People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

AMO University of Bouira

Faculty of Sciences and Applied Sciences

EG Department

Master Thesis

in Electrical engineering

Specialty: *Electronics of embedded systems*

Theme

Modeling and Control of a Robot

Realized By

- Naceri Mohamed

Discussed on : 01/07/2024

In front of the commission composed of:

MR REZKI Mohamed	Univ. Bouira	President
Mr ASRADJ Zahir	Univ. Bouira	Examiner
Mr ISSAOUNI Salim	Univ. Bouira	Advisor

2023/2024

Acknowledgment

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout our research work to complete the research successfully.

We would like to express our deep and sincere gratitude to our research supervisor, **Dr. S. ISSAOUNI** , for giving us the opportunity to do this research and providing invaluable guidance throughout the whole research.his vision, sincerity and motivation have deeply inspired us.

They have taught us the methodology to carry out the research and to present the research works as clearly as possible. It was a great privilege and honor to work and study under their guidance.

We are extremely grateful for what they have offered us. We would also like to thank them for their friendship, empathy, and great sense of humor.

We are extending our heartfelt thanks to our classmates, friends and doctors **Mr S. ISSAOUNI, Mr H. AIT ABBAS , Mr S. HAROUN, Mr A. FEKIK, and their family** for their acceptance and patience during these years of university.

We are extremely grateful to our parents for their love, prayers, caring and sacrifices for educating and preparing us for our future.

We are extremely grateful to our parents for their love, prayers, caring and sacrifices for educating and preparing us for our future.

We are very much thankful to our friends for their love, understanding, prayers and continuing support to complete this research work. Also, we express our thanks to our sisters for their support and valuable prayers.

We would like to say thanks to our teachers throughout our years of study in the University of Bouira, for their constant
encouragement.

we express our special thanks to them for their genuine support throughout this research work.

Finally, our thanks go to all the people who have supported us to complete the research work directly or indirectly.

Dedication

To My mother

the wisest and kindest, words just can't describe her beautiful soul. I can never thank her enough for being by my side through all my life,
and I will continue learning from her forever

My father

The most honest, humble and reliable. I can never forget the sacrifices he made throughout my whole life; I want to thank him for never letting me down.

My beloved two sisters and my brother for bearing me and supporting me and the huge moral support.

My mentor, Mr. S.ISSAOUNI for helping me during my years of study and for their moral support during my hard times.

My classmates who became brothers and sisters, for being by my side through thick and thin, who continually provide their moral, spiritual and emotional support.

A wise man once said: "The greatest good is what we do for one another"

Naceri Mohamed.

Abstract

Multi-degree of freedom robots are playing very important role in different application of automation. They are providing much more accuracy in carrying out a typical procedure as compared to the manual work done by human. In recent years the design, fabrication and development of robotic arms have been active research areas in robotics all around the world. This project describe a mechanical system, design concept and prototype implementation of a 6 DOF robotic arm, which should perform industrial task such as pick and place of fragile objects operation. This robot arm being controlled by micro-controller has base, shoulder, elbow, wrist rotation and a functional gripper. Gripper has been built as end-effector and is capable of grasping diverse objects within own workspace of the arm possible. PID controller is implemented on each motor. The microcontroller implement forward kinematics and position control of DC motors. The design aims to provide fine manipulation in performing pick and place task, while still maintaining the simplicity of design.

Résumé

Les robots à degrés de liberté multiples jouent un rôle très important dans différentes applications d'automatisation. Ils offrent une précision bien plus grande dans l'exécution d'une procédure typique par rapport au travail manuel effectué par l'homme. Ces dernières années, la conception, la fabrication et le développement de bras robotiques ont été des domaines de recherche actifs en robotique dans le monde entier. Ce projet décrit un système mécanique, un concept de conception et une mise en œuvre de prototype d'un bras robotique à 6 degrés de liberté (DOF), qui doit effectuer des tâches industrielles telles que la prise et la dépose d'objets fragiles. Ce bras robotique, contrôlé par un microcontrôleur, possède une base, une épaule, un coude, une rotation du poignet et une pince fonctionnelle. La pince a été construite comme effecteur final et est capable de saisir divers objets dans l'espace de travail du bras. Un contrôleur PID est implémenté sur chaque moteur. Le microcontrôleur met en œuvre la cinématique directe et le contrôle de position des moteurs à courant continu. La conception vise à offrir une manipulation fine pour effectuer des tâches de prise et de dépose, tout en maintenant la simplicité de la conception.

ملخص

تلعب الروبوتات متعددة درجات الحرية دوراً مهماً جداً في مختلف تطبيقات الأتمتة. فهي توفر دقة أكبر بكثير في تنفيذ إجراء معين مقارنةً بالعمل اليدوي الذي يقوم به الإنسان. في السنوات الأخيرة، أصبح تصميم وتصنيع وتطوير الأذرع الروبوتية مجالات بحث نشطة في الروبوتات حول العالم. يصف هذا المشروع نظاماً ميكانيكياً، ومفهوم التصميم وتنفيذ النموذج الأولي لذراع روبوتي ذو ستة درجات حرية، والذي يجب أن يقوم بمهام صناعية مثل التقاط ووضع الأشياء الهشة. يتم التحكم في هذا الذراع الروبوتي بواسطة متحكم دقيق، ولديه قاعدة وكتف وكوع ودوران معصم ومقبض وظيفي. تم بناء المقبض كعنصر نهاية وهو قادر على الإمساك بالأشياء المتنوعة داخل مساحة عمل الذراع الممكنة. يتم تنفيذ وحدة تحكم ب.ي.د على كل محرك. ينفذ المتحكم الدقيق الكينماتيكا الأمامية والتحكم في وضع المحركات التيار المستمر. يهدف التصميم إلى توفير معالجة دقيقة لأداء مهام الالتقاط والوضع، مع الحفاظ على بساطة التصميم.

Contents

List of Figures

List of Tables

Abbreviations List

D-H	Denavit-Hartenberg
DKM	Direct Kinematics Manipulator
DOF	Degrees of Freedom
FGM	Forward Geometric Model
IGM	Inverse Geometric Model
IKM	Indirect Kinematics Model
PID	Proportional, Integral, and Derivative
PUMA	Programmable Universal Machine for Assembly

General Introduction

Since the industrial revolution, a discipline has marked the evolution of the world technological: Robotics. The advent of robots in industry has made it possible to relieve man for repetitive and difficult work such as: moving heavy objects, tasks assemblies, micro welding, etc. Today, there are several types of robots designed for very specific tasks. The industrial robots currently in service are manipulator type robots. They are well established in the manufacturing processes modern and are used to increase the volume of production and improve the quality of the produced in the assembly lines of the automobile industry, they replace workers in arduous, repetitive or dangerous tasks (painting, welding, etc.)[1].

This evolution could not be achieved without the baggage of theoretical tools, bringing together the laws physics and mathematical equations, which was developed to describe the behavior of these systems which are robots. The work presented in this dissertation is in this framework. This involves modeling a manipulating robot, that is to say, developing the mathematical models making it possible to describe the static and dynamic behavior of robot[2].

In the first chapter, we present some general notions about robots, their classification and their areas of application.
Most of the work will be presented in the second chapter where the different models (geometric, kinematic and dynamic) of a PUMA560 robot with six degrees of freedom will be defined and elaborated. We end with a dynamic model simulation of the robot[3].

Chapter 1

General information on robotics

1.1 Introduction to Robotics

Robotics is a branch of science that involves the creation, alteration, and improvement of machines. The design, creation, and operation of robots. This field is sandwiched between mechanical and electronic components, Mechatronics, electrical engineering, computer science, electronics, AI, Nanotechnology and biological engineering. Additionally, Robotics involves the use of computers in the processing of information. for their regulation, sensation, and information gathering. “A Robot is a reprogrammable multifunctional manipulator designed to move materials, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks” [Robot Institute of America, RIA] OR

“A goal oriented machine that can sense, plan and act” [Peter Corke] These different fields of knowledge are employed to create machines that have the capacity to perform like human. Robots operate in different environments and have different purposes. Where the human cannot participate in any activity. Robots can be of any variety or shape, but some are designed to resemble humans in appearance. Such robots intend to emulate human movement, speech, cognition, and emotion primarily through walking, lifting, and speech. The individual’s preference. Several of today’s robotic devices are spontaneously involved, and help contribute to the domain of bio-inspired artificial intelligence. It’s a system that contains sensors that feedback, control systems, manipulators, and power[1].

Source and software together with the operating system are all necessary to complete a task. Designing, manufacturing, programming and testing a robotic could involve a combination of mechanical, Mechatronics, and Electrical Engineering, Computer Science, Electronics, AI, Nano-technology and Bioengineering. In rare instances, biology, medicine, and chemistry may be involved concerned[1].

1.2 Definition

1.2.1 Robot

The term "robot" comes from the Czech word "robota," which means "work." It is commonly stated that Sir Isaac Asimov was the first to use the term "robotics" in his 1940s novel. In the narrative, he suggested three laws to keep an eye on robot performance. Three philosophical rules for robotics were suggested by Sir Isaac Asimov and are still in use today [1]. The first law is known as the:

Robot First Law.

A robot cannot cause injury to a human being or, by remaining motionless, permit one to suffer harm.

Robot Second Law

A robot must always obey people, unless there is a higher-level rule that it should obey

Robot Third Law

Unless it violates a higher-level law, a robot must defend itself from damage.

Zeroth Law

A robot cannot hurt people or, by its actions, enable others to suffer damage.

1.2.2 Robotics

The term "robotics" comes from the English "robotics". It was invented by the novelist Isaac Asimov in 1942. It designates all the techniques and fields whose objective is the study, the design and production of robots.

1.3 History of Robots

We cite below some important dates which marked the history of evolution robots :

- In 1920: The word robot derived from Czech "robota" was introduced into literature scientific and technical [2].
- In 1959: first industrial robot, called Unimate (fig.1.1) was produced [3].

Figure 1.1: Robot Unimate [4]

- In 1961: the use of the Unimate industrial robot, in automobile production, started at General Motors [5].
- In 1971 Stanford University presented Stanford Arm [1].
- In 1973: the number of robots installed around the world reached 3000 [6].
- In 1973: The Hitachi Company developed the first robot with a processing system image for fixing bolts to a movable shape [6].
- In 1974: The ASEA company delivered the first all-electric, controlled industrial robots by microprocessor [6].
- In 1978: the Unimation company developed the PUMA robot (Programmable Universal Machine for Assembly)
- In 1979 Japan presents SCARA Robot [1]

- In 1996 Honda expose its Humanoid robot [1]
- In 1983: the number of robots in the world reached 66,000 [6].
- In 2004: the Motomana company presented a robot controller that can move four synchronized robots with up to 38 axes [6].
- In 2006: With KUKA, the German Aerospace Center (DLR) developed the third generation of its light robot (LBR) [6].
- In 2015: Robot sales increased to 253,748 units [7].

1.4 Objectives

To develop and operate a six degree-of-freedom robotic manipulator (articulated + spherical wrist) that can both grip and move delicate objects, adjusting their internal forces as needed. By using a force sensor—a tactile sensor that operates on the resistive principle—an actuator may keep a steady grasp on an item without endangering it.

To use a PID controller to accurately arrange things at the appropriate location by controlling the position of actuators.

The creation of a graphical user interface (GUI) that functions as an HMI, receiving user commands and sending them to a manipulator.

1.5 Classification of manipulator robots

Robotic systems are of interest to many civil and military fields. THE Major fields of application of robotics are:

1. Manufacturing production (machining, assembly, welding, polishing, forming, etc.)
2. Interventions in hostile environments (submarine, nuclear, exploration, planetary, etc.)
3. Transport systems for goods and people (intelligent vehicles, robotsmobile, etc.)
4. Help and assistance to people (personal robots, technical aids, etc.)

5. The many areas of health (surgery, rehabilitation, etc.).

6. Playfulness (toy robots)

They take very diverse forms from the point of view of their mechanical structure and their Order. Several types of 'generic' robots are illustrated in the figures that follow.

Figure 1.2: Humanoid robot - Scara type serial robot - Serial robot [8]

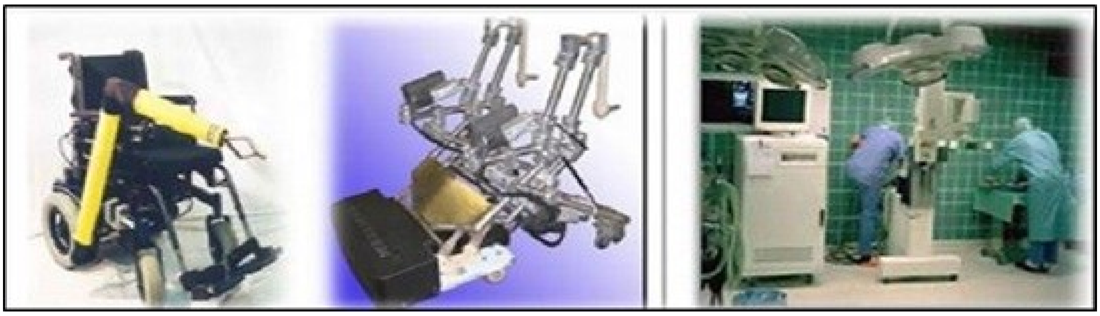

Figure 1.3: Chair equipped with an arm - Surgical robot walker//verticalist. [8]

Figure 1.4: Articulated hand-Robot dog-Robot hexapod [8]

1.6 Different categories of robots

1.6.1 Mobile robot

A mobile robot is a mechanical, electrical and computer system that act physically on its environment in order to achieve an objective assigned to it.

This machine is versatile and capable of adapting to certain variations in its conditions Operating. It is endowed with the function of perception, decision and action.

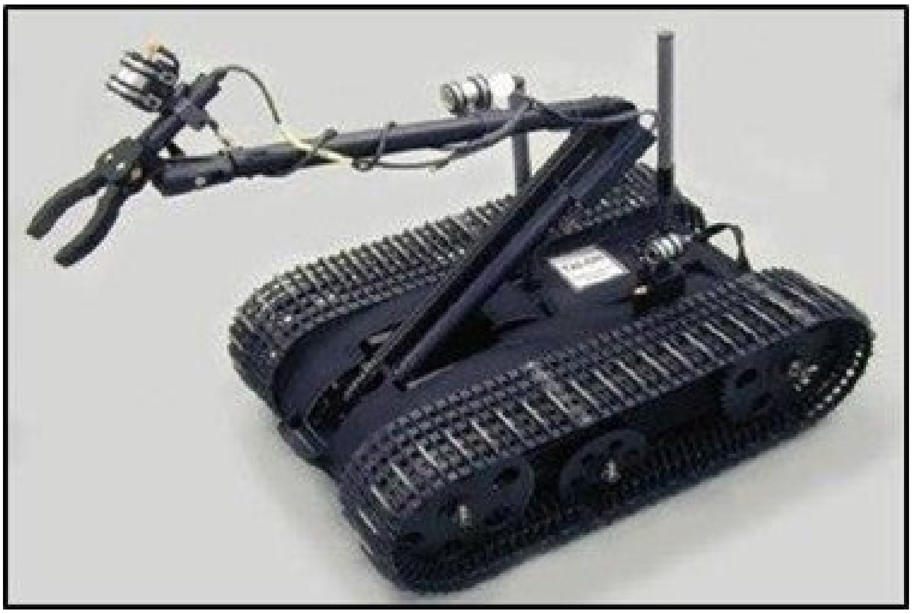

Figure 1.5: Robots mobiles [8]

1.6.2 SUB-MARINE ROBOT

An autonomous underwater robot is a robot that moves in the water in a manner autonomous unlike the ROV.
-ROV: A rov (which could be translated as “remotely controlled underwater vehicle”.

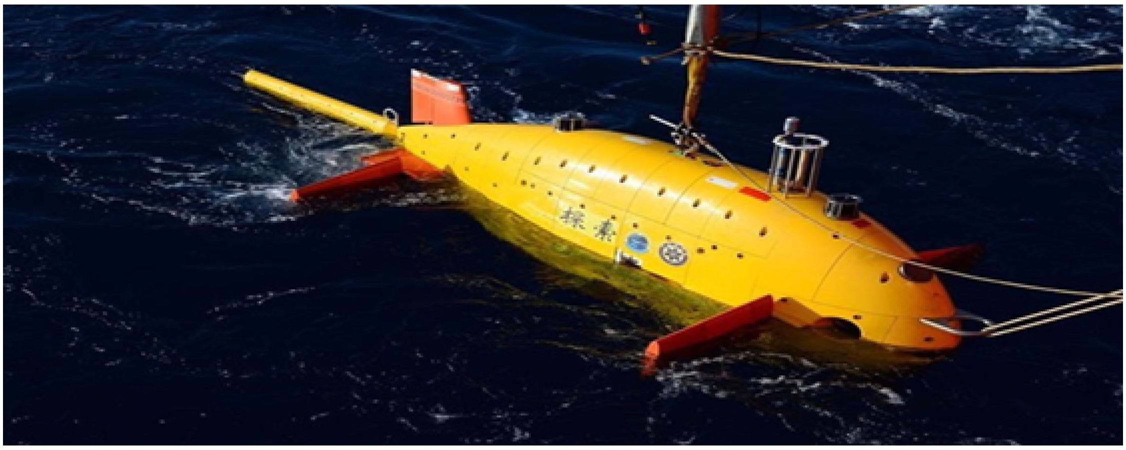

Figure 1.6: Sub-Marine robot [8]

1.6.3 Flying Robots

A drone designates an aircraft without a pilot on board, it can have civilian use or military. Drones are used for the benefit of the armed or security forces.

1.6.4 Industrial robotics

Industrial robotics is officially defined by los, as a control automatic, reprogrammable in three or more axes.
Typical applications include paint welding and assembly robots. Industrial robots are widely used in automobiles.

1.6.5 Domestic robotics

A home robot is a personal service robot used for household tasks. So far, there are only rare models. Domestic robots are used for example in washing dishes, cleaning and cooking.

Figure 1.7: Flying robot [8]

Figure 1.8: Industrial robots [8]

1.6.6 Medical robot

A medical robot is a robotic system used as part of an application therapeutic, for example during surgery or during a rehabilitation program neuromatrix.
Due to significant security constraints, this type of robot is generally equipped a low level of autonomy.

Figure 1.9: Robot domestique [8]

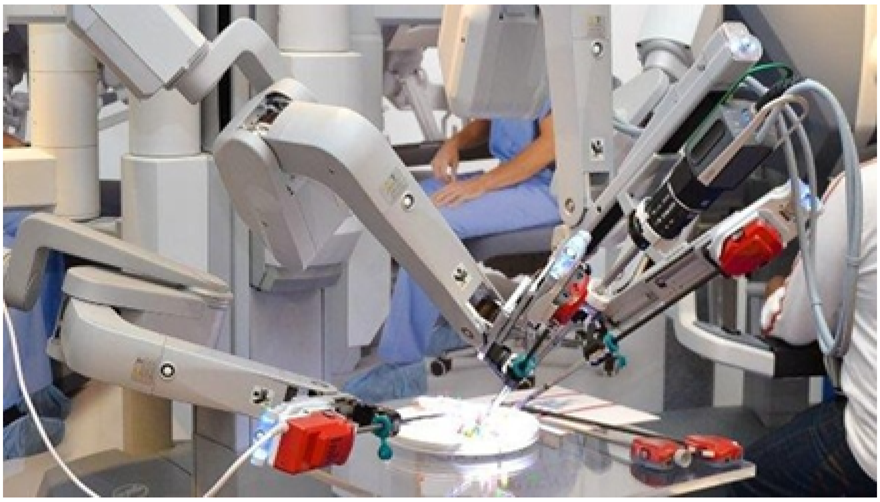

Figure 1.10: The medical robot [8]

1.6.7 Military robot

Robots are widely used by the army to simplify the lives of soldiers and limit injuries. Human losses, here are some categories:

- Mine-sweeping robots.
- Drones.
- Combat robots.

The first military robot being the 'Goliath' used by the German army during the World War 2.

Figure 1.11: Military robotics [8]

1.7 Manipulator arms

1.7.1 Definition

A manipulator arm is a generally programmable robot, with similar functions to a human arm.

The links of this manipulator are connected by axes allowing either rotational movement (as in an articulated robot) or translation (linear) of shift.

It can be stand-alone or manually controlled and can perform a variety of tasks with great precision.

The manipulator arms can be fixed or mobile (with or without wheels) and can be designed for industrial applications.

1.8 Mechanical Design

There are typically 4 main parts in a manipulator robot:

- **Terminal organ:** the terminal organ is a device attached to the mobile end of the mechanical structure (arm). It is intended to handle objects (clamping device,magnetic devices, etc.), or to transform them (tool, welding key, gun paint etc.).

A terminal organ can be multifunctional, in the sense that it can be equipped with several devices with different functionalities. It can also be monofunctional, but interchangeable. A robot, finally, perhaps multi-armed, each of arm carrying a different end organ. We will use the term organ indifferently terminal, gripper, tool or effector to name the attached interaction device at the moving end of the mechanical structure.

The articulated mechanical system (S.M.A) is a mechanism with a more structured structure or less close to that of the human arm. It allows you to replace, or extend, its action (the term "manipulator" implicitly excludes mobile robots autonomous). Its role is to bring the terminal organ into a situation (position and orientation) given, according to given speed and acceleration characteristics.Its architecture is a kinematic chain of bodies, generally rigid (orassumed as such), assembled by connections called articulations. Her motorization is carried out by electric, pneumatic or manipulators hydraulics, which transmit their movements to the joints by systems appropriate.

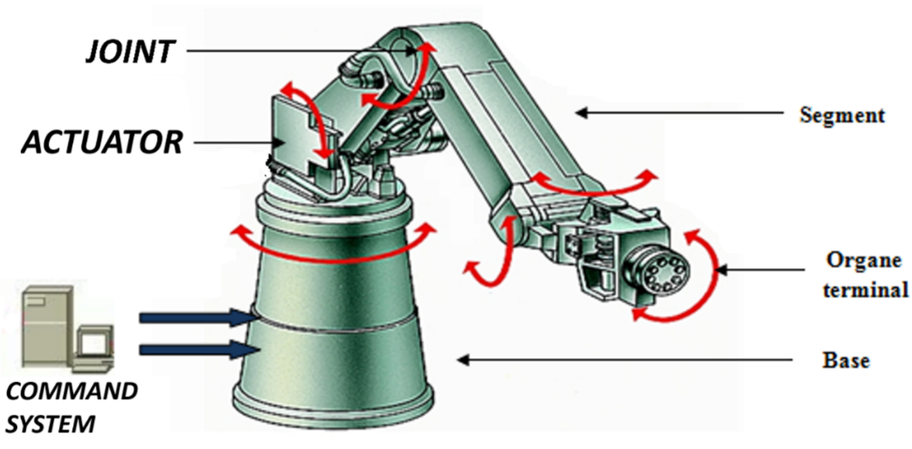

Figure 1.12: FIGURE Mechanical structure of the manipulator arms [9]

- **The actuators:** The S.M.A. comprises motors most often with transmissions, the assembly constitutes the actuators. Actuators use frequently permanent magnet, direct current, armature control.

- **Sensors:** the organs of perception make it possible to manage the relationships between the robot and its environment. So-called proprioceptive sensors when they measure the internal being of the robot (positions and speed of the joints) and exteroceptive when collecting information about the environment.

- **The control part:** summarizes the instructions for the servos controlling the actuators, based on the perception function and user orders.

Added to this is the human-machine interface through which the user program the tasks that the robot must perform

1.9 Application of Robotic Manipulators

Among their numerous uses are the following [2]:

1. They can be employed for military purposes.

2. They can carry out tasks in groups.

3. They are extensively employed in industry, with the proportion of robots to labor rising daily. Top global industries, like IBM and others, are entirely automated.

4. They are employed in hospitals to keep an eye on various activities. Both sewing and a variety of surgical procedures are easily accomplished by robots.

5. They have the ability to function as both a waiter and a cook in a restaurant.

6. Sports can be played by them and humans.

7. They are suitable for use in agriculture.

8. Nano-robots operate at the nanoscale, such as in medical settings to eliminate blood clots in vessels.

9. Distributed robotics

1.9.1 Michanical structure of the manipulator arms

The manipulator arm is made up of two main parts:

* Terminal organ: We will use the term prehensile terminal organ interchangeably, tool or effector to name the interaction device attached to the mobile end of the mechanical structure. The tasks assigned to the arms are very varied. For each specific operation or work, the terminal organ takes on a particular appearance.

* Supporting element: it is composed of a set of flexible or rigid bodies linked by joints, used to move the end organ from one configuration to another.

* The role of the carrier is to fix the position of the intersection point, denoted P, of the axes of the 3 last joints (center of the wrist); this position (P) only depends on the Configuration of solids (bodies) 1,2 and 3 (i.e., of the carrier),

* the wrist is intended for the orientation of the terminal organ (pliers, tool).

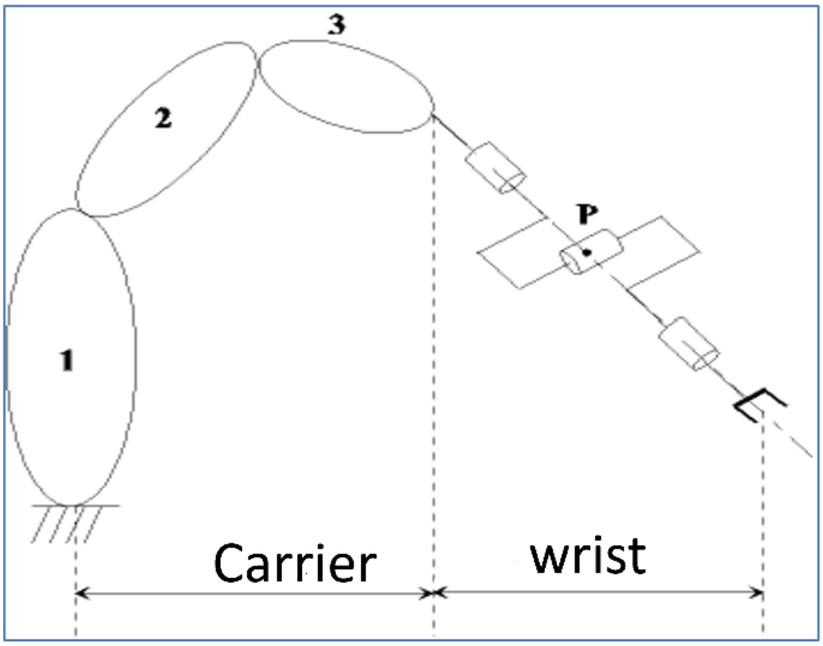

Figure 1.13: MECHANICAL DESIGN

1.10 Types of manipulator arm Robots

There are five basic configuration of robots which are:

- **Cartesian Configuration (PPP)**

  One popular kind of industrial robot is the Cartesian robot, sometimes referred to as the Cartesian coordinate robot. Each of its three "arms" operates along a linear axis of control. These axes are all at a straight angle to one another.

- **Cylindrical Configuration (RPP)**

  A cylindrical robot is made up of two prismatic joints and one revolute joint. mostly employed in cylindrical operations.

- **Spherical Configuration (RRP)**

  Stanford University showed a spherical arm with two revolute and one prismatic joint. Each of the three joints is perpendicular to the others.

- **Articulated/Anthropomorphic Configuration (RRR)**

  Anthropomorphic arms, also known as articulated or jointed arm robots, closely resemble human arms. There are three revolving joints in it.

- **SCARA Configuration (RRP)**

  The selective complaint articulated robotic assembly, or SCARA, is primarily made up of one prismatic joint and two revolute joints that are parallel to one another.

- **the ring bearer (RPR)**

  The RPR toroidal carrier is a type of parallel robot used for assembly tasks where the main direction of work is the vertical axis

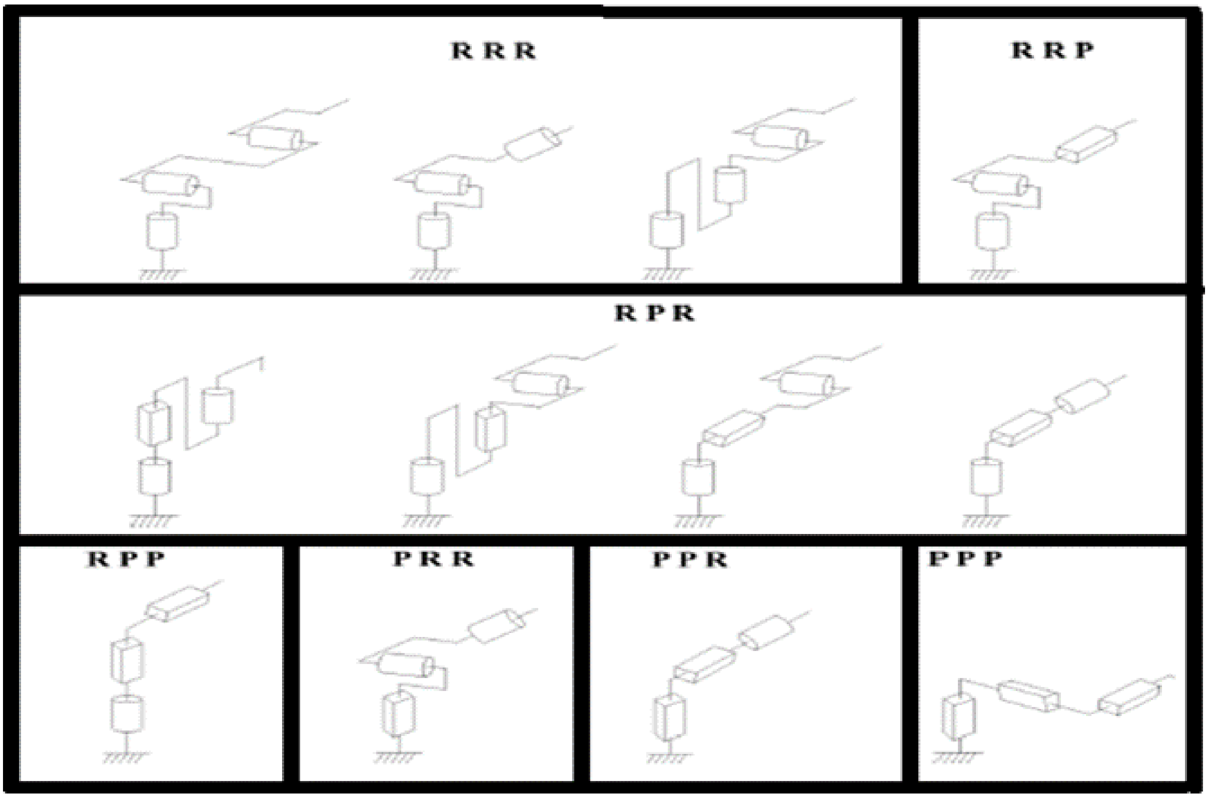

Figure 1.14: A set of solids connected together by bonds is called a mechanism [10]

These connections are highlighted with the aim of transforming or transmitting a movement. There are two types of mechanisms: open simple chain mechanisms, where the same bond or the same solid is never passed twice.
The most common system is complex chain type.
Systems comprising more than 2 links can be divided into 2 distinct categories.
Mechanisms are represented as structured trees, with the most precise ones being capable of handling heavy loads.

- The representation of the mechanism in the form of a kinematic diagram can be carried out in perspective or in projection.
- The graph, which is not normalized, is also used.

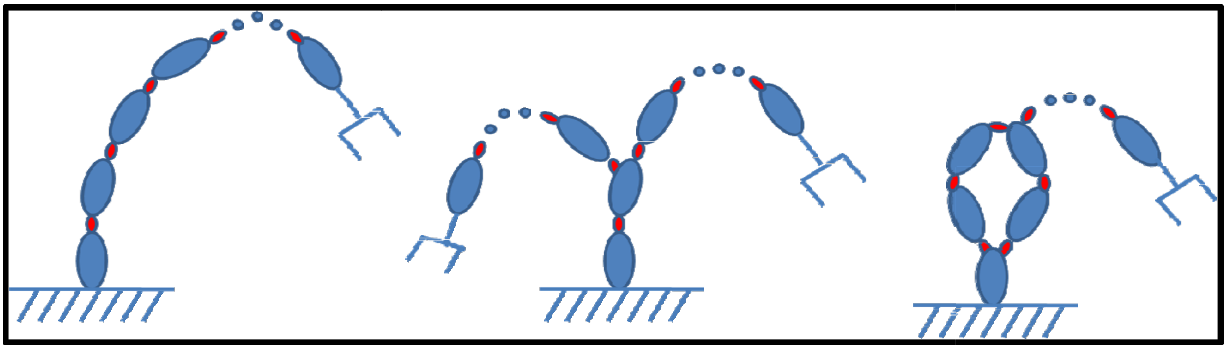

Figure 1.15: non-normalized graph of some mechanisms [11]

A robot has 2 basic parts [6]:

* **Holder**

  An articulated mechanical structure made up of the first 3 degrees of freedom of the frame. If P is the extremity and RO is the reference point connected to the frame, then the role of the carrier is to fix the position of P in RO.

* **wrist**

  It is used for the orientation of grippers or tools carried by the robot To calculate the different possible architectures, we only consider 2 parameters:

  (a) Joint type [rotary (R) or prismatic (P)].

  (b) The angle formed by two consecutive articulation axes (0° or 90°; except in very special cases, the consecutive axes of the robot are either parallel or perpendicular) Example SCARA robot.

  We agreed to call the first 3 d.o.f. Robot carrier. d.o.f. The remains form the wrist, characterized by much smaller dimensions and less mass.

1.11 Conclusion

In this chapter, we have given a general overview of robots and manipulator arms, the different constituents and mechanical structures as well as the terminology used in robot. In the next chapters we'll discuss the different types of controlling and modeling of 6DOF manipulator arm (PUMA 560) .

Chapter 2

Geometric Modeling

2.1 Introduction to Geometric Modeling

Geometric modeling is crucial in robotics for representing the spatial configuration of a robot, including the position and orientation of its links and joints. By using mathematical formulations like Denavit-Hartenberg parameters, we can systematically describe a robot's kinematics. This process allows us to determine the end effector's pose from the joint angles (forward kinematics) and to find the required joint angles for a desired end effector pose (inverse kinematics). Accurate geometric modeling is essential for precise and reliable robot movements in various applications.

The most used methods are:

- Denavit–Hartenberg

2.2 The Robot

The manipulator arm of the PUMA 560 system consists of five segments: base, shoulder, elbow, in-line wrist, orientation, and tool holder. These segments are interconnected by six rotational axes , each powered by a DC servomotor. The rotation of each joint is facilitated by a permanent magnet DC servomotor through a gearbox. Each motor is equipped with an incremental encoder, a potentiometer, and a 1/116 ratio reducer.

To ensure the proper functioning of the PUMA 560, precise control of the position and rotational speed of each joint is necessary. Positional feedback is provided by the encoders, while rotational speed data is computed by the robot's control computer.

The servomotors for the three main axes (axes 1, 2, and 3) are equipped with electromagnetic brakes. These brakes engage when the motor power is cut, keeping the robot arm in a fixed position. This safety mechanism prevents injuries or damage in the event of an unexpected power outage.

2.2.1 Control Computer

The control computer is the primary electrical component of the PUMA 560 system. It manages all communication to and from the robot's actuators, executing real-time calculations to generate control commands. The programming software, stored in the computer's central memory, interprets these commands, which are then transmitted by the controller to the robot's actuators.

With incremental encoders and potentiometers, the controller accurately receives positional data for each axis, allowing for closed-loop control of the robot's movements. [12].

2.2.2 Description of the PUMA 560 Robot

The chosen robot for this study is the PUMA 560 manipulator. PUMA robots are the most widely used assembly robots in the industry and are highly prevalent in academic settings. The PUMA (Programmable Universal Machine for Assembly) was originally designed by Vic Schienman and funded by General Motors and the Massachusetts Institute of Technology in the mid-1970s. It was manufactured for many years by Unimation, a company later acquired by Westinghouse and subsequently sold to Staubli, a leading Swiss robotics company.
The PUMA 560 system consists of two main components: the robotic manipulator arm and its control computer [12].

2.3 Denavit-Hartenberg Parameters

The Denavit-Hartenberg (D-H) parameters are widely used by roboticists to define the elementary homogeneous transformation matrices with a minimal set of parameters. These matrices enable the transition from the reference frame of one robot link to the next link in the kinematic chain. The links are assumed to be perfectly rigid, and the joints are considered ideal [13].

Each link i of the robot is assigned a reference frame R_i at the joint j where it connects with the previos link i-1. this reference frame is defined as follows:

- The axisZ_i is aligned with the axis of joint j.
- The axis is aligned with the common perpendicular to the axes Z_i and Z_{i-1} . If Z_i and Z_{i-1} are parallel or collinear, the choice of X_i is not unique.
- The axis Y_i, not shown in the figure, is chosen to form a right-handed orthonormal coordinate system with X_i and Z_i .

The elementary transformations that transition from frame $R_{i-1}$to frame R_i [Figure 2.1] are defined by the following four parameters:

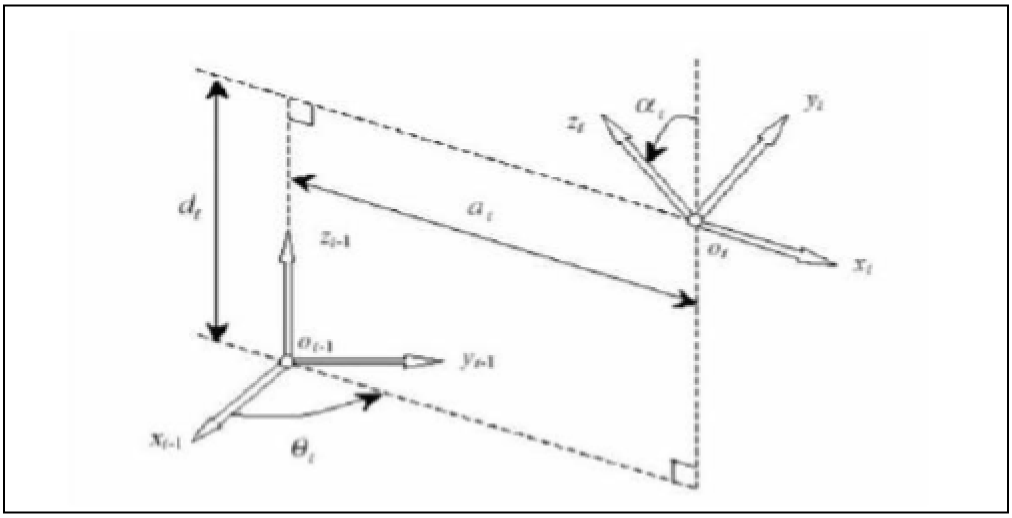

Figure 2.1: Representation of the D-H Parameters [8]

Here is (D-H) Parameters :

The Denavit-Hartenberg (D-H) parameters are used to represent the relative positions and orientations of adjacent links in a robotic manipulator. The parameters include:

. θ_i : Joint angle, the rotational displacement around Z_{i-1}.

. d_i : Link offset, the translational displacement along Z_{i-1}.

. r_i : Link length, the distance between Z_{i-1} and Z_i along X_i.

. α_i : link twist,the angle between Z_{i-1} and Z_i along X_i.

It should be noted that the angles are positive when the rotation is counterclockwise. The joint variable q_j associated with the j-th joint is either q_j or r_j,depending on whether the joint is revolute or prismatic, which is expressed by the relation:

$$q_j = \sigma_j \bullet \theta_j + \sigma_j \bullet r_j \tag{2.1}$$

With:

* $\sigma_j = 0$ if the j-th joint is revolute

* $\sigma_j = 1$ if the j-th joint is prismatic

* $\sigma_j = 1 - \sigma_j$

In terms of the homogeneous transformation matrix, the four elementary transformations defining the frame R_j in the frame R_{j-1}yield the following matrix:

$$T_{j-1,j} = Rot(X, \alpha_j) \bullet Trans(Z, d_j) \bullet Rot(Z, \theta_j) \bullet Trans(X, a_j) \tag{2.2}$$

After its development , we obtain:

$$T_{j-1,j} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_j) & -\sin(\alpha_j) & 0 \\ 0 & \sin(\alpha_j) & \cos(\alpha_j) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & d_j \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta_j) & -\sin(\theta_j) & 0 & 0 \\ \sin(\alpha_j) & \cos(\alpha_j) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r_j \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{j-1,j} = \begin{bmatrix} \cos(\theta_j) & -\sin(\theta_j) & 0 & d_j \\ \cos(\alpha_j)\sin(\theta_j) & \cos(\alpha_j)\cos(\theta_j) & -\sin(\alpha_j) & -r\sin(\alpha_j) \\ \cos(\alpha_j)\sin(\theta_j) & \sin(\alpha_j)\cos(\alpha_j) & \cos(\alpha_j) & r\cos(\alpha_j) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

the homogeneous transformation matrix $T_{j-1,j}$ is often written in the form:

$$T_{j-1,j} = \begin{bmatrix} A_{j-1,j} & p_{j-1,j} \\ 0 & 1 \end{bmatrix} \tag{2.3}$$

* ${}^{j-1}A_j$: The rotation matrix (3×3), also called the orientation matrix or the direction cosine matrix, represents the rotation between the two frames R_{j-1} and R_j, and can be obtained by: ${}^{j-1}A_j = Rot(X, \alpha_j) \times Rot(Z, \theta_j)$

* ${}^{j-1}P_j$:The position matrix (3×1) that defines the origin of frame R_j in frame R_{j-1}

These matrices combine to describe both the rotation and translation necessary to move from one coordinate frame to the next in the kinematic chain of a robotic manipulator.

2.4 Forward Geometric Model of a Manipulator Robot

It expresses the position and orientation of the reference frame R_E attached to the tool, relative to a fixed frame R_F (e.g.,the workshop frame),as a function of the motorized (and electronically controlled) joint variables $q_1, q_2, ...q_n$of the mechanism.

After defining the four parameters $\alpha_j, d_j, \theta_j, r_j$for all the frames of the robot, as well as the position of its base in space, the geometry of the arm can be completely specified at any given moment.

The FGM is obtained by the successive multiplication of the transformation matrices between frames. It is therefore expressed in the form of a matrix defined as follows:

$${}^{F}T_E = {}^{F}T_0 \, {}^{0}T_1(q_1) \, {}^{1}T_2(q_2) ... {}^{n-1}T_n(q_n) \, {}^{n}T_E$$

We can also represent it with this equation :

$$X = f(q)$$

X : est le vecteur des coordonnées opérationnelles, il peut être défini avec les éléments de la matrice ${}^{F}T_E$ tel que :

$$X = [P_x P_y P_z s_x s_y s_z n_x n_y n_z a_x a_y a_z]^T$$

q : being the vector of joint variables such that : $q = [q_1, q_2, ..., q_n]^r$

2.5 Geometric Modeling

2.5.1 Frames and Parameters

The Denavit-Hartenberg method is employed to describe the morphology of the robot. The geometric dimensions of the PUMA 560 with 6 degrees of freedom are provided in the [Appendix], while the Denavit-Hartenberg (D-H) parameters are listed in the table. The joint range distribution is also presented in the table.

Figure 2.2: A look at Unimate Puma 560 [14]

- Puma 560 D-H Table

Link Number	Alpha (α)	Link length(a)	Joint offset (d)	Theta(θ)
1	pi/2	0	0	q_1
2	0	0.4318	0	q_2
3	-pi/2	0.0203	0.1500	q_3
4	pi/2	0	0.4318	q_4
5	-pi/2	0	0	q_5
6	0	0	0	q_6

Table 2.1: DH Parameter Table (Denavit-Hartenberg)

.

- Movement limitations of manipulator arm puma 560

$-160 < q_1 < 160$

$-225 < q_2 < 45$

$-45 < q_3 < 255$

$110 < q_4 < 170$

$-100 < q_5 < 100$

$-266 < q_6 < 266$

2.5.2 Forward Geometric Model (FGM) of the Chosen Robot

The calculation of the forward geometric model (FGM) leads to identifying the transformation matrix ${}^0T_6$ between R_0 and R_6. To do this, we perform the following successive calculations:

${}^0T_6 ={}^0 T_1 \cdot{}^1 T_6$

${}^1T_6 ={}^1 T_2 \cdot{}^2 T_6$

${}^2T_6 ={}^2 T_3 \cdot{}^3 T_6$

${}^3T_6 ={}^3 T_4 \cdot{}^4 T_6$

${}^4T_6 ={}^4 T_5 \cdot{}^5 T_6$

2.5.3 Particularity of Anthropomorphic Robots

When the kinematics of the robot involve two successive axes j and $j+1$ that are parallel ($\alpha_{j+1} = 0$), the rotations add up, and a transformation matrix can be defined [15].

$^{j-1}T_{j+1} =^{j-1} T_j \cdot^j T_{j+1}$

With:

$q_{j,j+1} = q_j + q_{j+1}$

The calculation of the homogeneous transformation matrices for the PUMA 560 with 6 DOF for $r = 1; \ldots ; 6$:

$$T(0,1) = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -\sin(q_1) & -\cos(q_1) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

$$T(1,0) = \begin{bmatrix} \cos(q_1) & 0 & -\sin(q_1) & 0 \\ -\sin(q_1) & 0 & -\cos(q_1) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

$$T(1,2) = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & D_2 \\ \sin(q_2) & \cos(q_2) & 0 & 0 \\ 0 & 0 & 0 & r_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.6}$$

$$T(2,1) = \begin{bmatrix} \cos(q_2) & \sin(q_2) & 0 & -\cos(q_2)D2 \\ -\sin(q_2) & \cos(q_2) & 0 & -\sin(q_2)D2 \\ 0 & 0 & 1 & -r_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.7}$$

$$T(2,3) = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & D3 \\ 0 & 0 & -1 & -r_3 \\ \sin(q_3) & \cos(q_3) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8}$$

$$T(3,2) = \begin{bmatrix} \cos(q_3) & 0 & \sin(q_3) & -\cos(q_3)D3 \\ \sin(q_3) & 0 & \cos(q_3) & \sin(q_3)D3 \\ 0 & -1 & 0 & -r_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.9}$$

$$T(3,4) = \begin{bmatrix} \cos(q_4) & -\sin(q_4) & 0 & D_4 \\ 0 & 0 & 1 & 0 \\ -\sin(q_4) & -\cos(q_4) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.10}$$

$$T(4,3) = \begin{bmatrix} \cos(q_4) & 0 & \sin(q_4) & -\cos(q_4)D_4 \\ -\sin(q_4) & 0 & -\cos(q_4) & \sin(q_4)D_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.11}$$

$$T(4,5) = \begin{bmatrix} \cos(q_5) & -\sin(q_5) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin(q_5) & \cos(q_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.12}$$

$$T(5,4) = \begin{bmatrix} \cos(q_5) & 0 & \sin(q_5) & 0 \\ -\sin(q_5) & 0 & \cos(q_5) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.13}$$

$$T(5,6) = \begin{bmatrix} \cos(q_6) & -\sin(q_6) & 0 & 0 \\ \sin(q_6) & \cos(q_6) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.14}$$

$$T(6,5) = \begin{bmatrix} \cos(q_6) & \sin(q_6) & 0 & 0 \\ -\sin(q_6) & \cos(q_6) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.15}$$

Finally:

$$^{0}T_6 =^{0} T_1 \cdot^{1} T_3 \cdot^{3} T_4 \cdot^{4} T_5 \cdot^{5} T_6 = \begin{bmatrix} s_x & n_x & a_x & p_x \\ s_y & n_y & a_y & p_y \\ s_z & n_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.16}$$

With:

$$\begin{aligned}
sx = {} & sin(q_6) \times sin(q_5) \times sin(q_4) \times sin(q_1 + q_2) - \cos(q_3) \times \cos(q_4) \times \cos(q_1 + q_2) \\
& - \cos(q_5) \times sin(q_3) \times \cos(q_1 + q_2) - \cos(q_6) \times \cos(q_5) \times sin(q_4) \\
& \times sin(q_1 + q_2) - \cos(q_4) \times \cos(q_4) \times \cos(q_1 + q_2) + sin(q_3) \times sin(q_5) \\
& \times \cos(q_1 + q_2) \\
sy = {} & \cos(q_6) \times \cos(q_3) \times sin(q_5) + \cos(q_4) \times \cos(q_5) \times sin(q_3) + sin(q_6) \times \cos(q_3) \times \cos(q_5) \\
& - \cos(q_4) \times sin(q_3) \times sin(q_5) \\
sz = {} & sin(q_6) \times sin(q_5) \times sin(q_4) \times \cos(q_1 + q_2) + \cos(q_3) \times \cos(q_4) \times sin(q_1 + q_2) \\
& + \cos(q_5) \times sin(q_3) \times sin(q_1 + q_2) - \cos(q_6) \times \cos(q_5) \times sin(q_4) \\
& \times cos(q_1 + q_2) + \cos(q_3) \times \cos(q_4) \times sin(q_1 + q_2) - sin(q_3) \times sin(q_5) \times sin(q_1 + q_2) \\
nx = {} & \cos(q_6) \times sin(q_5) \times sin(q_4) \times sin(q_1 + q_2) \times \cos(q_3) \times \cos(q_4) \times \cos(q_1 + q_2) \\
& - \cos(q_5) \times sin(q_3) \times \cos(q_1 + q_2) + sin(q_6) \times \cos(q_5) \times sin(q_4) \\
& \times sin(q_1 + q_2) - \cos(q_3) \times \cos(q_4) \times \cos(q_1 + q_2) + sin(q_3) \times sin(q_5) \times \cos(q_1 + q_2) \\
ny = {} & \cos(q_6) \times \cos(q_3) \times \cos(q_5) - \cos(q_4) \times sin(q_3) \times sin(q_5) - sin(q_6) \times \cos(q_3) \\
& \times \cos(q_5) + \cos(q_4) \times \cos(q_5) \times sin(q_3) \\
nz = {} & \cos(q_6) \times sin(q_5) \times sin(q_4) \times \cos(q_1 + q_2) + \cos(q_3) \times \cos(q_4) \times sin(q_1 + q_2) \\
& + \cos(q_5) \times sin(q_3) \times sin(q_1 + q_2) + sin(q_6) \times \cos(q_5) \times sin(q_4) \times \cos(q_1 + q_2) \\
& + \cos(q_3) \times \cos(q_4) \times sin(q_1 + q_2) - sin(q_3) \times sin(q_5) \times sin(q_1 + q_2) \\
ax = {} & \cos(q_4) \times sin(q_1 + q_2) + \cos(q_3) \times sin(q_4) \times \cos(q_1) \times \cos(q_2) - sin(q_1) \times sin(q_2)
\end{aligned}$$

$ay = \sin(q_3) \times \sin(q_4)$

$az = \cos(q_4) \times \cos(q_1 + q_2) - \cos(q_3) \times \sin(q_4) \times \sin(q_1 + q_2)$

$px = d_3 \times \cos(q_1 + q_2) + r_3 \times \sin(q_1 + q_2) + d_2 \times \cos(q_1) + d_4 \times \cos(q_3) \times \cos(q_1 + q_2)$

$py = r_2 + d_4 \times \sin(q_3)$

$pz = r_3 \times \cos(q_1 + q_2) - d_3 \times \sin(q_1 + q_2) - d_2 \times \sin(q_1) - d_4 \times \cos(q_3) \times \sin(q_1 + q_2)$

2.5.4 Work space of a robot

The workspace of a manipulator arm refers to the set of all positions and orientations that the end-effector can reach. For a 6-DOF (Degrees of Freedom) manipulator arm, the workspace is determined by the limits of its joints and the physical dimensions of its links. Factors Influencing the Workspace

- Joints limits.
- Link lengths.
- Joint types.
- Base position.

Workspace of puma 560 for
$q_1 = [0, 360]$ degrees
$q_2 = [0, 360]$ degrees
$q_3 = 60$ degrees

2.6 Inverse Geometric Model (IGM)

2.6.1 Introduction

We have seen that the forward geometric model of a robot allows us to calculate the operational coordinates, giving the position of the end effector based on the joint coordinates. The inverse problem consists of calculating the joint coordinates corresponding to a given position of the end effector.

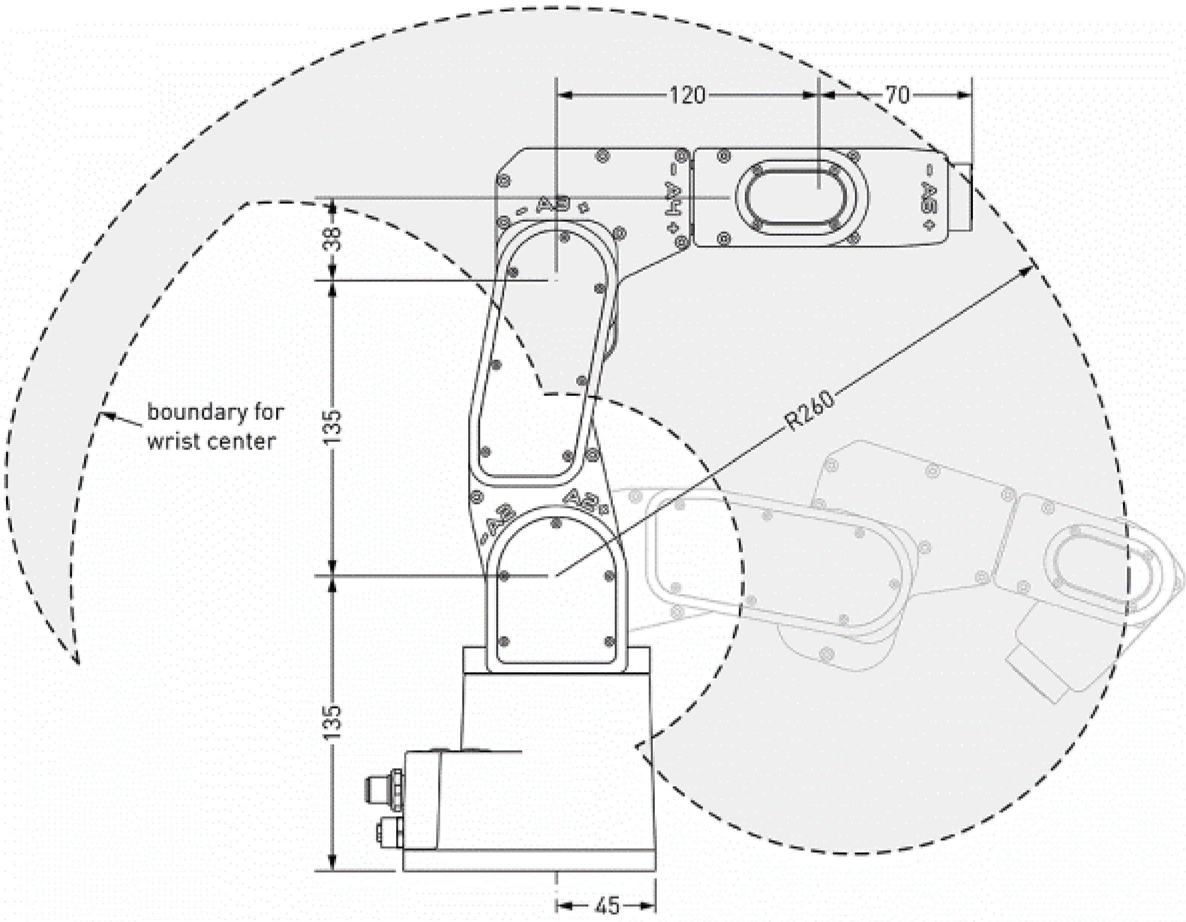

Figure 2.3: a sketch represent the workspace of 6-dof robot[16]

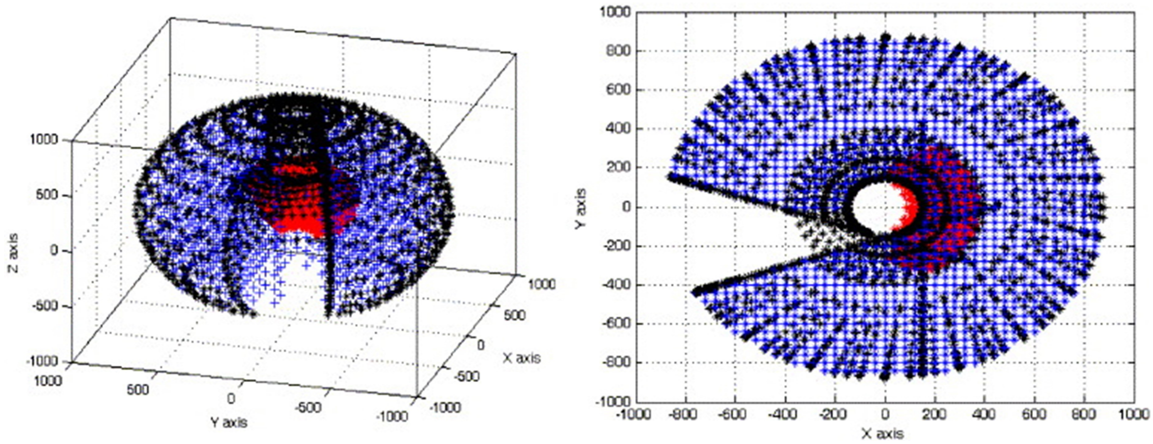

Figure 2.4: Workspace of puma 560[17]

When it exists, the explicit form that provides all possible solutions (there is rarely a unique solution) is called the inverse geometric model (IGM). There are three main methods for calculating the IGM:

- Paul's Method [18]: This method handles each specific case separately and is suitable for most industrial robots.
- Pieper's Method [19]: This method solves the problem for six-degree-of-freedom robots with three intersecting rotational joints or three prismatic joints.
- Roth's and Raghavan General Method: This method provides the general solution for six-joint robots based on a polynomial of degree at most 16.

We will focus on using Paul's method, as it is suitable for most industrial robots.

2.6.2 Calculation of the Inverse Geometric Model (IGM)

The position of the end effector of an n-degree-of-freedom manipulator robot is described by the forward geometric model, which can be expressed as:

$$^{n}T_{1}{=}^{0}T_{1}(q_{1})^{1}T_{2}(q_{2})....^{n-1}T_{n}(q_{4}) \tag{2.17}$$

This desired position can be represented by the homogeneous transformation matrix U_0 such that:

$$U_0 = \begin{bmatrix} S_x & N_x & A_x & P_x \\ S_y & N_y & A_y & P_y \\ S_z & N_z & A_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

We aim to solve the following system of equations:

$$U_{0}{=}^{0}T_{1}(q_{1})^{1}T_{2}(q_{2})....^{n-1}T_{n}(q_{n}) \tag{2.18}$$

To find the solutions to the proposed equation, Paul suggested a method that involves pre-multiplying both sides of equation [2.5] successively by the inverse of the matrices $^{j-1}T_j$ for j varying from 1 to n-1. This approach allows for isolating and identifying the joint variables one by one. For example, in a 6-DOF robot, the process is as follows: First, we multiply the left side of equation [2.5] by ${}^{1}T_{0}$:

$$ {}^{1}T_{0}U_{0}={}^{1}T_{2}{}^{2}T_{3}{}^{3}T_{4}{}^{4}T_{5}{}^{5}T_{6} \tag{2.19} $$

By identifying each term on both sides of equation [2.6], we reduce it to a system of equations that depends only on q_1 which we solve according to Table II-3. Then, we pre-multiply the left side of expression [2.6] by ${}^{2}T_{1}$ and calculate q_2.

The sequence of equations allowing the calculation of all is as follows:

$U_0 =^0 T_1^1 T_2^2 T_3^3 T_4^4 T_5^5 T_6$

${}^{1}T_0 U_0 =^1 T_2^2 T_0^3 T_4^4 T_5^5 T_6$

${}^{2}T_1 U_1 =^2 T_3^3 T_4^4 T_5^5 T_6$

${}^{3}T_2 U_2 =^3 T_4^4 T_5^5 T_6$

${}^{4}T_3 U_3 =^4 T_5^5 T_6$

${}^{5}T_4 U_4 =^5 T_6$

2.6.3 The Inverse Geometric Model (IGM) of the Chosen Robot

The forward geometric model (FGM) of the robot has already been established in the previous paragraphs. The geometric parameters are given in Table II-3. Since the robot has a spherical wrist, we used Paul's method with kinematic decoupling.

The desired position of the tool with respect to the reference frame R_0 is given by the matrix U_0:

$$ U_0 = \begin{bmatrix} S_x & N_x & A_x & P_x \\ S_y & N_y & A_y & P_y \\ S_z & N_z & A_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.20} $$

The system of equations that we need to solve is:

$U_0 =^0 T_6(q)$

2.6.4 Position Equation

Calculation of q3

Since ${}^{0}p_6 = {}^{0}p_4$, we can write that the fourth column of the product of the transformations ${}^{0}T_1^1T_2^2T_3^3T_4$ to the fourth column of U_0:

$$ {}^{0}T_4 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = {}^{0}T_1^1T_2^2T_3^3T_4 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{2.21} $$

$$ \begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix} = \begin{bmatrix} (-\sin(q_1)\sin(q_2) + \cos(q_1)\cos(q_2)\cos(q_3)D_4 - (-\cos(q_1)\sin(q_2) - \sin(q_1)\cos(q_2))r_3 + \cos(q_1)D_2 \\ \sin(q_)D4 + r_2 \\ (-\cos(q_1)\sin(q_2) - \sin(q_1)\cos(q_2)\cos(q_3)D_4 - (sin(q_1)sin(q_2) - cos(q_1)cos(q2))r_3 - sin(q_1)D_2 \\ 1 \end{bmatrix} \tag{2.22} $$

$p_x = -\sin(q_1)\sin(q_2) + \cos(q_1)\cos(q_2)\cos(q_3)D_4 - (-\cos(q_1)\sin(q_2) - \sin(q_1)\cos(q_2))r_3 + \cos(q_1)D_2$

$p_y = \sin(q_)D4 + r_2$

$p_z = -\cos(q_1)\sin(q_2) - \sin(q_1)\cos(q_2)\cos(q_3)D_4 - (sin(q_1)sin(q_2) - cos(q_1)cos(q2))r_3 - sin(q_1)D_2$

We pre-multiply both sides by ${}^{1}T_0$and we identify, term by term, the elements on both sides. We will have:

- Identification between the corresponding elements $U_2, {}^{2}T_4$:

$$ \begin{aligned} U_2[1,4] &= {}^{1}T_4[1,4], \\ U_2[2,4] &= {}^{2}T_4[2,4], \\ U_2[3,4] &= {}^{3}T_4[3,4], \end{aligned} $$

$$ \begin{cases} cos(q_1)cos(q2)p_x - sin(q_q1)p_z + sin(q_2)(-sin(q_1)p_x - cos(q_1)p_z) - cos(q_2)D_2 = cos(q_3)D_4 + D_3 \\ -sin(q_2)cos(q_1)p_x - sin(q_1)p_2 + sin(q_2)(-sin(q_1)p_x - cos(q_1)p_2) + sin(q_2)D2 = -r_3 \\ p_y - r_2 = sin(q_3)D_4 \end{cases} \tag{2.23} $$

So

$=> sin(q_3) = \frac{(p_y - r_2)}{D_4}$

Let's say that for example: C_3=cos(q_3) and s2=sin(q_2) and so on:

$q_3 = arctan(\frac{S_3}{C_3})$

We can calculate q_2 by considering the first two equations. Initially, by squaring each equation and then summing them up :

$X.sin(q_2) + Y.cos(q_2) = Z$ with

$X = 2 * r_3 * D_3;$

$Y = D_2 * (C_3 * D4 + D_3);$

$Z = Px^2 + Pz^2 - C3^2 * D4^2 - 2 * C3 * D4 * D3 - r3^2 - D2^2;$

$sin(q_2) = \frac{XZ - \varepsilon X\sqrt{X^2 + Y^2 + Z^2}}{X^2 + Y^2}$

$cos(q_2) = \frac{XZ - \varepsilon X\sqrt{X^2 + Y^2 + Z^2}}{X^2 + Y^2}$

q_2 **result is:**

$q_2 = atan2(\frac{S_2}{C_2});$

$$\begin{cases} cos(q_1)cos(q2)p_x - sin(q_q1)p_z + sin(q_2)(-sin(q_1)p_x - cos(q_1)p_z) - cos(q_2)D_2 = cos(q_3)D_4 + D_3 \\ -sin(q_2)cos(q_1)p_x - sin(q_1)p_2 + sin(q_2)(-sin(q_1)p_x - cos(q_1)p_2) + sin(q_2)D2 = -r_3 \\ p_y - r_2 = sin(q_3)D_4 \end{cases} \tag{2.24}$$

From equations 1 and 2 we got:

$X_1sin(q_2) + Y_1cos(q_2) = Z_1$

$X_2sin(q_2) + Y_2cos(q_2) = Z_2$

$X_1 = -xos(q_2)P_z - sin(q_2)P_x[a]$

$Y_1 = -xos(q_2)P_x - sin(q_2)P_z[b]$

$Z_1 = -xos(q_3)D_4 + cos(q_2)D_2 + D_3[C]$

$Z_2 = -r_3 - sin(q_2)D_2[D]$

The condition $(X_1Y_2)-(X_2Y_1) \neq 0$ means that the equations 1 and 2 are independent.

$$sin(q_1) = \frac{X_1Z_1 + \varepsilon X_1\sqrt{X_1^2+Y_1^2-Z_1^2}}{X_1^2+Y_1^2}$$

$$cos(q_1) = \frac{X_1Z_1 + \varepsilon X_1\sqrt{X_1^2+Y_1^2-Z_1^2}}{X_1^2+Y_1^2}$$

q1 solution is :

$$\text{q1=atan2}\frac{s_1}{c_1} \tag{2.25}$$

Calculating q_4:

To get q_4 solution, first we need the calculation of the wrist:

$$UU_0 = \begin{bmatrix} F_x & G_x & H_x \\ F_y & G_y & H_y \\ F_z & G_z & H_z \end{bmatrix} \tag{2.26}$$

F, G and H equations are:

$$F[x] = UU_0[1,1]$$
$$F[y] = UU_0[2,1]$$
$$F[z] = UU_0[3,1]$$

$$Fx = (cos(q1)*cos(q2)*cos(q3) - sin(q1)*sin(q2)*sin(q3))*sx + [\,sin(q3)\,]*sy + (-sin(q1)*cos(q2)*cos(q3) - cos(q1)*sin(q2)*cos(q3))*sz;$$

$$Fy = (-cos(q1)*cos(q2)*cos(q3) + sin(q1)*sin(q2)*sin(q3))*sx + [\,cos(q3)\,]*sy + (sin(q1)*cos(q2)*sin(q3) + cos(q1)*sin(q2)*sin(q3))*sz;$$

$$Fz = (cos(q1)*sin(q2) + sin(q1)*cos(q2))*sx + (-sin(q1)*sin(q2) + cos(q1)*cos(q2))*sz;$$

$$G[x] = UU_0[1,2]$$
$$G[y] = UU_0[2,2]$$
$$G[z] = UU_0[3,2]$$

$$Hx = (cos(q1)*cos(q2)*cos(q3) - sin(q1)*sin(q2)*sin(q3))*ax + [\,sin(q3)\,]*ay + (-sin(q1)*cos(q2)*cos(q3) - cos(q1)*sin(q2)*cos(q3))*az;$$

$$Hz = (cos(q1)*sin(q2) + sin(q1)*cos(q2))*ax + (-sin(q1)*sin(q2) + cos(q1)*cos(q2))*az;$$

[cos (q_4) Hx-sin (q_4)] Hz=0 ➔ from this equation we get ➔

$$q_4 = \arctan\left(\frac{H_x}{H_z}\right), \tag{2.27}$$

Calculation of q_5:

$$UU_2[1,1] = A[5,6][1,1]$$
$$UU_2[1,2] = A[5,6][1,2]$$
$$UU_2[1,3] = A[5,6][1,3]$$

$$\cos(q_5)[\cos(q_4)F_x - \sin(q_4)F_z] + \sin(q_5)F_y = \cos(q_6)$$
$$\cos(q_5)[\cos(q_4)G_x - \sin(q_4)G_z] + \sin(q_5)G_y = -\sin(q_6)$$
$$\cos(q_5)[\cos(q_4)H_x - \sin(q_4)H_z] + \sin(q_5)H_y = 0$$

We found q5 as a result from equation 3:

$$\sin(q_5) = \frac{XZ - \varepsilon X\sqrt{X^2 + y^2 - Z^2}}{X^2 + y^2}$$
$$\cos(q_5) = \frac{XZ + \varepsilon X\sqrt{X^2 + y^2 - Z^2}}{X^2 + y^2}$$

$$q_5 = \text{arctan2}\left(\frac{\sin(q_5)}{\cos(q_5)}\right) \quad (2.28)$$

Calculation of q_6 :

From the previous equations 1 and 2 we get:

$$X_1\sin(q_3) + Y_1\cos(q_3) = Z_1$$
$$X_2\sin(q_3) + Y_2\cos(q_3) = Z_2$$
$$X_1Y_1 - X_2Y_2 = 0$$

From which we immediately derive that:

$$q_6 = \textbf{arctan}\left(\frac{Z_1}{X_1}, \frac{Z_2}{Y_2}\right) \quad (2.29)$$

With:

$X_1 = -1;$

$Y_2 = 1;$

$Z_1 = \cos(q_5)[\cos(q_4)G_x - \sin(q_4)G_z] + \sin(q_5)G_y$

$Z_2 = \cos(q_5)[\cos(q_4)F_x - \sin(q_4)F_z] + \sin(q_5)F_y$

Using Paul's method, we successfully solved the inverse geometric model for the proposed robot. Following this geometric modeling, both direct and inverse, we will proceed to kinematic analysis in the next chapter. This will enable us to calculate Cartesian and joint velocities.

2.7 Conclusion

In this chapter, we presented a method for calculating the forward geometric model of robots with a simple open structure using Denavit-Hartenberg parameters. We selected the PUMA 560 manipulator robot to apply these geometric models, as well as other models that will be discussed in subsequent chapters.

We demonstrated that operational coordinate representation for rotation can be achieved using direction cosines or Euler angles. This section concluded with the calculation of the robot's workspace, excluding its singular positions.

As we have covered the forward geometric model (FGM) of a manipulator, the inverse problem, which involves determining the joint variables for a given desired configuration of the end effector, is referred to as the inverse geometric model (IGM).

We also introduced Paul's method for calculating the IGM. This intuitive method allows the user to select the equations to solve and is applicable to a wide range of kinematic chains, particularly those with specific geometric parameters such as zero distances or angles with sines and cosines equal to 0, 1, or -1. Additionally, this analytical method provides all possible solutions for the inverse geometric model.

Chapter 3

Kinematics Of Manipulator

3.1 Introduction

Kinematics is the branch of science that examines the movement of manipulator links without regard to the forces that cause it. In that case the motion is determined with trajectory, i.e. position, velocity, acceleration and additional higher derivative terms.
Dynamics deals the relation between the applied forces/torques and the resulting motion of an industrial manipulator

3.1.1 Direct Kinematics (DKM)

The Direct Kinematic Model (DKM) allows calculating the components of the kinematic screw from the generalized joint velocities, which are the time derivatives of the generalized coordinates. The kinematic screw is defined by the velocity and rotation components of the different parts of the robot.

Jacobian Matrix

Jacobian Matrix The Jacobian matrix, J, is a matrix that establishes a relationship between the joint velocities $\dot{\theta}$ (the time derivatives of the generalized coordinates θ) and the linear and angular velocities $\dot{X}$ of the robot's end effector. The relationship can be expressed as follows:

$\dot{X}$=$\mathbf{J}(\theta)\times\ \dot{\theta}$

- $\dot{X}$ represents the vector of linear and angular velocities of the end effector.

- $\dot{\theta}$ is the vector of joint velocities (generalized velocities).
- J(θ) is the Jacobian matrix that depends on the generalized coordinates θ.

Calculation of the Kinematic Screw

The kinematic screw T can be defined as a vector combining the linear and angular velocities. For a manipulator robot, the kinematic screw at the end effector can be written as:

$$T = \begin{bmatrix} V \\ \omega \end{bmatrix} \tag{3.1}$$

- V is the vector of linear velocities.
- ω is the vector of angular velocities.

The relationship between the joint velocities and the kinematic screw is thus established via the Jacobian matrix. Therefore, for a given set of joint velocities, the Jacobian matrix allows determining the kinematic screw of the robot, providing an overview of the mobility of the end effector in terms of linear and angular velocities.

Importance of the Jacobian Matrix

- **Mobility Analysis:** It helps understand how joint movements affect the end effector's movements.
- **Robot Control:** Used in control algorithms to define required trajectories and velocities.
- **Motion Planning:** Helps plan optimal trajectories considering speed and acceleration constraints.
- **Stability and Performance:** Contributes to stability analysis and performance optimization of the robot.

Calculation of Indirect Kinematics Model (IKM)

The indirect calculation of the Jacobian matrix involves using the geometric model of the manipulator robot.

$$X = f(q) \tag{3.2}$$

By definition, the Jacobian matrix is the matrix of partial derivatives of the function fff with respect to the generalized coordinates, thus:

$$J(q) = \frac{\partial X}{\partial q} \tag{3.3}$$

This differentiation method is easy to implement for robots with two or three degrees of freedom in the plane, but for robots with more than three degrees of freedom, manual differentiation becomes difficult.

Steps for Indirect Calculation

1. Geometric Model:

- Define the geometric relationship between the joint coordinates q and the end effector's position and orientation X. This is usually represented as a set of equations X=f(q).

2. Differentiation:

- Differentiate the geometric model X=f(q) with respect to the generalized coordinates q to obtain the Jacobian matrix.

3. Jacobian Matrix Expression:

- The Jacobian matrix J(q) is expressed as: J(q)= $\frac{\partial X}{a q}$

For a robot with six degrees of freedom, the Jacobian matrix J(q)is a 6x6 matrix:

$$J(q)\begin{bmatrix} \frac{\partial X}{\partial q1} & \frac{\partial X}{\partial q2} & . & . & . & \frac{\partial X}{\partial q6} \\ \frac{\partial y}{\partial q1} & \frac{\partial y}{\partial q2} & . & . & . & \frac{\partial y}{\partial q6} \\ \frac{\partial z}{\partial q1} & \frac{\partial z}{\partial q2} & . & . & . & \frac{\partial z}{\partial q6} \\ \frac{\partial \alpha}{\partial q1} & \frac{\partial \alpha}{\partial q2} & . & . & . & \frac{\partial \alpha}{\partial q6} \\ \frac{\partial \beta}{\partial q1} & \frac{\partial \beta}{\partial q2} & . & . & . & \frac{\partial \beta}{\partial q6} \\ \frac{\partial \gamma}{\partial q1} & \frac{\partial \gamma}{\partial q2} & . & . & . & \frac{\partial \gamma}{\partial q6} \end{bmatrix} \tag{3.4}$$

- **x , y , z** represent the position coordinates of the end effector.
- α, β, γ represent the orientation coordinates (e.g., Euler angles or roll, pitch, yaw).
- **q1 , q2 , . . . , q6** represent the generalized coordinates (joint variables).

Practical Considerations

While this method is straightforward for robots with a small number of degrees of freedom, the complexity increases significantly for robots with more than three degrees of freedom. Manual differentiation becomes impractical, and computational tools or symbolic computation software are often used to handle the differentiation and obtain the Jacobian matrix for complex robotic systems.

In conclusion, the indirect calculation of the Jacobian matrix through geometric modeling and partial differentiation is a foundational technique in robotics, enabling the analysis and control of robotic manipulators in various applications.

Direct Calculation of the Jacobian Matrix

A very common method for kinematic calculations allows obtaining the Jacobian matrix through a direct calculation based on the influence each joint k in the chain has on the terminal frame Rn.

The velocities $v_{K,N}$ and angular velocities $\omega_{K,N}$ can be calculated separately for the cases of a prismatic joint and a revolute joint:

For a Prismatic Joint

• **Prismatic Joint:** A prismatic joint translates along a fixed axis, contributing to the linear velocity of the end effector without affecting its angular velocity.

$$V_{K,N} = z_{K-1} \tag{3.5}$$

Here, z_{K-1} is the unit vector along the axis of the prismatic joint K in the previous frame K-1.

$\omega_{K-n} = 0$

Since a prismatic joint does not contribute to the angular velocity, $\omega_{K,n}$ is zero.

For a Revolute Joint

Revolute Joint: A revolute joint rotates about a fixed axis, contributing to both the linear and angular velocities of the end effector.

$$V_{K,n} = \omega_{K,n} x(p_n - p_K) \tag{3.6}$$

Here:

- $\omega_{K,n} = \dot{\theta}_K Z_{K-1}$ is the angular velocity vector of the revolute joint k.
- $P_n$is the position vector of the end effector.
- P_K is the position vector of the joint k.

The cross product $\omega_{K,n} \times (P_n - P_K)$ gives the linear velocity contribution of the revolute joint k to the end effector.

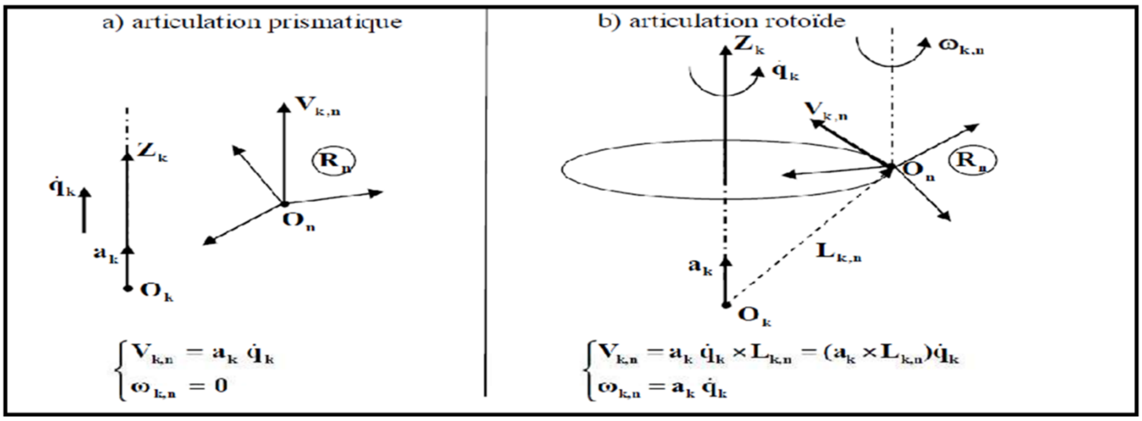

Figure 3.1: Influence du type de l'articulation sur le repère terminal

General Jacobian Matrix Construction

By assembling the contributions of all joints in the manipulator, we can construct the full Jacobian matrix JJJ.

1. **Position Part:** Consists of the linear velocity contributions from each joint.

$$J_v = [V_{1,n}, V_{2,n}, ..., V_{K,n}] J = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} \tag{3.7}$$

2. **Orientation Part:** Consists of the angular velocity contributions from each joint.

$$J_\omega = [\omega_{1,n}, \omega_{2,n}, ..., \omega_{K,n}] \tag{3.8}$$

3. **Full Jacobian Matrix:** Combines both the position and orientation parts.

$$T = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} \tag{3.9}$$

For a manipulator with n degrees of freedom, the Jacobian matrix J is a 6×n matrix, reflecting the contributions of all joints to the end effector's linear and angular velocities.

By using this direct method, the Jacobian matrix can be efficiently computed by considering the specific characteristics of prismatic and revolute joints, providing a comprehensive understanding of how each joint affects the overall motion of the robot.

Calculation of the DKM by Recurrence Equations

Knowing J_n , the translational and rotational velocities of the frame can be obtained from the relation (III-2). However, from the perspective of the number of operations, it is more efficient to use the following recurrence equations:

$$\begin{cases} {}^{j}V_j = {}^{j}A_{j-1}({}^{j-1}V_{j-1} + {}^{j-1}\widehat{\omega}_{j-1}{}^{j-1}p_j) + \sigma \dot{q}_j {}^{j}a_j \\ {}^{j}\omega_j = {}^{j}A_{j-1}{}^{j-1}\omega_{j-1} + \bar{\sigma}\dot{q}_j {}^{j}a_j \end{cases} j = 1.....n \tag{3.10}$$

We initialize the recurrence with the operational velocitiesV_0 and ω_0 of the robot's base. The velocity obtained in this case is ${}^{n}X_n$ To find ${}^{0}X_n$ we project this vector into the frame R_0:

$$ {}^{0}V_n = {}^{0}A_n {}^{n}V_n and {}^{0}\omega_n {}^{n}\omega_n \tag{3.11}$$

The calculation of the DKM, either by the Jacobian matrix or by recurrence equations, gives the vector $\begin{bmatrix} V_n \\ \omega_n \end{bmatrix}$ where V_n is the time derivative of the position vector p_n. However, ω_n does not represent the derivative of the orientation. Therefore, it is necessary to find a relationship between the operational coordinates X and the kinematic model.

Utilization of the Jacobian Matrix

The Jacobian matrix J_n is one of the most important quantities in the analysis and control of robot motion. It appears in virtually every aspect of robotic manipulation: in planning, in determining singular configurations, in deriving the dynamic equations of motion, and in transforming forces and torques from the end effector to the manipulator joints.

3.2 Application on the PUMA 560 Robot

3.2.1 Calculation of the Jacobian by Recurrence Equations

The results obtained by the DKM, we have calculated the Jacobian in different frames: ${}^{6}J_6$, ${}^{0}J_6$.

Calculate the Jacobian ${}^{0}J_6$ for the PUMA 560 robot. The elements of the K^{th} column of

$$
{}^{0}J_6 = \begin{bmatrix} J[1.1] & J[1.2] & J[1.3] & J[1.4] & J[1.5] & J[1.6] \\ J[2.1] & J[2.2] & J[2.3] & J[2.4] & J[2.5] & J[2.6] \\ J[3.1] & J[3.2] & J[3.3] & J[3.4] & J[3.5] & J[3.6] \\ J[4.1] & J[4.2] & J[4.3] & J[4.4] & J[4.5] & J[4.6] \\ J[5.1] & J[5.2] & J[5.3] & J[5.4] & J[5.5] & J[5.6] \\ J[6.1] & J[6.2] & J[6.3] & J[6.4] & J[6.5] & J[6.6] \end{bmatrix} \tag{3.12}
$$

$J[1.1] = -cos(q_1)(d_3sin(q_2) - r_3cos(q_2) + d_4cos(q_3)sin(q_2)) - sin(q_1)(d_2) + d_3cos(q_2) + r_3sin(q_2) + d_4cos(q_2)cos(q_3)$

$J[1.2] = r_3\left[cos(q_1+q_2)\right] - \left[d_3 + d_4cos(q_3)\right]sin(q_1+q_2)$

$J[1.3] = -d_4sin(q_3)\left[cos(q_1)cos(q_2) - sin(q_1)sin(q_2)\right]$

$J[1.4] = 0$

$J[1.5] = 0$

$J[1.6] = 0$

$J[2.1] = 0$

$J[2.2] = 0$

$J[2.3] = d_4cos(q_3)$

$J[2.4] = 0$

$J[2.5] = 0$

$J[2.6] = 0$

$J[3.1] = sin(q_1)\left[d_3sin(q_2) - r_3cos(q_2) + d_4cos(q_3)sin(q_2)\right]$

$-cos(q_1)\left[d_2 + d_3cos(q_2) + r_3sin(q_2) + d_4cos(q_2)cos(q_3)\right]$

$J[3.2] = -\left[d_3 + d_4cos(q_3)\right]\left[cos(q_1+q_2)\right] - r_3sin(q_1+q_2)$

$J[3.3] = d_4sin(q_3)\left[cos(q_1)sin(q_2) + cos(q_2)sin(q_1)\right]$

$J[3.4] = 0$

$J[3.5] = 0$

$J[3.6] = 0$

$J[4.1] = 0$

$J[4.2] = 0$

$J[4.3] = sin(q_1+q_2)$

$J[4.4] = -sin(q_3)\left[cos(q_1+q_2)\right]$

$J[4.5] = cos(q_4)sin(q_1+q_2) + cos(q_3)sin(q_4)\left[cos(q_1+q_2)\right]$

$J[4.6] = cos(q_4)\left[sin(q_1+q_2) + cos(q_3)sin(q_4)cos(q_1+q_2)\right]$

$J\,[5.1] = 1$

$J\,[5.2] = 1$

$J\,[5.3] = 0$

$J\,[5.4] = cos(q_3)$

$J\,[5.5] = sin(q_3)sin(q_4)$

$J\,[5.6] = sin(q_3)sin(q_4)$

$J\,[6.1] = 0$

$J\,[6.2] = 0$

$J\,[6.3] = cos(q_1 + q_2)$

$J\,[6.4] = sin(q_3)\,[cos(q_1)sin(q_2) + cos(q_2)sin(q_1)]$

$J\,[6.5] = cos(q_4)sin(q_1 + q_2) - cos(q_3)sin(q_4)sin(q_1 + q_2)$

$J\,[6.6] = cos(q_4)(cos(q_1 + q_2)) - cos(q_3)sin(q_4)sin(q_1 + q_2)$

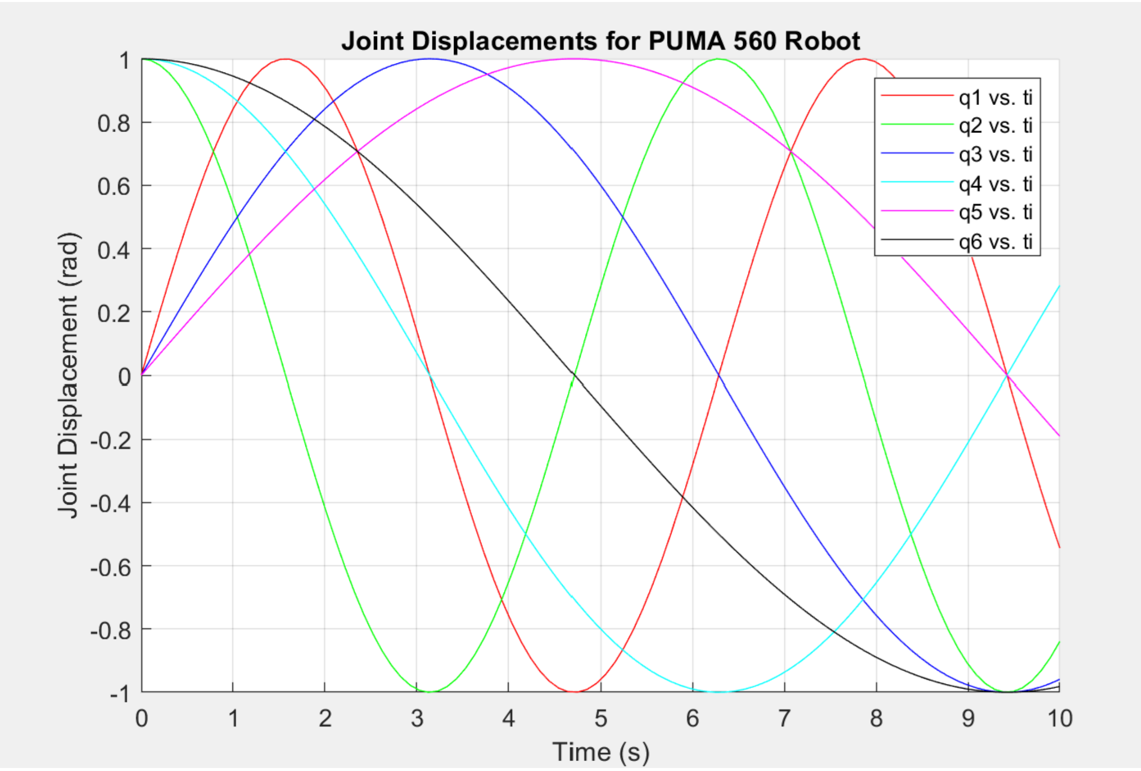

Figure 3.2: joint Diplacements for puma 560

3.2.2 Inverse Kinematic Model in the Regular Case

In this case, the Jacobian matrix J is a square matrix of order n and its determinant is non-zero. The most general method involves calculating J^{-1},the inverse of J, which allows determining the joint velocities $\dot{q}$using the relation:

$$\dot{q} = J^{-1} \times \dot{X} \tag{3.13}$$

When the Jacobian matrix JJJ has the following form:
$J = \begin{bmatrix} A & 0 \\ B & C \end{bmatrix}$ The matrices A and C being square and invertible, it is easy to show that

the inverse of this matrix is written:
$J^{-1} = \begin{bmatrix} A^{-1} & 0 \\ -C^{-1}BA^{-1} & B^{-1} \end{bmatrix}$

The resolution of the problem therefore reduces to the much simpler inversion of two smaller matrices.
When the manipulator robot has six degrees of freedom and a spherical wrist, the general matrix J is as shown in relation [III-14], with A and C being 3×3 matrices.

3.2.3 Calculation of the Inverse Jacobian Matrix

$$J^{-1} = \begin{bmatrix} 11 & 12 & 13 & 14 & 15 & 16 \\ 21 & 22 & 23 & 24 & 25 & 26 \\ 31 & 32 & 33 & 34 & 35 & 36 \\ 41 & 42 & 43 & 44 & 45 & 46 \\ 51 & 52 & 53 & 54 & 55 & 56 \\ 61 & 62 & 63 & 64 & 65 & 66 \end{bmatrix} \tag{3.14}$$

$J^{-1}[11] = 0$
$J^{-1}[12] = 0$
$J^{-1}[13] = -\frac{1}{-r_2 cosq_2 sinq_3 - r_2 sinq_2 cosq_3 + cosq_2 D_3}$
$J^{-1}[14] = 0$
$J^{-1}[15] = 0$
$J^{-1}[16] = 0$
$J^{-1}[21] = 0$
$J^{-1}[22] = \frac{1}{D_3 cosq_2}$
$J^{-1}[23] = 0$
$J^{-1}[24] = 0$
$J^{-1}[25] = 0$

$J^{-1}[26] = 0$
$J^{-1}[31] = \frac{-1}{r_2}$
$J^{-1}[32] = \frac{sinq_3 D_3 - r_2}{cosq_3 D_3 r_2}$
$J^{-1}[33] = 0$
$J^{-1}[34] = 0$
$J^{-1}[35] = 0$
$J^{-1}[36] = 0$

$$J^{-1}[41] = \frac{-sinq_4 cosq_5}{(sinq_4^2 + cosq_4^2) sinq_5 r_2}$$

$$J^{-1}[42] = \frac{sinq_3 sinq_4 sinq_5}{(sinq_4^2 + cosq_4^2) sinq_5 r_2 cosq_3}$$

$$J^{-1}[43] = \frac{-\left(-\cos q_3 \cos q_5 \cos q_4 \sin q_2 - \cos q_5 \cos q_4 \cos q_2 \sin q_3 - \cos q_2 \cos q_3 \sin q_4^2 \sin q_5 - \cos q_2 \cos q_3 \cos q_4^2 \sin q_5 + \sin q_2 \sin q_3 \cos q_4^2 \sin q_5\right)}{(-r_2 \cos q_2 \sin q_3 - r_2 \sin q_2 \cos q_3 + \cos q_2 D_3) \sin q_5 \left(\sin q_4^2 + \cos q_4^2\right)}$$

$$J^{-1}[44] = \frac{cosq_5 cosq_4}{sinq_5(sinq_4^2 + cosq_4^2)}$$

$$J^{-1}[45] = 1$$

$$J^{-1}[46] = \frac{-sinq_4 cosq_5}{sinq_5(sinq_4^2 + cosq_4^2)}$$

$$J^{-1}[51] = \frac{cosq_4}{(sinq_4^2 + cosq_4^2) r^2}$$

$$J^{-1}[52] = \frac{-sinq_3 cosq_4}{(sinq_4^2 + cosq_4^2) r^2 cosq_3}$$

$$J^{-1}[53] = \frac{(sinq_2 cosq_3 + cosq_2 sinq_3) sinq_4}{((-r_2 cosq_2 sinq_3 - r_2 sinq_2 cosq_3 + cosq_2 D_3)(sinq_4^2 + cosq_4^2))}$$

$$J^{-1}[54] = \frac{sinq_4}{sinq_4^2 + cosq_4^2}$$

$$J^{-1}[55] = 0$$

$$J^{-1}[56] = \frac{cosq_4}{{sinq_4}^2 + {cosq_4}^2}$$

$$J^{-1}[61] = \frac{sinq_4}{({sinq_4}^2 + {cosq_4}^2) sinq_5 r_2 cosq_3}$$

$$J^{-1}[62] = \frac{-sinq_3 sinq_4}{({sinq_4}^2 + {cosq_4}^2) sinq_5 r_2 cosq_3}$$

$$J^{-1}[63] = \frac{-((sinq_2 cosq_3 + cosq_2 sinq3) cosq_4)}{((-r cosq_2 sinq_3 - r_2 sinq_2 cosq_3 + cosq_2 D_3) sinq_5 ({sinq_4}^2 + {cosq_4}^2))}$$

$$J^{-1}[64] = \frac{-cosq_4}{sinq_5({sinq_4}^2 + {cosq_4}^2)}$$

$$J^{-1}[65] = 0$$

$$J^{-1}[66] = \frac{sinq_4}{sinq_5({sinq_4}^2 + {cosq_4}^2)}$$

3.3 Conclusion

In this chapter, we explored how to derive the direct kinematic model of a robot by calculating the geometric Jacobian matrix and determining its first-order inverse kinematic model. We then applied these principles to the PUMA 560 robot.

It is clear that the Jacobian matrix is an essential tool in kinematics and in identifying singular configurations.

We observed that for the same joint, the values obtained from direct modeling of the robot and from calculating the inverse model are not identical but are very close. This minor discrepancy can be attributed to factors such as repeatability and precision.

In the next chapter, we will shift our focus to dynamic modeling, where we will analyze the torques and forces acting on the system.

Chapter 4

Dynamic Model

4.1 Introduction

The dynamic model defines the relationship between the torques (and/or forces) applied to the actuators and the resulting positions, velocities, and accelerations of the robot.

Applications of the dynamic model include:

- **Simulation:** Utilizing the forward dynamic model to predict the robot's behavior.
- **Actuator Sizing:** Determining the appropriate specifications for the robot's actuators.
- **Parameter Identification:** Estimating the robot's inertial and friction parameters.
- **Control:** Implementing the inverse dynamic model for precise control of the robot's movements.

Various formalisms have been developed to derive the dynamic model of robots [19], [20], [21]. The most commonly used formalisms include:

- Lagrange Formalism
- Newton-Euler Formalism

4.2 Overview of Dynamic Modeling Formalisms

4.2.1 The Lagrange Formalism

The Lagrange method may not yield the most efficient model in terms of computational operations, but it is the simplest method considering its objectives. We will consider an ideal robot without friction, without elasticity, and not subjected to or exerting any external force.

The Lagrange formalism describes the equations of motion, assuming the external force on the end effector is zero, by the following equation:

$$\Gamma_i = \frac{d}{dt}\frac{\partial L}{\partial(\dot{q}_1)} = \frac{\partial L}{\partial(\dot{q}_1)} \quad (4.1)$$

With i=1......n

That means:

$$\Gamma_i = \frac{d}{dt}\frac{\partial E}{\partial \dot{q}_1} - \frac{\partial E}{\partial \dot{q}_1} + \frac{\partial U}{\partial q_1} + \Gamma e_i \quad 4(.2)$$

Where:

- L: Lagrangian of the system, equal to E-U.
- E: Total kinetic energy of the robot.
- U: Total potential energy of the robot.
- τ_i: Torque applied to joint iii, considered as the input torque.
- $\tau_e i$: Result of the forces exerted by the end effector on its environment.

To apply this formalism, we start by gathering the geometric and mechanical data available for the robot arm: dimensions, masses, inertias, friction, etc.

We can then establish the expression for the kinetic energy, which depends on the configuration and joint velocities. Using the same data, we calculate the potential energy, representing the effect of gravity, which also depends on the configuration.

4.2.2 Calculation of Potential Energy

The gravitational potential energy of an object is equal to the work done to overcome the gravitational force during a change in height [22]. For a manipulator robot, the potential energy U is given by:

$$U = \sum_{i=1}^{n} m_i g h_i \tag{4.3}$$

Where:

- m_i is the mass of link i.
- g is the acceleration due to gravity.
- h_i is the height of the center of mass of link i.

4.2.3 Calculation of Kinetic Energy

For a manipulator robot, the kinetic energy E can be written as [16]:

$$E = \frac{1}{2}\sum_{i=1}^{n} (m_i \parallel v_i \parallel^2 + {\omega_i}^T I_i \omega_i) \tag{4.4}$$

Where:

- m_i is the mass of link i.
- v_i is the linear velocity of the center of mass of link i.
- ω_i is the angular velocity of link i.
- I_i is the inertia tensor of link i.

Or:

The potential energy, being a function of the joint variables q, The torque τ can be expressed as:

$$A(q) \bullet \ddot{q} + V(q, \dot{q}). \bullet + G(q) = \Gamma \quad (4.5)$$

V.4. Forward Dynamic Model

$$\Gamma = A(q) \bullet \ddot{q} + B(q) \bullet [\dot{q}.\dot{q}] + C(q) \bullet [\dot{q}^2]) \bullet \dot{q} + G(q) \quad (4.6)$$

Where:

- q: Position vector.
- A(q): Inertia matrix of order (n×n).
- $V(q, \dot{q})$:Centrifugal and Coriolis force vector of order (n×1).
- G(q): Gravity vector of order (n×1).
- τ Torque vector of order (n×1).

By expressing the velocity-dependent term $V(q, \dot{q})\dot{q}$ in a different form, all the matrices become functions of only the position of the manipulator. In this case, the dynamic equation is called the configuration space equation and is described in the following form:

$$\Gamma = A(q) \bullet \ddot{q} + B(q) \bullet [\dot{q}.\dot{q}] + C(q) \bullet [\dot{q}^2] \bullet \dot{q} + G(q) \text{ ..} \quad (4.7)$$

4.2.4 The Inertia Matrix

The inertia matrix A(q) is a symmetric square matrix of order 6:

$$A(q)=\begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 & 0 \\ a_{12} & a_{22} & a_{23} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & a_{35} & 0 \\ 0 & 0 & 0 & a_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{66} \end{bmatrix} \quad (4.8)$$

Or:

$$a_{11} = I_{m1} + I_1 + I_3.CC2 + I_7.SS23 + I_{10}.SC23 + I_{11}.SC2 + I_{21}.SS23 + \\ + 2.[I_5.C2.S23 + I_{12}.C2.C23 + I_{15}.SS23 + I_{16}.C2.S23 + I_{22}.SC23] = 3.76949$$

$$a_{12} = I_4.S2 + I_8.C23 + I_9.C2 + I_{13}.S23 - I_{18}.C23 = -0.13707$$

$a_{21} = a_{12}$,

$a_{31} = a_{13}$

$a_{32} = a_{23}$

$$A = \begin{bmatrix} 3.9273 & -0.1107 & -0.1344 & 0 & 0 & 0 \\ -0.1107 & 6.7729 & 0.3242 & 0 & 0 & 0 \\ -0.1344 & 0.3242 & 1.1625 & 0 & 0.0019 & 0 \\ 0 & 0 & 0 & 0.2016 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1796 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1930 \end{bmatrix}$$

4.2.5 Centrifugal force matrix

$$C(q) = \begin{bmatrix} 0 & c_{12} & c_{13} & 0 & 0 & 0 \\ c_{21} & 0 & c_{23} & 0 & 0 & 0 \\ c_{31} & c_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ c_{51} & c_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.9)$$

Or:

$$c_{12} = I_4.C2 - I_8.S23 - I_9.S2 + I_{13}.C23 + I_{18}.S23$$

$$c_{13} = 0.5.b_{123} = -I_8.S23 + I_{13}.C23 + I_{18}.S23$$

$$c_{21} = -0.5.b_{112} = I_3.SC2 - I_5.C223 - I_7.SC23 + I_{12}.S223 - I_{15}.2.SC23 - I_{16}.C223 - I_{21}.SC23 - - I_{22}.(1-2.SS23) - 0.5.I_{10}.(1-2.SS23) - 0.5.I_{11}.(1-2.SS2)$$

$$c_{23} = 0.5.b_{223} = -I_{12}.S3 + I_5.C3 + I_{16}.C3$$

$$c_{31} = -0.5.b_{113} = -I_5.C2.C23 - I_7.SC23 + I_{12}.C2.S23 - I_{15}.2.SC23 - I_{16}.C2.C23 - I_{21}.SC23 - - I_{22}.(1-2.SS23) - 0.5.I_{10}.(1-2.SS23)$$

$$c_{32} = -c_{23} = I_{12}.S3 - I_5.C3 - I_{16}.C3$$

$$c_{51} = -0.5.b_{115} = SC23 - I_{15}.SC23 - I_{16}.C2.C23 - I_{22}.CC23$$

$$c_{52} = -0.5.b_{225} = -I_{16}.C3 - I_{22}$$

$$C = \begin{bmatrix} 0 & 0.68608 & -0.00391 & 0 & 0 & 0 \\ -0.35529 & 0 & 0.37323 & 0 & 0 & 0 \\ -0.36223 & -0.37323 & 1.1625 & 0 & 0.0019 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -0.00214 & -0.00118 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

4.2.6 Gravity vector G

$$G(q) = \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \tag{4.10}$$

$$g_2 = g_1.C2 + g_2.S23 + g_3.S2 + g_4.C23 + g_5.S23 = -8.44$$

$$g_3 = g_2.S23 + g_4.C23 + g_5.S23 = 1.02$$

$$g_5 = g_5.S23 = -0.0282$$

$$G(q) = \begin{bmatrix} 0 \\ -8.44 \\ 1.02 \\ 0 \\ -0.0282 \end{bmatrix}$$

Ou,

$S_i = sin(\theta_i)$,

$C_i = cos(\theta_i)$,

$C_{ij} = cos(\theta_i + \theta_j)$,

$S_{ijk} = sin(\theta_i + \theta_j + \theta_k)$,

$CC_i = cos(\theta_i).cos(\theta_i)$

$Csi = cos(\theta i).sin(\theta i)$

I_1	1.43 ± 0.05
I_2	1.75 ± 0.07
I_3	1.38 ± 0.05
I_4	0.69 ± 0.02
I_5	0.372 ± 0.031
I_6	0.333± 0.016
I_7	0.298 ± 0.029
I_8	-0.134 ± 0.014
I_9	0.0238 ± 0.012
I_{10}	-0.0213 ±0.0022
I_{11}	-0.0142 ± 0.0070
I_{12}	-0.011 ± 0.0011
I_{13}	-0.00379 ± 0.0009

. .

I_{14}	0.00164 ± 0.000070
I_{15}	0.00125 ± 0.0003
I_{16}	0.00124 ± 0.0003
I_{17}	0.00642 ± 0.0003
I_{18}	0.00043 ± 0.00031
I_{19}	0.0003± 0.0014
I_{20}	-0.000202 ± 0.02008
I_{21}	-0.001 ± 0.0006
I_{22}	-0.000058 ± 0.000015
I_{23}	0.00004 ±0.00002
I_{m1}	I_{m1}= 1.14 ± 0.27
I_{m2}	I_{m2}=4.71 ± 0.54
I_{m3}	I_{m3}=0.827 ± 0.093
I_{m4}	I_{m4}=0.2 ± 0.016
I_{m5}	I_{m5}=0.179 ± 0.014
I_{m6}	I_{m6}=0.193 ± 0.016

Table 4.1: moment of inertia (Kgm^2)[23]

g_1	-37.2 ± 0.5
g_2	-8.44 ± 0.20
g_3	1.02 ± 0.50
g_4	0.249 ± 0.0255
g_5	-0.028 ± 0.0056

Table 4.2: gravity constants (N.m) [23]

.

The PUMA robot has the same general configuration space equation as its 6-DOF counterpart. In this configuration, the last three joints are locked, maintaining their initial states while the robot moves (this is for the simulation of the first three joints). By using the robot's configuration equation and setting the last joints to zero, we can define a general equation that allows us to use the PUMA robot.

4.3 Conclusion

In this chapter, we presented the dynamic modeling formalism for simple open-structure manipulators and applied Lagrange's method to the PUMA 560 robot.

Acceleration induces new forces known as inertia forces. Since the PUMA 560 consists solely of rotary joints, we also account for centrifugal and Coriolis forces. This model will be essential in the next chapter, where we will develop a control law for the robot and convert these equations into a computational model using MATLAB.

Chapter 5

Control of the Manipulator Robot

5.1 Introduction

In the field of robotics, the control of manipulator robots is a crucial aspect, ensuring that these complex machines can execute tasks with precision and efficiency. The control problem involves determining the generalized forces (forces or torques) that actuators must apply to achieve the desired motion, while also meeting performance criteria such as accuracy, stability, and responsiveness. This chapter discusses various control techniques used for manipulator arms, focusing on classical control, Jacobian control, nonlinear decoupling control, adaptive control, Lyapunov-based control, passive control, and predictive control.

This thesis focuses on robots that utilize servomotors with high reduction ratios, resulting in rigid joints. Joint rigidity is essential, particularly during interactions with the environment or collisions, where inaccuracies in modeling can cause significant contact forces, potentially damaging the robot or its surroundings. We provide an overview of various control techniques documented in the literature, describing each method in the following sections.

5.2 Control of Movements

The control of manipulator robots has been the focus of extensive research. The main methods employed include:

- Classical PID control
- Nonlinear decoupling control
- Passive control
- Lyapunov-based control
- Adaptive control
- Robust variable structure control (sliding modes)

While it is beyond the scope of this chapter to cover all these approaches in detail, we will focus on classical PID control (proportional, integral, and derivative control), which is often considered an ideal theoretical solution for controlling manipulator robots .

5.3 Regulation

5.3.1 Definition

Regulation involves controlling various system actions to correct malfunctions related to design or implementation, ensuring efficient and effective operation.

5.3.2 Objectives of Regulation

Proper Execution of a Process: Ensures consistent and reliable operations by maintaining control variables within setpoints.

System Stability: Maintains stability by preventing fluctuations and oscillations, ensuring smooth and continuous operation.

Improved Accuracy: Enhances precision by continuously comparing setpoints with measured values and making necessary adjustments.

Enhanced Product Quality: Improves product quality by keeping process parameters within desired ranges, reducing errors and defects.

5.4 PID Regulation

5.4.1 Definition of the PID Controller

The PID controller, also known as the PID corrector (Proportional, Integrator, Derivator), is the most widely used control method in the industry. It is a closed-loop control system that improves the performance of a system or process. This regulator is extensively applied in various industrial sectors, where its correction capabilities are utilized across numerous physical dimensions .

5.4.2 General Principle

A PID controller incorporates three control actions: Proportional (P), Integral (I), and Derivative (D). It is the most commonly used regulator in the industry, capable of adjusting many physical variables. The PID controller combines the beneficial effects of the three fundamental control actions. The integral term (I) eliminates steady-state error, while the derivative term (D) enhances the system's response speed compared to a basic proportional controller.

By leveraging the strengths of all three components, the PID controller provides a robust solution for maintaining desired system performance across various applications.

5.4.3 Classical Control

Classical control techniques are commonly used in manipulator robots equipped with high reduction ratio servomotors. When the system exhibits linear behavior, traditional methods like decentralized PID control can effectively manage movement. Enhancements to the classical scheme, such as those noted by Gorla et al. (1984) and Canudas et al. (1997), include feedforward signals to mitigate the effects of gravity and coupling forces [25]. Assuming that the joint positions and velocities are measurable and the measurements are noise-free, the control law can be formulated as follows:

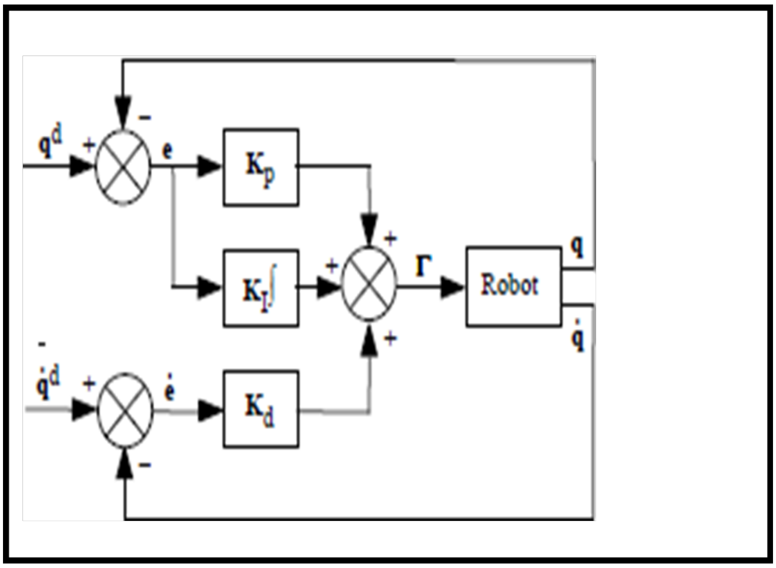

Figure 5.1: Figure :classical pid controller [11]

$$\Gamma = K_p(q^d - q) + K_d(\dot{q}^d - \dot{q}) + K_I \int^c (q^d - q)dr \tag{5.1}$$

Tracking Error

$$\mathrm{e}(t) = q^d(t) - q(t) \tag{5.2}$$

Where:

- q∧d(t) represents the desired positions in the joint space.
- q ∧d(t) represents the desired velocities in the joint space.
- Kd,Kp and Ki are positive definite diagonal matrices of dimension (n ×n), with the generic elements being the proportional gains Kpj, derivative gains Kdj, and integral gains Kij, respectively.

5.4.4 Nonlinear Decoupling Control

Nonlinear Decoupling Control, also known as dynamic control or computed torque control, is used in applications that require rapid movements and involve dynamic constraints. This technique accounts for interaction forces by considering all joints collectively and applies nonlinear decoupling theory. By leveraging the robot's dynamic model, it formulates control laws that result in centralized nonlinear control. Feedforward signals are often employed to minimize nonlinear effects [27].

This approach enables control in both joint space and Cartesian space, allowing decoupled joints to achieve high-speed movements with substantial inertia. However, its effectiveness heavily depends on the accuracy of the system model; any discrepancies can lead to imperfect decoupling, which is a significant limitation.

5.4.5 Jacobian Control

Pioneered by Whitney [28], Jacobian Control uses the inverse Jacobian matrix of the manipulator to determine the desired joint velocities. Also referred to as resolved motion control, this method includes approaches like resolved velocity control, resolved acceleration control [29], and resolved force control. It manages the end effector's position in Cartesian space by coordinating the movements of multiple joints.

5.4.6 Adaptive Control

Adaptive control techniques aim to address the shortcomings of nonlinear decoupling control, such as the need for precise knowledge of the robot model parameters or adaptability to varying operational conditions. These methods seek to estimate or adjust parameter values online to improve the accuracy of the control law. Notably, [27] proposed the Slotine-Li control, also known as passive adaptive control. Extensive research on adaptive control is presented by [27]. Despite its advantages, adaptive control demands significant computational power, which is a major drawback.

5.4.7 Lyapunov-Based Control

Lyapunov-based methods have been effectively used to control manipulator arms for tracking tasks. These techniques ensure asymptotic convergence without the need to linearize the system or achieve decoupling, as highlighted by [27].

5.4.8 Passive Control

Passive control treats the robot as a passive system, one that dissipates energy. These control laws modify the robot's natural energy to accomplish tasks. Using Hamiltonian formalism, passive control aims to minimize system energy by incorporating a nonlinear passive block in the feedback loop. This approach tends to be more robust than nonlinear decoupling, especially when the goal is not to cancel out nonlinearities [27].

5.4.9 Predictive Control

Predictive control uses the system model and setpoints to predict the system's future behavior, allowing it to act based on prediction errors. [27] propose three different schemes: fixed endpoint, finite horizon, and a combination of both. A significant advantage of this control type is its ability to proactively manage system evolution.

5.5 Simulation

This section details the simulation results for the puma 560 robot, which employs a PID controller for each of its rotary joints. The PID parameters were calibrated to enhance the system's response and reduce tracking errors. We evaluated the robot's performance by using two different reference inputs for the control variables. The evaluation focused on the desired trajectories and the corresponding tracking errors. For the simulation, two types of reference inputs were specified for the control variables q1,q2,q3,q4,q5,q6 :

Reference 1:

- For $t<1.5$ seconds, the values of q1,q2,q3,q4,q5,q6 are all zero.
- For $t\succeq 1.5$ seconds, the values of q1,q2,q3,q4,q5,q6 are all equal to 2.

Reference 2:

he values of q1,q2,q3,q_1, q_2, q_3,q1,q2,q3, and q4q_4q4 follow a third-degree polynomial trajectory .

And we used two types of reference trajectories to evaluate the robot's performance:

Reference 1:

XE=0.1*cos(T)+0.2;

YE=0.12*sin(T)+0.1;

ZE=0.05*cos(T)-0.07;

Reference 2:

XE=0.2*$e^{0.002*T}$;

YE=0.22*$e^{0.002*T}$;

ZE=1.5T;

5.6 Simulation Results for QREF

5.6.1 Simulation Results (Reference 1):

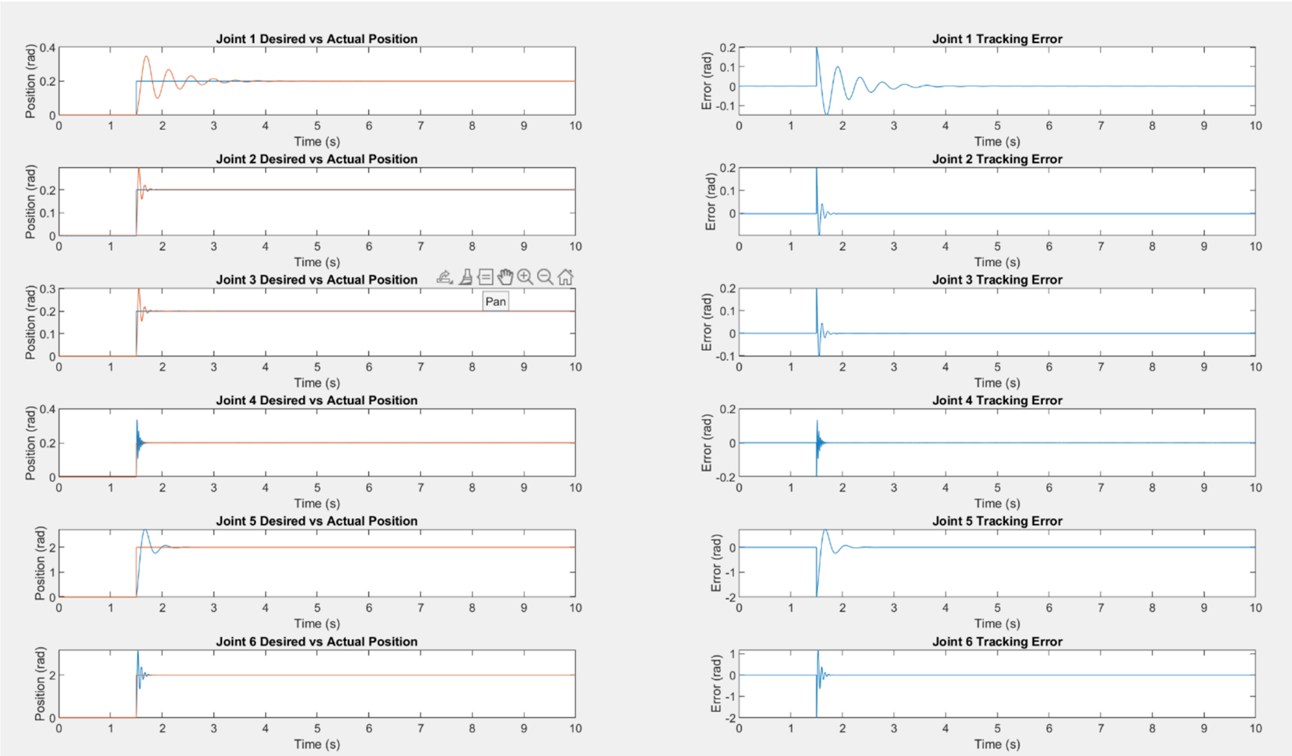

Figure 5.2: Simulation Results (Reference 1)

5.6.2 Simulation Results (Reference 2):

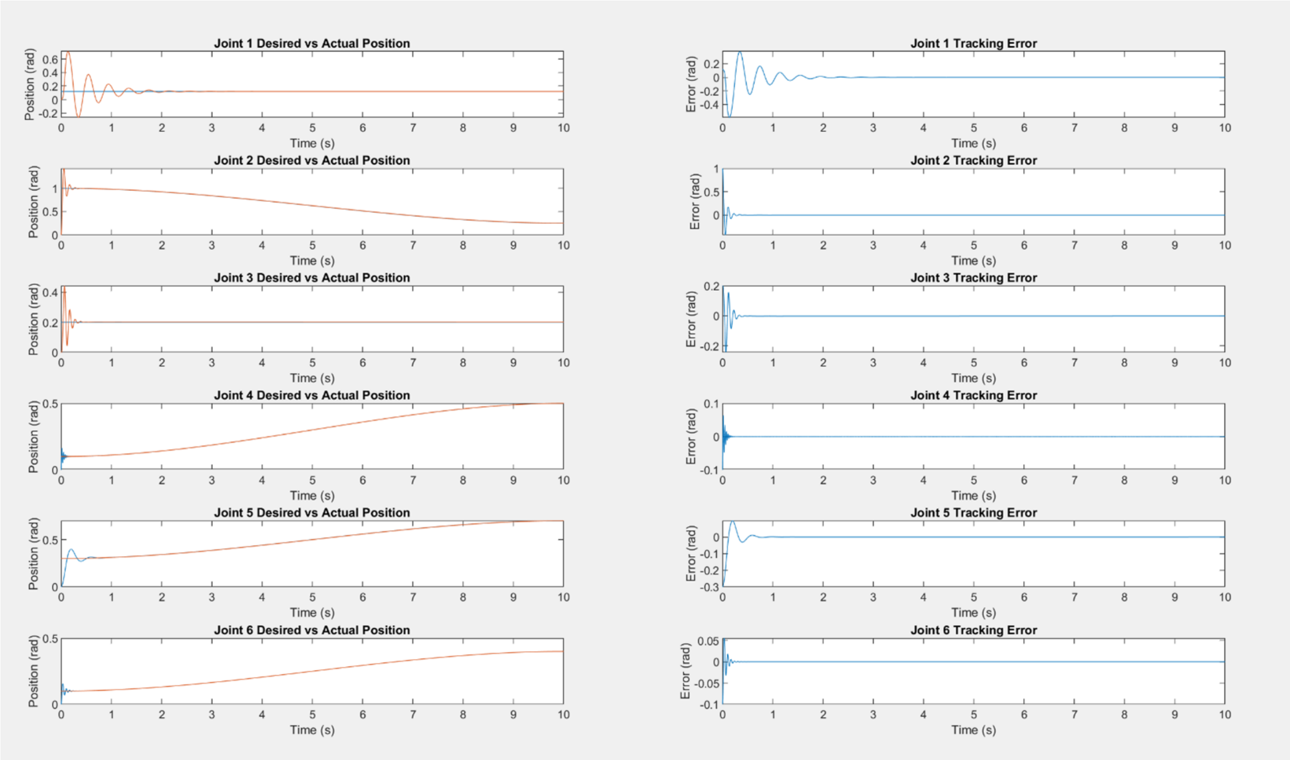

Figure 5.3: Simulation Results (Reference 2)

The results shown in Figure 5.2 and Figure 5.3 demonstrate the control system's effective ability to make the joints of the PUMA 560 robot follow the desired values q(d). The observed discrepancies between q(d) and q (e1, e2, e3, e4, e5, e6) exhibit limited oscillations, indicating an efficient response of the regulator (likely a PID) in maintaining the joints close to the desired positions.

5.7 resultat de simulation pour (XREF)

5.7.1 Simulation Results (Reference 1)

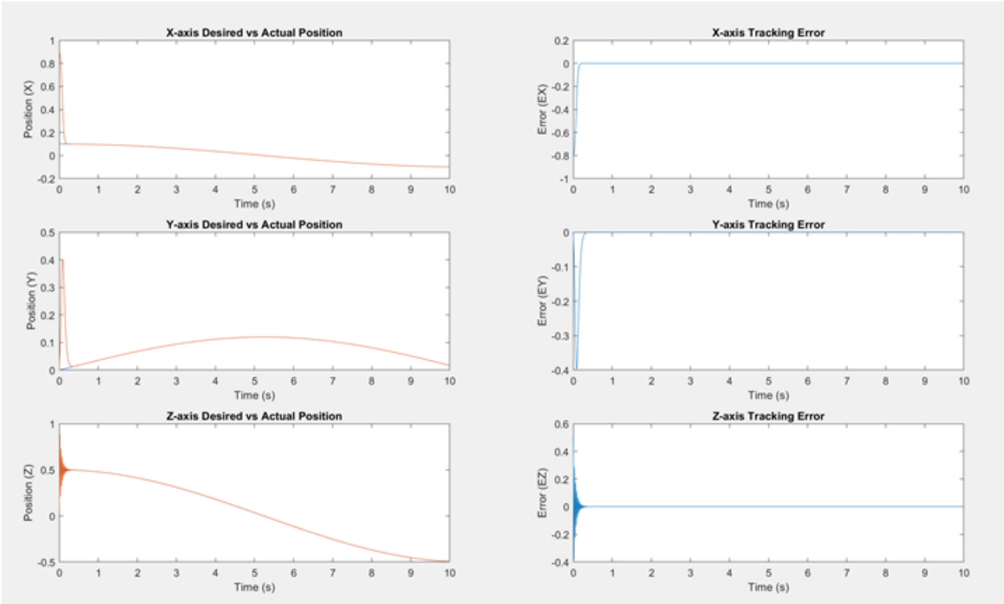

Figure 5.4: Cartesian Trajectory Tracking and Errors for Positions XE, YE, ZE (REFERANCE 1)

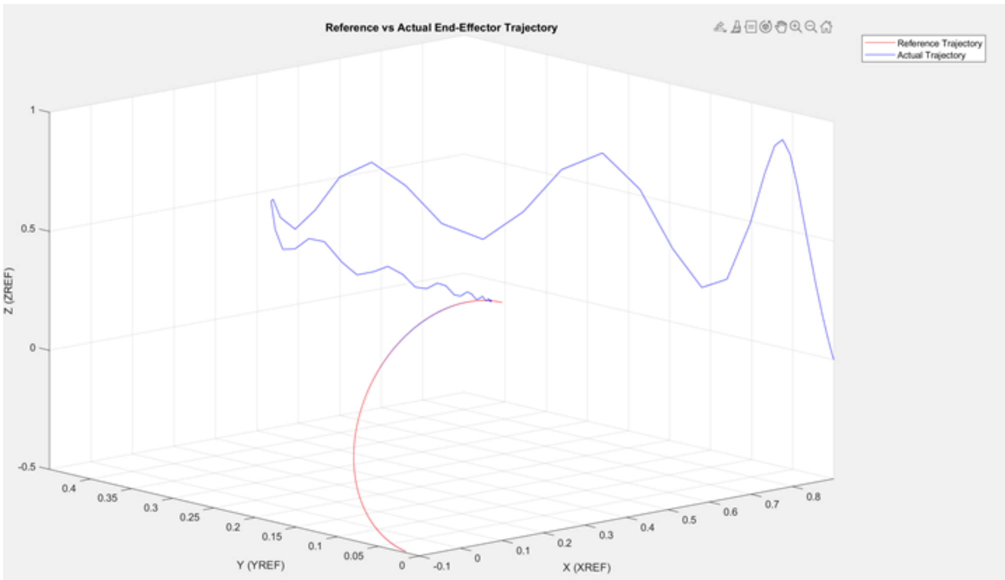

Figure 5.5: 3D trajectory of the simulation(Ref trajectory vs actual trajectory)

The position trajectory in X (XE), Y (YE), and Z (ZE) closely follows the reference trajectory (XEref, YEref, ZEref), despite minor oscillations that are quickly corrected.

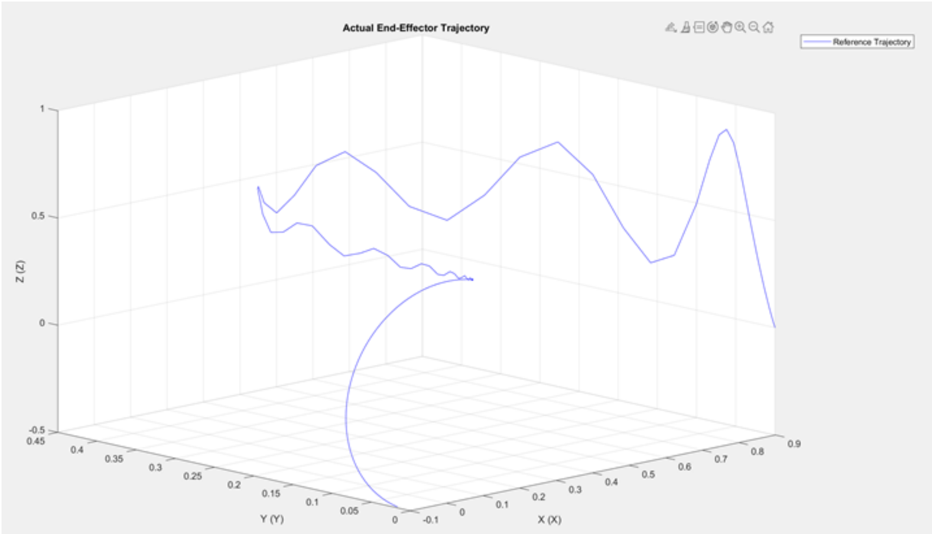

Figure 5.6: 3D trajectory of the simulation(actual trajectory)

The errors e1, e2, e3, e4, e5, e6 remain low and stabilize rapidly, demonstrating the PID controller's effective ability to correct deviations.

5.7.2 simulation result (Reference 2) :

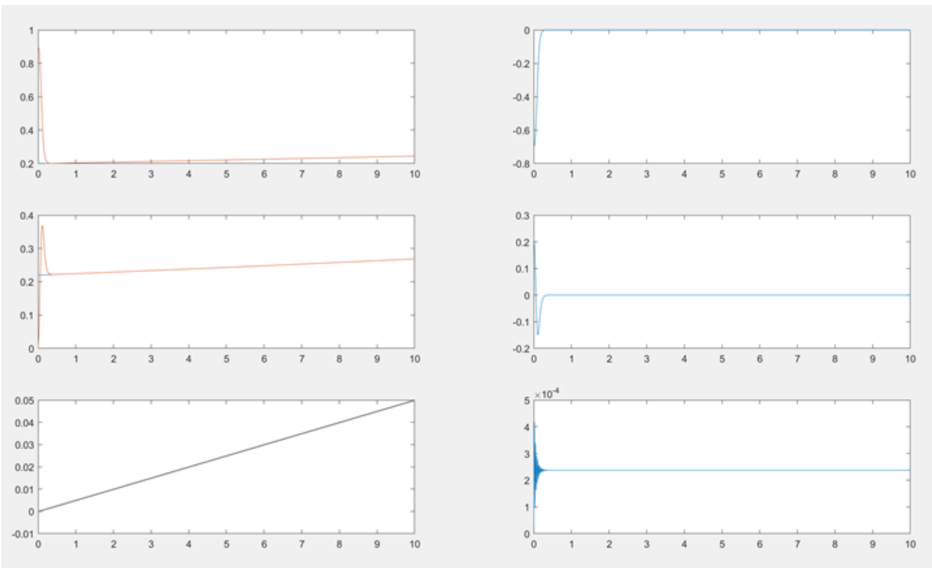

Figure 5.7: Cartesian Trajectory Tracking and Errors for Positions XE, YE, ZE

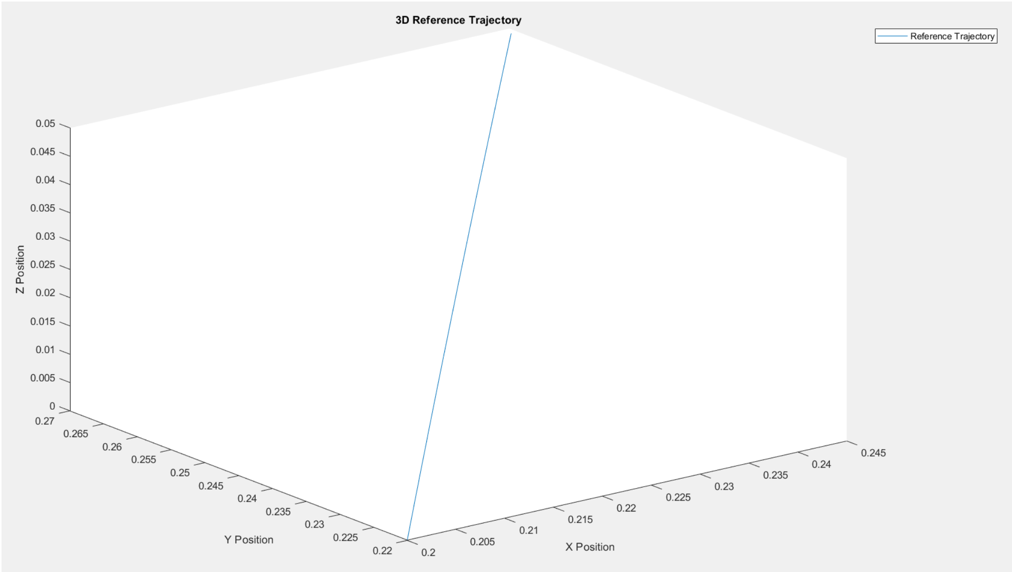

Figure 5.8: 3D Reference Trajectory

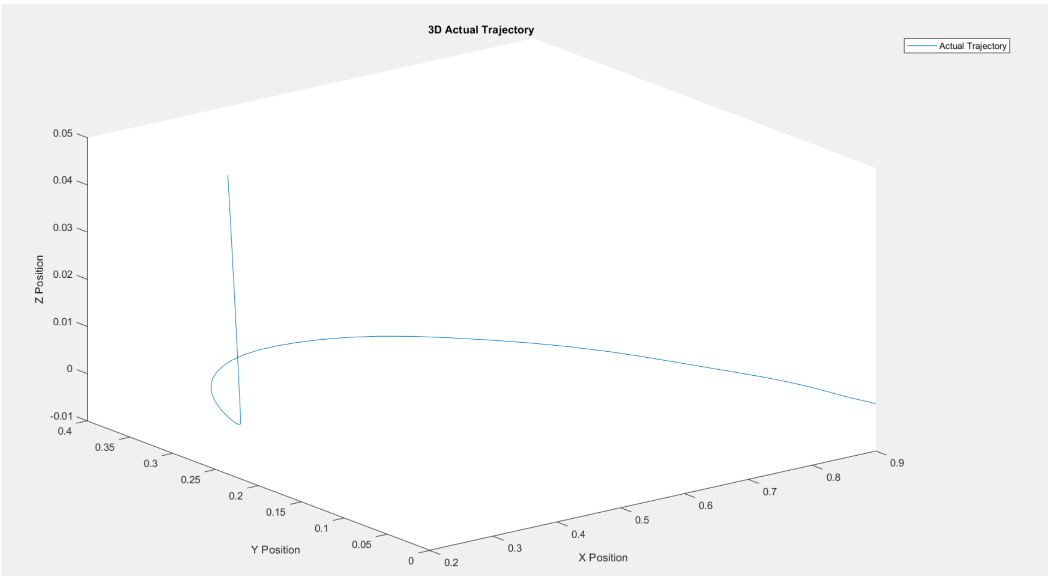

Figure 5.9: 3D Actual Trajectory

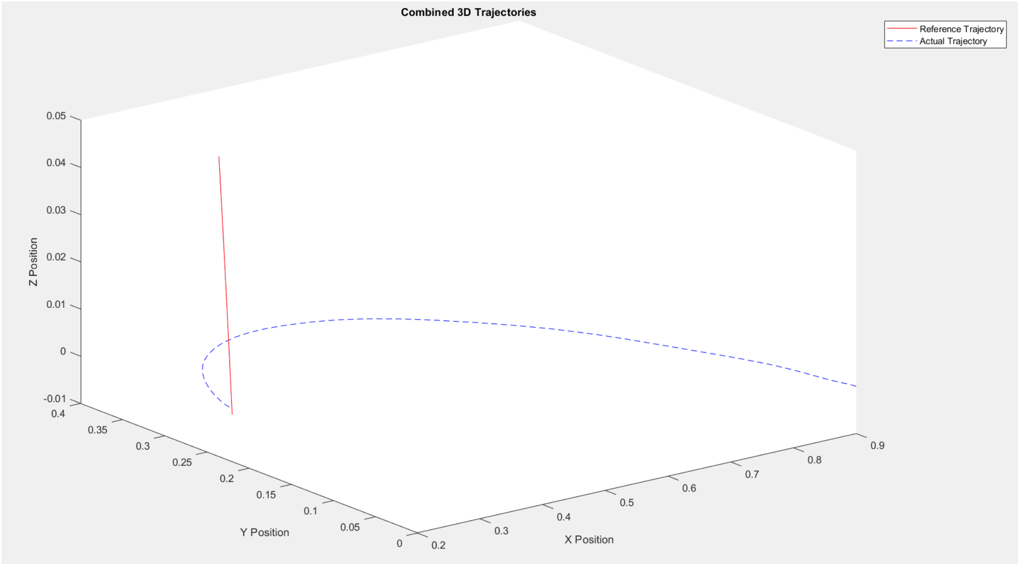

Figure 5.10: 3D Trajectory of Cartesian Positions for a Robot puma 560

The robot effectively follows complex trajectories defined by third-degree polynomials, demonstrating the robustness and efficiency of the PID control used. Overall, the simulations show that the Puma robot can efficiently track desired trajectories with minimal errors, regardless of whether the references are simple or complex. The PID controller is crucial in achieving this performance, ensuring both precision and stability of the system.

5.8 Conclusion

In this chapter, we introduced a PID control technique for the PUMA560 manipulator robot. We demonstrated a MATLAB simulation for controlling trajectories and velocities of the robot. The simulation enables visualization of trajectories and velocities for multiple setpoints. We utilized three reference trajectories in our study.

The system's divergence is attributed to the non-diagonal nature of the inertia matrix, which heavily depends on the configuration q. Moreover, significant centrifugal and Coriolis forces can occur at high velocities.
Due to these factors, employing PID control results in variable speed and precision performances that are challenging to predict due to the highly nonlinear characteristics of the controlled process.

General Conclusion

The control of manipulator robots is a primary focus of robotics research. Manipulator robots exhibit nonlinear behavior and are tasked with operations that demand high precision and rapid trajectory execution. To improve manipulator performance, this work presents the classical PID control, a well-established control law in robotics that requires an accurate dynamic model of the robot.

The goal of this work is to model and control a 6-DOF industrial manipulator robot, specifically the PUMA 560. The project begins with direct geometric modeling of robots with simple open structures, using Euler angles to represent rotational coordinates. Denavit-Hartenberg (D-H) parameters and careful frame selection were employed for this purpose.

The second part addresses the inverse geometric problem using Paul's method coupled with kinematic decoupling techniques. The following section delves into the kinematic analysis of the robot, including the computation of the Jacobian matrix and the determination of its inverse kinematic model. These principles were then applied to the selected robot.

Subsequently, the dynamic modeling was completed, providing a direct dynamic model of the robot, though it remains uncertain due to tolerance variations. The study then explored control application in joint space.
From the control schemes, three observations emerged: First, joint space control is easy to implement but is limited to controlling one axis at a time, as the tool's trajectory cannot be visualized. Three reference trajectories were presented, and MATLAB simulation results

were displayed.

This work does not aim to exhaustively address all constraints but proposes a control scheme that facilitates the execution of various tasks while adhering to kinematic constraints, using proprioceptive sensor information. In this specific study, only position and velocity were considered.

Future work

This work opens several avenues for future research:

- Simplifying the components of all models within the control loop to minimize execution time and facilitate implementation on a computer.
- Avoiding the use of step-type trajectories when controlling robot actuators.
- The provided program can control any manipulator with an (RRR) architecture; users need only update the inertial and gravity parameters.
- While proper tuning of PID parameters produces satisfactory results, significant oscillations often remain.
- Simultaneous control of the robot's axes in joint space has some drawbacks concerning the precision expected of the robot. Implementing artificial intelligence techniques could yield better results.

References

[1] P. COIFFET "les robots Tome 2", interaction avec l'environnement, septembre 1981.

[2] W. KHALIL , E. DOMBRE "Bases de la modélisation et de la commande des robots-manipulateurs de type série", date de création 08/05/2012

[3] W. Khalil , E. Dombre " Modelisation, identification and control of robots, Hermes Penton". Science, London, 2002, page 480

[4] https://robotsguide.com/robots/unimate/, consulted 2024

[5] M. Spong, W. Spong, M. Vidyasagar " Robot dynamics and control, John Wiley & Sons" New York, USA, 1989.

[6] Unimation (Europe) Ltd., "PUMA 560 Mk2 Robot System Technical Manual", Sept. 1985.

[7] C. Lee S. Lee G. Lee& Ziegler "Geometric Approah in Solving Inverse Kinematics for the PUMA Robot", IEEE trans. on Aerospace and Electronic Systems, Volume AES-20, No. 6, Nov. 1984.

[8] Khachouche Samir ," Conseption et Realisation d'un Bras Manipulateur De Type PUMA quatre Dégréé de Liberté ,2018

[9] https://www.larousse.fr/encyclopedie/images/Robot_industriel/1006457/, consulted 2024

[10] https://www.sciencedirect.com/topics/computer-science/industrial-robot/, consulted 2024

[11] S. Chehaidia " Modelisation et Commande D'un Robot Manipulateur a six Degres De Liberte" 2016

[12] Lee C.S.G. & Ziegler, "Geometric Approah in Solving Inverse Kinematics for the PUMA Robot", IEEE trans. on Aerospace and Electronic Systems, Volume AES-20, No. 6, Nov. 1984.

[13] K. Becir "Cours de modélisation des robots industriels, université Badji mokhtar annaba".

[14] https://www.researchgate.net/figure/The-PUMA-robot-from-16_fig5_329590080

[15] W. Khalil, E. Dombre, « Modélisation, Identification et Commande des Robots », 2ième édition, Hermes Science Publication, 1999.

[16] https://www.phisterequipment.com/mecademic-robots.html/,2024

[17] https://www.sciencedirect.com/science/article/abs/pii/S0736584506000792/, 2024

[18] R.C.P. Paul " Robot manipulators: mathematics, programming and control" MIT Press, Cambridge, 1981

[19] D.L. Pieper , "The kinematics of manipulators under computer control", Ph. D. Thesis, Stanford University, 1968

[20] P. Coiffet " Les Robots ; Tome 1 : Modélisation et commande, Hermès" Paris, 1981.

[21] M. Vukobratovic , V. Potkonjak " Dynamics of manipulation robots; Vol. 1: Theory and applications,Springer-Verlag" New York, 1982.

[22] M. Žefran, F. Bullo « Lagrangian Dynamics », "Robotics and Automation Handbook", CRC Press LLC publication 2005.

[23] D.L. Pieper , "The kinematics of manipulators under computer control", Ph. D. Thesis, Stanford University, 1968

[24] B. Armstrong, O. Khatib, J. Burdick "The Explicit Dynamic Model and Inertial Parameters of the PUMA 560 arm", Stanford Artificial Intelligence Laboratory Stanford University

[25] F. Makhloufi , "Modélisation et commande des robots manipulateurs par les outils de l'intelligence artificielle", thèse présentée pour obtention du diplôme de DOCTORAT, université Badji Mokhtar Annaba ,2015.

[26] G. Rooryck "robotique et dynamique industrielle, quand et comment robotiser", Hermes publishing (France) 1981

[27] A. Lounis "TP Modélisation et Commande des Robots", douane , 2024

[28] D.E. Whitney "Resolved motion rate control of manipulators and human prostheses". IEEE Trans, on Man Machine Systems, Vol. MMS-10(2). juin 1969. p. 47-53.

[29] J. Luh, M. Walker, et R. Paul "Resolved acceleration control of mechanical manipulators". IEEE Trans. on Automatic Control (ITAC), 25(3):468–474. (1980).