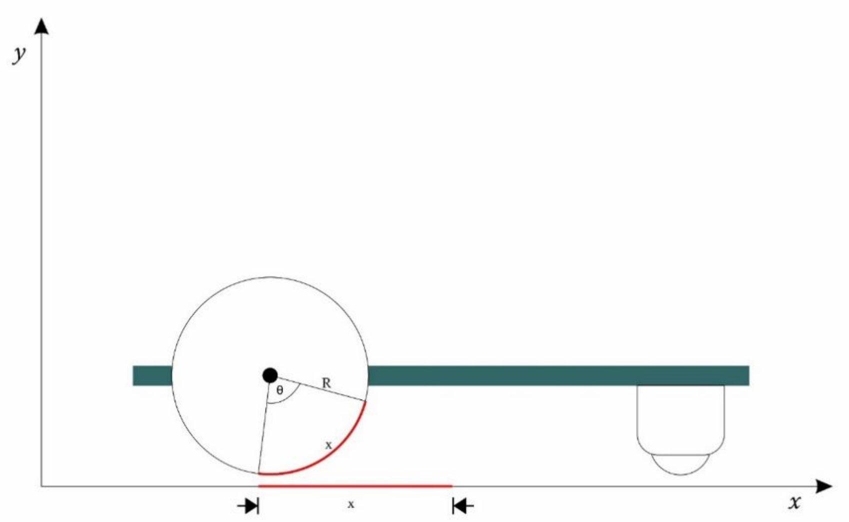

Figure 3-3 : Side view of mobile robot.

Calculate the arc length:

$$x = R\theta$$

Velocity for the left wheel:

$$\dot{x}_l = R\dot{\theta}_l$$

Velocity for the right wheel:

$$\dot{x}_r = R\dot{\theta}_r$$

Velocity for the center point:

For $\begin{cases} \dot{\theta}_r = 0 \\ \dot{\theta}_l \neq 0 \end{cases} \rightarrow \ \dot{x}_c = \frac{1}{2} R\dot{\theta}_l$

For $\begin{cases} \dot{\theta}_r \neq 0 \\ \dot{\theta}_l = 0 \end{cases} \rightarrow \ \dot{x}_c = \frac{1}{2} R\dot{\theta}_r$

For $\begin{cases} \dot{\theta}_r \neq 0 \\ \dot{\theta}_l \neq 0 \end{cases} \rightarrow \ \dot{x}_c = \frac{1}{2} R(\dot{\theta}_r + \dot{\theta}_l)$

Expression for linear velocity:

$$\dot{x} = \frac{1}{2} R(v_l + v_r)$$

$$\dot{y} = 0$$

$$\begin{pmatrix} \dot{x}_{c_0} \\ \dot{y}_{c_0} \end{pmatrix} = \begin{pmatrix} \cos & -\sin \\ \sin & \cos \end{pmatrix} \begin{pmatrix} \dot{x}_{c_1} \\ \dot{y}_{c_1} \end{pmatrix}$$

$$\begin{pmatrix} \dot{x}_{c_0} \\ \dot{y}_{c_0} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos & -\sin & 0 \\ \sin & \cos & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2} R(v_l + v_r) \\ 0 \\ \frac{R}{L}(v_r - v_l) \end{pmatrix}$$

These are Kinematics equations:

$$\left\{ \begin{array}{c} \dot{x} = \frac{R}{2}(v_l + v_r)\cos(\theta) \\ \\ \dot{y} = \frac{R}{2}(v_l + v_r)\sin(\theta) \\ \\ \dot{\theta} = \frac{R}{L}(v_r - v_l) \end{array} \right\}$$

3.3.3. The 'Unicycle' model

From (Figure 3.4) we have:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} v\cos(\theta) \\ v\sin(\theta) \\ \text{w} \end{pmatrix}$$

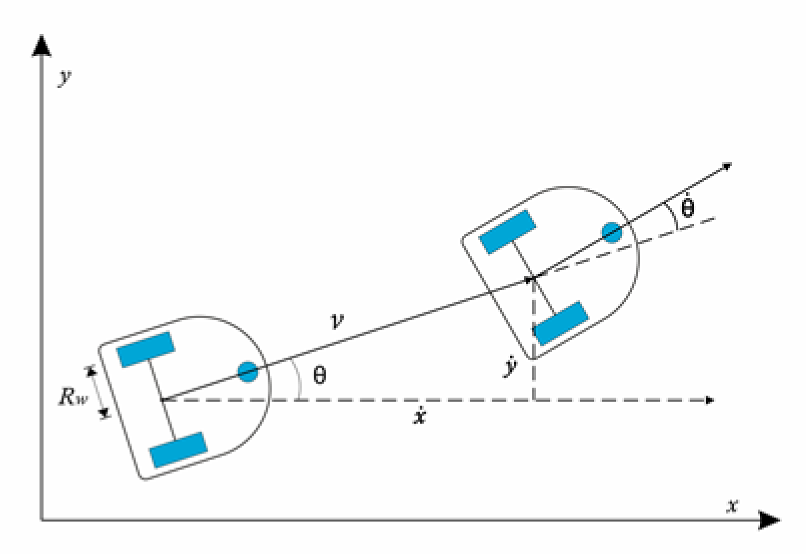

Figure 3-4 : Top view of a mobile robot.

From the Kinematics equations and the Unicycle model, we have:

$$v = \frac{R}{2}(v_r + v_l) \quad \rightarrow \quad \frac{2v}{R} = v_r + v_l$$

$$W = \frac{R}{L}(v_r - v_l) \quad \rightarrow \quad \frac{WL}{R} = v_r - v_l$$

3.3.4. Calculation of change in global coordinates

From (Figure 3.5) we can expressed the displacement for each wheel and the center point.

$$d_l = R\Delta\theta \qquad d_r = (R + L)\Delta\theta$$

$$d = (R + \tfrac{1}{2}L)\Delta\theta \qquad d = R\Delta\theta + \tfrac{1}{2}L\Delta\theta$$

$$d = d_l + \frac{d_r - d_l}{2}$$

$$d = \frac{d_l + d_r}{2}$$

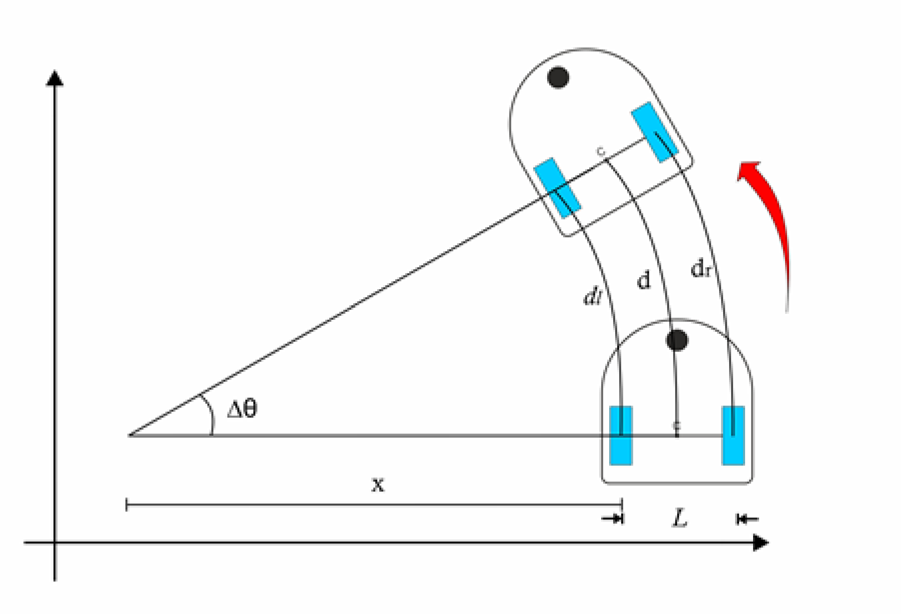

Figure 3-5 : Curve motion of a robot.

From (Figure 3.6) we can express the change in coordinates for the center point by the following terms:

$$\Delta x = d\cos(\theta)$$

$$\Delta y = d\sin(\theta)$$

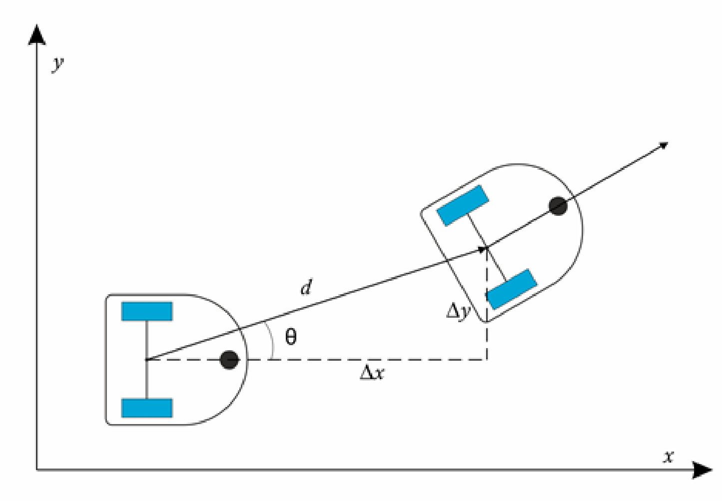

Figure 3-6 : Top view of a robot.

From 1 and 2 and the points mentioned in (Figure 3.7).

$$P' = P + \begin{pmatrix} d\cos(\theta) \\ d\sin(\theta) \\ \Delta\theta \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} d\cos(\theta) \\ d\sin(\theta) \\ \Delta\theta \end{pmatrix}$$

$$d = \frac{d_l + d_r}{2}$$

$$\Delta\theta = \frac{d_r - d_l}{L}$$

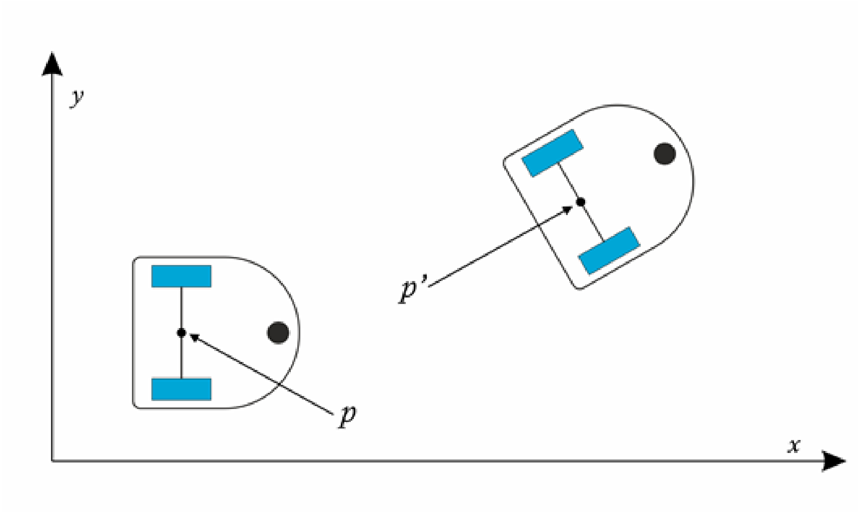

Figure 3-7 : Center point of the robot in global world.

Calculate encoder ticks and the left and right wheels displacement.

$$\Delta_{\text{tick}_r = \text{tick}'_r - tick_r} \quad , \quad \Delta_{\text{tick}_l = \text{tick}'_l - tick_l}$$

$$d_r = 2\pi R \frac{\Delta_{\text{tick}_r}}{N} \quad , \quad d_l = 2\pi R \frac{\Delta_{\text{tick}_l}}{N}$$

3.3.5. PID regulator:

The term PID stands for proportional integral derivative and it is one kind of device used to control different process variables like pressure, flow, temperature, and speed in industrial applications. In this controller, a control loop feedback device is used to regulate all the process variables.

This type of control is used to drive a system in the direction of an objective location otherwise level. It is almost everywhere for temperature control and used in scientific processes, automation & myriad chemical. In this controller, closed-loop feedback is used to maintain the real output from a method like close to the objective otherwise output at the fixe point if possible[24].

3.3.5.a.Proportional Control Action

In a controller, with proportional control action, there is a continuous relationship between the output of the controller (M) (Manipulated Variable) and Actuating Error Signal E (deviation) (Figure 3.8)[25].

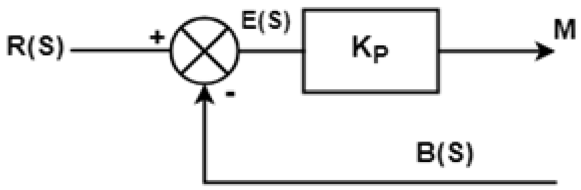

Figure 3-8 : Diagram of P regulator.

$$M(t) = K_{\mathrm{p}} * E(t)$$

Where, Kp is proportional gain and proportional sensitivity.

3.3.5.b. Integral Control Action

In a controller, with integral control action, the output of the controller is changed at a rate, which is proportional to the actuating error signal E(t) (Figure 3.9) [25].

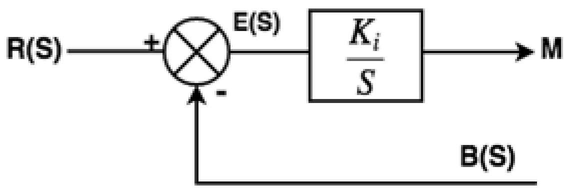

Figure 3-9 : Diagram of I regulator.

$$M(t) = K_i \int E(t) + M(0)$$

-Where K_i is Integral Control Action constant

M (0) =control output at which t = 0

3.3.5.c. Derivative Control action

In controller with derivative control action the output of the controller depends on the rate of change of the E (t) (Figure 3.10)[25].

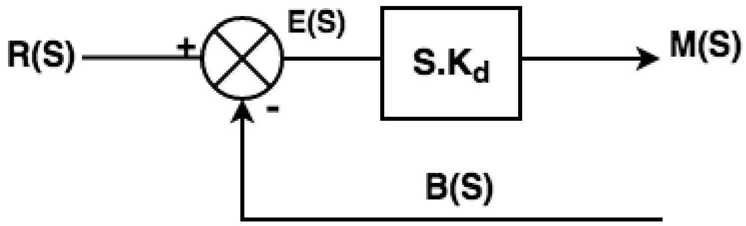

Figure 3-10 : Diagram of D regulator.

$$M(t) = K_d \frac{d}{dt} E(t)$$

-Where K_d is derivative control action constant.

3.3.6. Go to goal:

To drive a robot to goal location we need to calculate goal angel.

$$G_{Angle} = arctan2(Y_{goal} - y\, ,\, X_{goal} - x)$$

After that we need to calculate the error (e)

$$e = G_{Angle} - \theta$$

We feed the error to the PID algorithm for correction.

$$w = PID(e)$$

Then we calculate the left and right wheel velocity needed to drive the robot to the goal point.

$$V_r = \frac{2V_0 + Lw}{2R}$$

$$V_l = \frac{2V_0 - Lw}{2R}$$

V_r *and* V_l Are the output of our system.

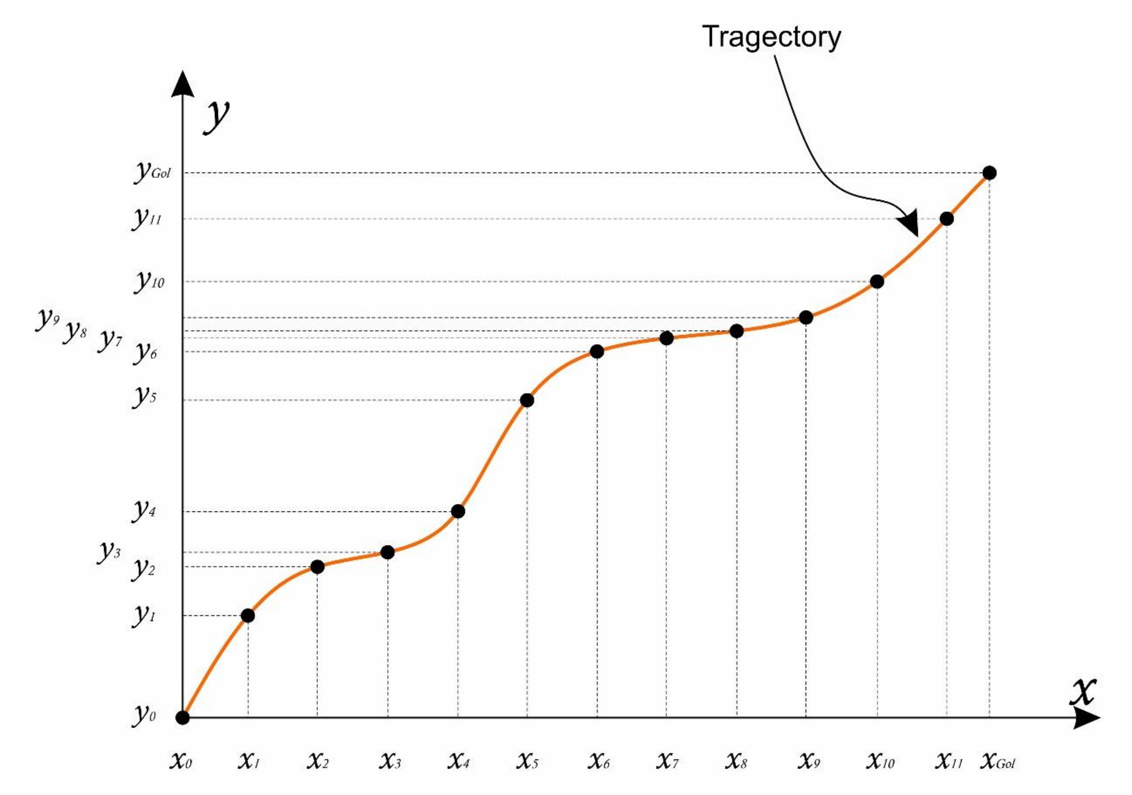

Figure 3-11 : Trajectory represented by a series of points.

Now we design a robot to go from point to another, to make follow a path we just need to feed the controller a vector of point represented in the global frame.

3.4. Tools:

3.4.1. Arduino board:

3.4.1. Definition:

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board[26].

The Arduino board is invented for the electronics students to use this in their projects. The Arduino boards are provided as open source that helps the user to build their projects and instruments according to their need. This electronic platform contains microcontrollers, connections, LEDs and many more. There are various types of Arduino boards present in the market that includes Arduino UNO, Red Board, LilyPad, Mega, and Leonardo. All these Arduino boards are different in specifications, features and uses and are used in different type of electronics project [27].

3.4.2. Arduino mega:

This board is considered as the microcontroller that uses the Atmega2560 in it. There are total 54 input pins and output pins in it in which 14 pins are of PWM output, 4 pins are of hardware port, 16 pins as analog inputs. The board also contain one USB connection, ICSP header, power jack and one REST pin [27].

Flash memory of 256KB size that uses to store the data in it. The Arduino Mega board can be attached to computer system via USB connection and power supply can be provided to board by using battery or AC to DC adapter. As the board has large number of pins fitted in it that make the board suitable for projects that requires more number of pins in it (Figure 3.11) [27].

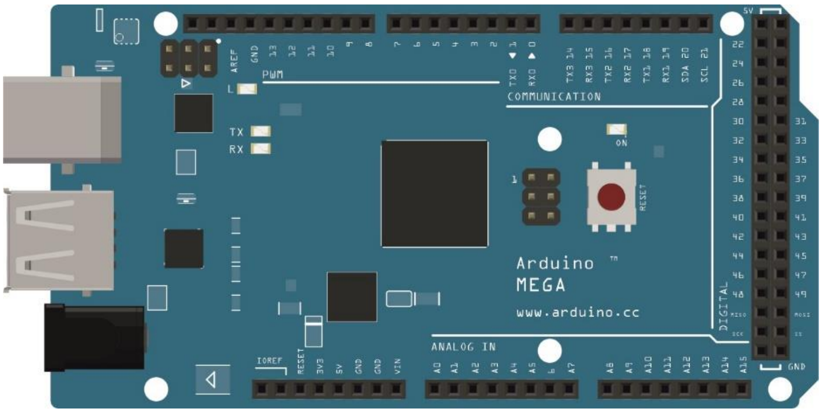

Figure 3-12 : Arduino Mega board.

3.4.2. Sensors:

3.4.2.a. Definition of sensors:

A sensor is a device that detects and responds to some type of input from the physical environment. The input can be light, heat, motion, moisture, pressure or any number of other environmental phenomena. The output is generally a signal that is converted to a human-readable display at the sensor location or transmitted electronically over a network for reading or further processing [28].

3.4.2.b. Wheel encoder:

Encoder sensors are a type of mechanical motion sensor that create a digital signal from a motion. It is an electro-mechanical device that provides users (commonly those in a motion control capacity) with information on position, velocity and direction. There are two main types of encoder: linear and rotary (Figure 3.12) [29].

It is used to measure rotational speed of motors or wheels. For light based encoders a series of markings or holes are made around the rotating disk that are used to break a light beam or change reflectivity that is detected using an infrared sensor [30].

Encoder sensors have become a widely used class of sensors where feedback information from a moving mechanical system is required [29].

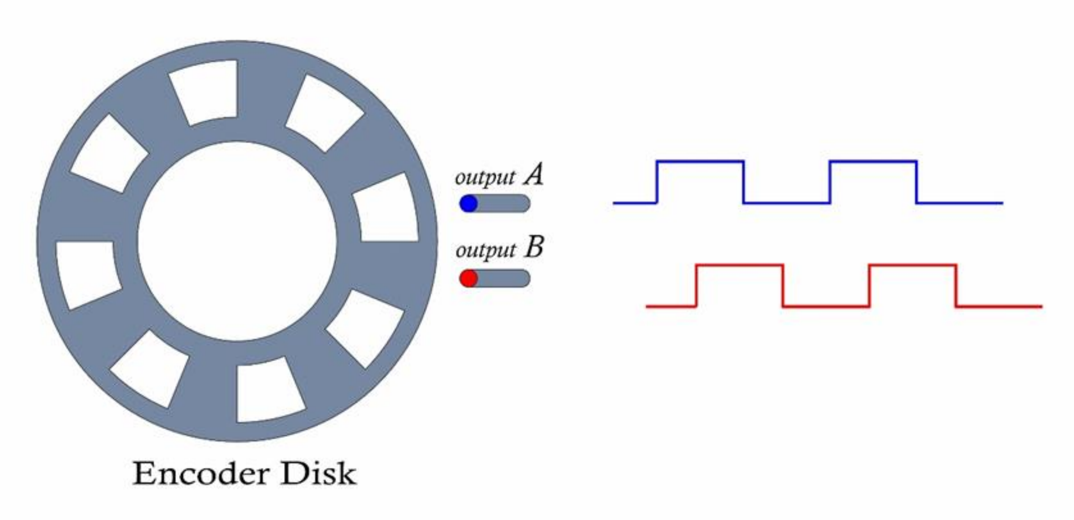

Figure 3-13 : Encoder disk and output signals.

3.4.3. Actuators

3.4.3.a.DC motors:

A DC motor is an electrical machine that converts electrical energy into mechanical energy. In a DC motor, the input electrical energy is the direct current which is transformed into the mechanical rotation.

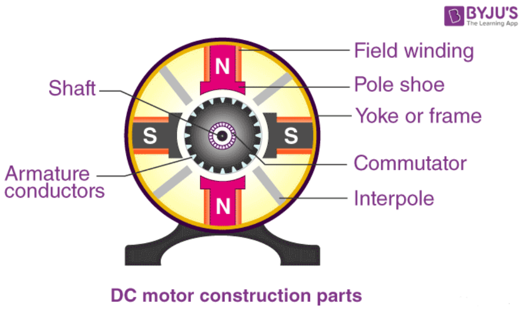

Figure 3-14 : DC motor parts.

3.4.3.b. Driver motor L298N:

This L298N Motor Driver Module is a high-power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05-5V regulator. L298N Module can control two DC motors with directional and speed control (Figure 3.14).

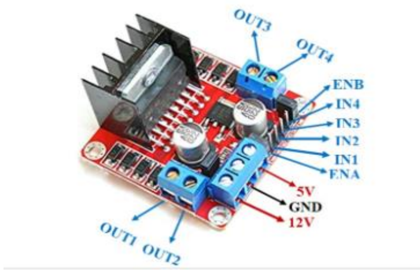

Figure 3-15 : L298N motor driver.

3.4.4. Electrical scheme:

(Figure 3.15) shows all the connections needed to link all the parts of the robot. Arduino mega, Motor driver L298N, two encoders attached to DC motor with mechanical redactor and the power supply.

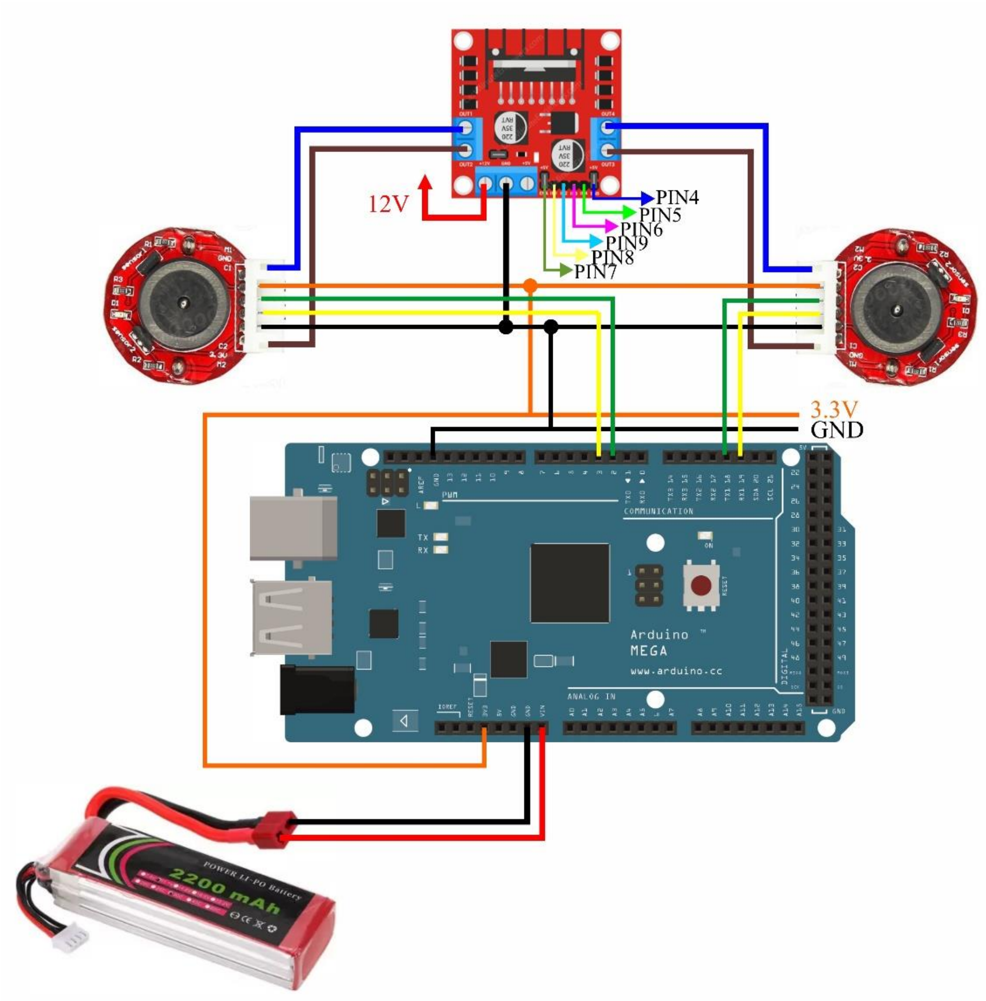

Figure 3-17 : Mobile robot circuit.

3.5. Conclusion:

In this chapter we explain the concept of modeling a mobile robot and the math behind it. and implementing a controller to drive this robot with a reasonable precision.

Also we give a brief explanation of different parts needed in the most of robots in general.

Chapter 4

Parameter tuning and results

4.1. Introduction:

In this chapter, we will tune and test the program in the real world and see how this parameter effect the system and how to get optimal results.

4.2. PID Tuning:

For tuning the PID gains (Kp, Ki, Kd) we will try every regulator alone (P, PD, PID) and observe the behavior of the system.

4.2.1. P regulator:

With a p gain of:

Kp = 100.

We see that the system is keep oscillating away from the desired point (Figure 4.1).

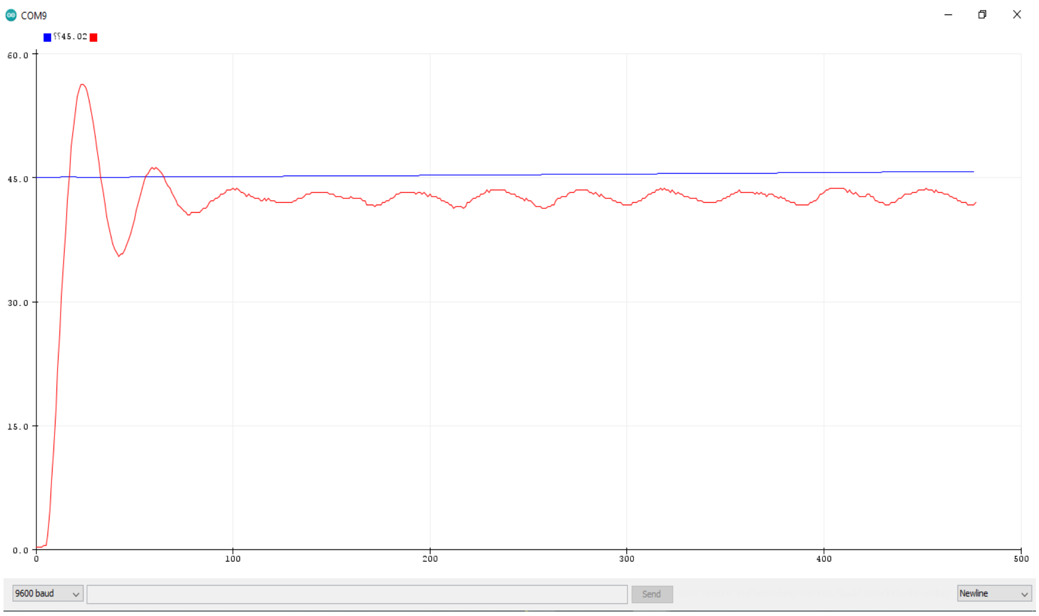

Figure 4-1 : P regulator with Kp = 100.

4.2.2. PD regulator:

With p and d gain of:

Kp = 100, Kd = 100.

We see that the system overshoot above the desired point and keep oscillating under it (Figure 4.2).

But the overshoot gets smaller the the p gain alone.

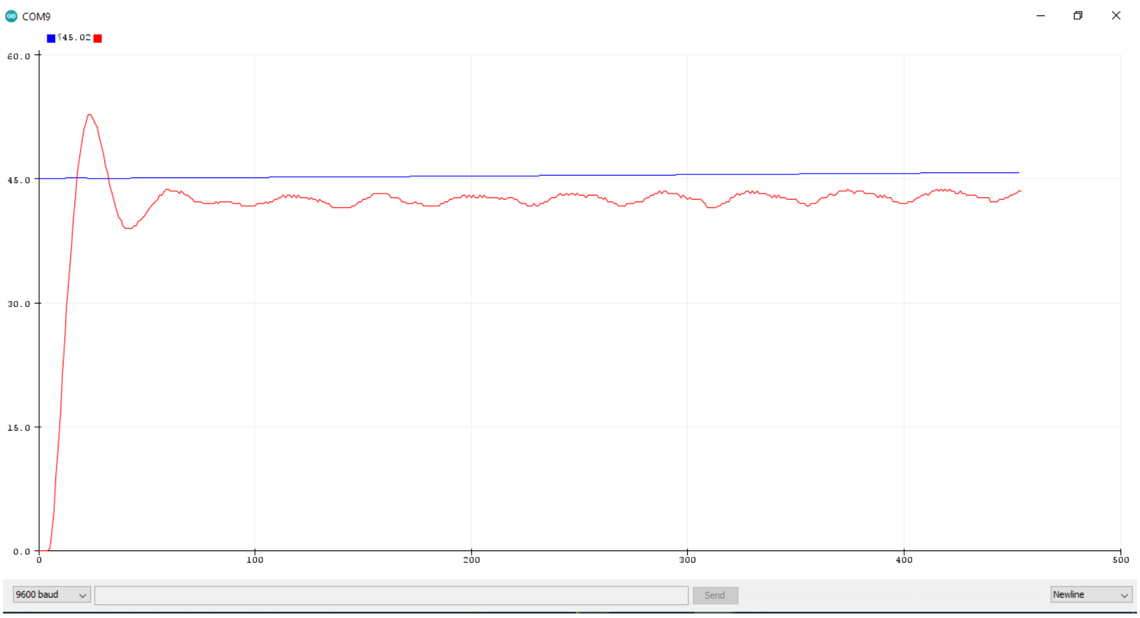

Figure 4-2 : PD regulator with Kp = 100, Kd = 100.

4.2.3. PID regulator:

With a PID gain of:

Kp = 15, Ki = 0.9, Kd = 100.

We see that the system keep oscillating around the desired point ,and the error get smaller after a short time (Figure 4.3).

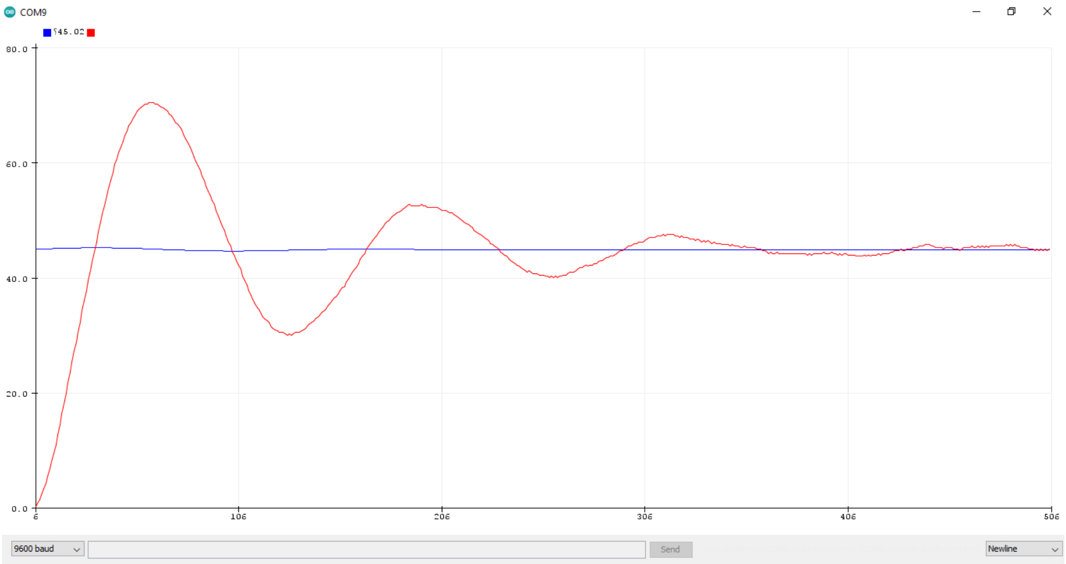

Figure 4-3 : PID regulator with Kp = 15 , Ki = 0.9 , Kd = 100.

We can improve the system by keep tuning the PID gains, after test we found that the following gains produce near optimal results.

Kp = 100, Ki = 0.4, Kd = 500.

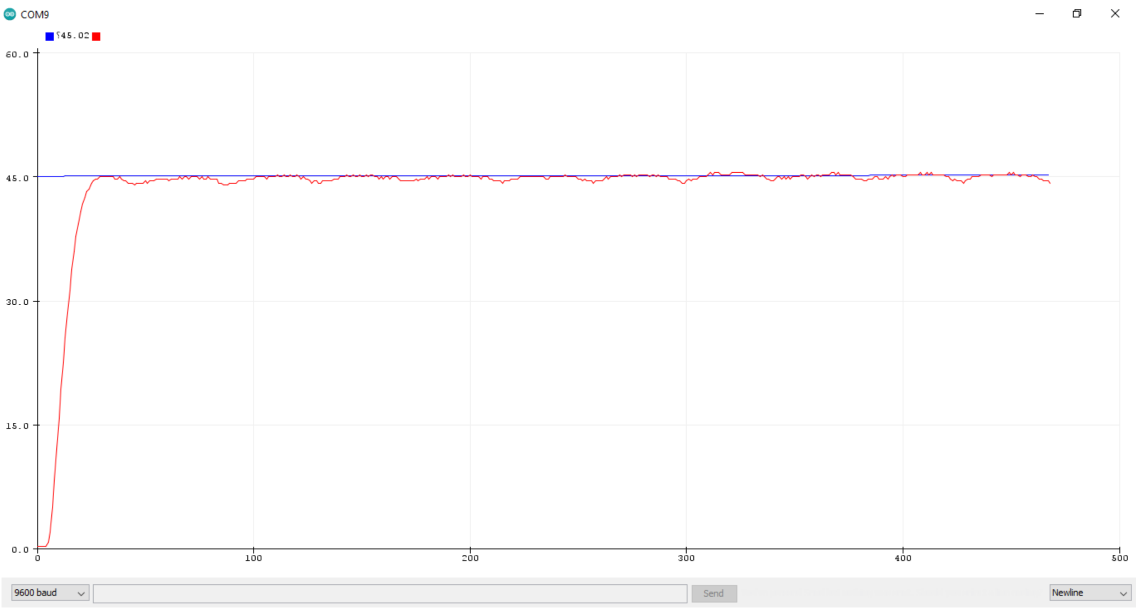

Figure 4-4 : PID regulator with Kp= 15 , Ki = 0.4 , Kd = 500.

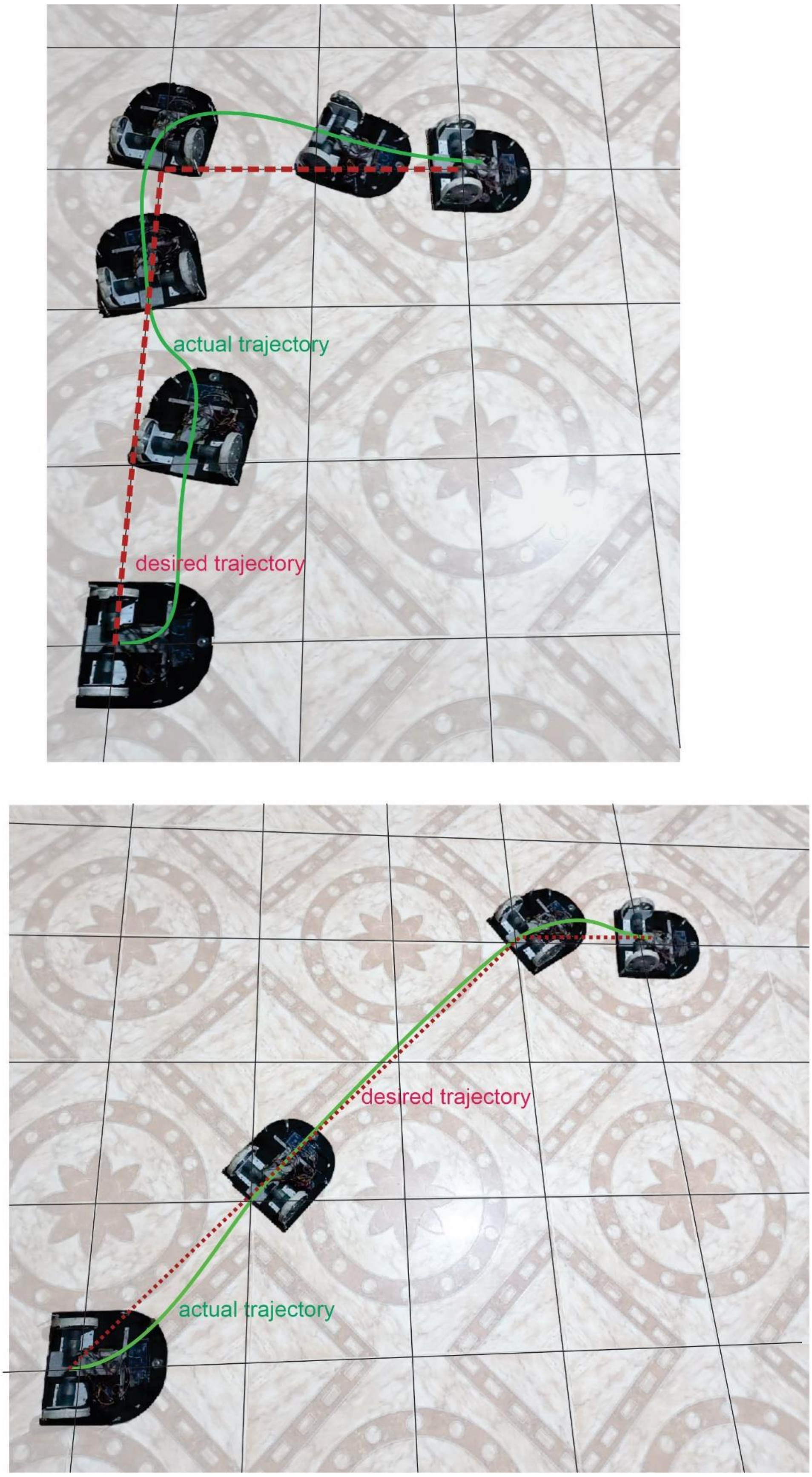

Figure 4-5 : Trajectory following examples.

4.3. Conclusion:

From the results shown in the above figures we can conclude that the robot expected movements of our robot is very satisfactory. These results have been obtained after extensive trial and errors settings of the PID controllers parameters.

General Conclusion:

This work has been achieved through different stages:

We used wheel encoder sensors to detect the amount of displacement for each wheel. After collecting the number of signals coming from the wheel encoder sensor, we calculate the distance traveled for each wheel in each program cycle inside the Arduino with the available processing, we can calculate the new coordinates of the robot and its orientation angle. This was followed by computing the angle difference between the robot and the target. Then we influence the drive of the wheels to make this difference as small as possible. With this, we have directed the robot toward the goal. If we make the points (targets) many too close to each other, we will have drawn a path for the robot.

The decision center of the robot was the Arduino Mega, and to drive the motors we used the L298N dual H-bridge.

Accurately measuring the distance between the two wheels and the wheel radius is necessary for the validity of the results. The results were satisfactory to the point that it is not without errors that must be taken into account. If the wheel moves suddenly, it will cause it to move without moving the robot, which will prevent the calculations from being wrong.

- Cumulative error because the measurements cannot be perfect, and with the length of the work period, the error accumulates to become unacceptable.

Arduino has limited capabilities. It is not possible to implement algorithms that work with artificial intelligence or computer vision. Below we suggest some hypotheses to improve the project.

- Using the wheel encoder sensor independent of the engine wheel to avoid drifting.

- Using a DSP instead of a microcontroller to increase the speed and efficiency of the algorithms

- Increasing the number and type of sensors to increase the robot's awareness of its surroundings.

- Using artificial intelligence algorithms instead of classic algorithms.

Bibliography:

[1] BALI, C, ABAIDI, H. « Réalisation d'un robot mobile avec évitement d'obstacle et trajectoire programmé ». Mémoire de Master en Génie des systèmes industriels. Université, Mohamed Khider de Biskra. 2012

[2] PhD, Renato M.E. Sabbatini. "Sabbatini, RME: An Imitation of Life: The First Robots".

[3] Waurzyniak, Patrick (2006). "Masters of Manufacturing: Joseph F. Engelberger". Society of Manufacturing Engineers. 137 (1). Archived from the original on 9 November 2011.

[4] M. Ben-Ari and F. Mondada, Elements of Robotics,page 1(15/311).

[5] https://www.conveyco.com/blog/types-and-applications-of-amrs/

[6] https://web.archive.org/web/20160617031837/https://www.iso.org/obp/ui/. Archived from the original on 2016-06-17.

[7] M. Ben-Ari and F. Mondada, Elements of Robotics,page 4(18/311).

[8] "Worries about premature industrialisation". The Economist. Archived from the original on 2017-10-21. Retrieved 2017-10-21.

[9] https://www.robots.com/faq/what-are-the-main-types-of-robots.

[10] Fundamental Algorithms in MATLAB® Robotics,Vision and Control by Peter Corke.

[11] https://www.asme.org/topics-resources/content/10-humanoid-robots-of-2020.

[12] https://en.wikipedia.org/wiki/Humanoid_robot.

[13] https://link.springer.com/chapter/10.1007/978-3-319-62533-1_1.

[14] Kenzo Nonami - Farid Kendoul - Satoshi Suzuki - Wei Wang - Daisuke Nakazawa Autonomous Flying Robots.

[15] Hunt, V. Daniel (1985). "Smart Robots". Smart Robots: A Handbook of Intelligent Robotic Systems. Chapman and Hall. p. 141. ISBN 978-1-4613-2533-8.

[16] Introduction to Autonomous Mobile Robots by Roland SIEGWART and Illah R. NOURBAKHSH.

[17] https://groups.csail.mit.edu/drl/courses/cs54-2001s/diffdrive.html

[18] Fiorini, P., & Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. The International Journal of Robotics Research, 17(7), 760-772.

[19] Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. Computer, 22(6), 46-57.

[20] Ge, Shuzhi S., and Yun J. Cui. "Dynamic motion planning for mobile robots using potential field method." Autonomous Robots 13.3 (2002): 207-222.

[23] Modelling and trajectory tracking of wheeled mobile robots N.Leenaa, K.K.Sajub

[24] Roland SIEGWART Illah R. NOURBAKHSH Introduction to Autonomous Mobile Robots (p-47).

[25] https://www.elprocus.com/the-working-of-a-pid-controller/

[26] https://www.javatpoint.com/control-system-p-pi-and-pid-controller

[27] https://learn.sparkfun.com/tutorials/what-is-an-arduino/all

[28] https://www.educba.com/types-of-arduino/

[29] https://www.techtarget.com/whatis/definition/sensor

[30] https://www.azosensors.com/article.aspx?ArticleID=1730

[31] https://www.robotgear.com.au/Category.aspx/Category/35-Encoders