



**POLITECNICO  
DI TORINO**

POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Mechatronica  
Dipartimento di Automatica e Informatica

**KINEMATIC ANALYSIS OF A 7 DOF  
ANTHROPOMORPHIC ROBOTIC  
ARM HAVING FINGERS MODELLED  
UNDER RIGID-BODY MECHANICS**

**Supervisor:** Prof. PAOLO PRINETTO  
**Co-Supervisor:** Ing. MARINA INDRI

**Candidate:**  
DEEPTI NITHYANAND PRABHU  
ID Number: 225085

Academic year 2017-2018



*To my grandfathers  
Mr. V. L. Prabhu  
and  
Dr. V. S. Kenkare*

# Acknowledgement

Firstly, I would like to thank and express my sincere gratitude to supervisor Prof. Paolo Prinetto and my co-supervisor Ing. Marina Indri for giving me the opportunity to work under their noble guidance.

I am truly grateful to Dr. Giuseppe Airò Farulla and Mr. Andrea Bulgarelli for their constant guidance, encouragement and valuable advice time and again.

I am thankful to the Mechatronics Engineering course at Politecnico di Torino for letting me get this experience which will be beneficial throughout my future life. Finally, I would like to thank my family and friends, for their unconditional support without which this would not have been possible.

# Abstract

An anthropomorphic arm, is a programmable structure with functions similar to those of a human arm. These arms can be used to perform a variety of tasks with great accuracy. The first step in the robotic arm design is developing a proper mathematical model for the motion of the mechanism. This work of thesis, which lies within the PARLOMA framework, focuses on the kinematic aspect of an anthropomorphic arm developed to convey information through the exchange of tactile sign languages. There is also enough considerations given to model the mechanical system of a robotic finger.

The PARLOMA project aims to design and develop a low-cost system for remote communication between deaf-blind people. The focus is on the development of arm. The requirements posed by such challenging project are carefully revised. Then the thesis can broadly be divided into three main categories:

The direct kinematics: the arm has seven revolute joints and seven degrees of freedom: three at shoulders, one at elbow and three at the wrist. The direct kinematics is calculated according to the Denavit Hartenberg convention.

The inverse kinematics: This is modelled by using the redundancy parameterization. The arm angle is used to calculate the unique values for the computation of the joint angles.

Modelling of the robotic fingers: To make the hand cost effective and light weight, we need to use the minimum possible mechanical components. Using the Lagrangian method, three generalized differential equations are calculated. These equations can now be used to calculate the values of any unknown parameter in the system, like the spring constants.

# Contents

<b>Acknowledgement</b>	<b>III</b>
<b>Abstract</b>	<b>IV</b>
<b>List of Figures</b>	<b>VII</b>
<b>List of Tables</b>	<b>IX</b>
<b>1 Introduction</b>	<b>1</b>
1.1 The PARLOMA project . . . . .	2
1.1.1 Tactile sign language . . . . .	2
1.1.2 The architecture of the system . . . . .	5
1.1.3 Input Module . . . . .	6
1.1.4 Output Module . . . . .	7
1.2 Aim of the thesis . . . . .	8
1.2.1 General findings . . . . .	8
1.2.2 Research outline . . . . .	9
<b>2 Literature review</b>	<b>10</b>
2.1 History . . . . .	11

2.1.1	Leonardo Da Vinci’s Robotic Arm . . . . .	11
2.1.2	Automata to the Industrial Revolution . . . . .	13
2.1.3	Robots of Expositions and Industrial Robotic Arms	15
2.2	Anatomy of a human hand . . . . .	21
2.3	Progress in the field of robotic hand . . . . .	22
2.3.1	EthoHand . . . . .	22
2.3.2	SmartHand . . . . .	23
2.3.3	Spring Hand . . . . .	23
2.3.4	Blackfinger . . . . .	24
2.4	The PARLOMA hand . . . . .	25
<b>3</b>	<b>Overview and requirements of the modelling system</b>	<b>27</b>
3.1	Basics for the modelling system . . . . .	27
3.1.1	End effector and Cartesian/Euclidean space . .	27
3.1.2	Degrees of freedom and redundancy . . . . .	28
3.1.3	Rotation matrix and reference frame . . . . .	28
3.1.4	Homogeneous Transformation matrix . . . . .	31
3.2	Requirements of Robotic Arm . . . . .	32
3.3	Requirements for the robotic finger . . . . .	33
<b>4</b>	<b>Direct Kinematics</b>	<b>35</b>
4.1	Methods available . . . . .	38
4.1.1	Denavit Hartenberg convention . . . . .	38
4.1.2	Neural Network analysis . . . . .	40
4.1.3	Quasi-closed solution method . . . . .	40

4.1.4	Taylor series expansion . . . . .	41
4.2	Method applied in the case of a 7 DOF robotic arm . .	41
4.3	Results achieved . . . . .	41
<b>5</b>	<b>Inverse Kinematics</b>	<b>47</b>
5.1	Methods available . . . . .	47
5.1.1	Redundancy parameterization . . . . .	47
5.1.2	Jacobians . . . . .	48
5.1.3	Cyclic coordinate descent . . . . .	48
5.2	Method applied in the case of a 7 DOF robotic arm . .	48
5.3	Method applied . . . . .	48
5.4	Results achieved . . . . .	49
5.4.1	Joint angles . . . . .	52
<b>6</b>	<b>Robotic finger mechanism</b>	<b>61</b>
6.1	Approaches . . . . .	61
6.1.1	Bond-Graph method . . . . .	61
6.1.2	Free body diagram . . . . .	62
6.1.3	Lagrange method . . . . .	62
6.2	Method used for modelling . . . . .	62
6.3	Results achieved . . . . .	64
<b>7</b>	<b>Conclusion</b>	<b>69</b>
7.1	Future work . . . . .	70
	<b>Bibliography</b>	<b>71</b>

# List of Figures

1.1	PARLOMA . . . . .	2
1.2	Alphabets of the Italian Sign Language . . . . .	4
2.1	Leonardo da Vinci's automobile, courtesy of Biblioteca Ambrosiana in Milan . . . . .	12
2.2	De Vaucanson's <i>Androide</i> . . . . .	13
2.3	Jaquet-Droz's musician . . . . .	14
2.4	Kempelen's <i>Turk</i> , with sophisticated left arm mechanisms	15
2.5	Early modern robotic arm-the sketch of Pollard's arm .	16
2.6	The Exposition robots: <i>far left</i> , Harry May's Alpha; <i>right</i> , three of the Westinghouse <i>Elektros</i> . . . . .	17
2.7	Unimate robotic arm . . . . .	17
2.8	Sophisticated Robotic Arms . . . . .	18
2.9	Aichi Expo, Japan in 2005 with the robot exhibits . . .	20
2.10	Anatomy of a human hand . . . . .	21
2.11	EthoHand . . . . .	22
2.12	SmartHand . . . . .	23
2.13	SpringHand . . . . .	24
2.14	Blackfinger . . . . .	24

2.15	PARLOMA hand . . . . .	25
3.1	Two RF's $\mathbb{R}_a$ and $\mathbb{R}_b$ with a common origin but different orientation . . . . .	29
3.2	Translation with respect to a frame . . . . .	31
4.1	7 DOF manipulator . . . . .	36
4.2	Denavit-Hartenberg convention . . . . .	38
4.3	The coordinate frames assigned to the manipulator . . . . .	42
5.1	Arm angle . . . . .	49
5.2	Elbow joint angle . . . . .	52
5.3	Shoulder joint angle . . . . .	54
6.1	Spring loaded finger . . . . .	63

# List of Tables

3.1	Comparison between robotic manipulators by manufacturers . . . . .	34
4.1	Denavit Hartenberg parameters . . . . .	39
4.2	Comparison between robotic manipulators by manufacturers . . . . .	43
6.1	List of symbols . . . . .	63

# Chapter 1

## Introduction

An anthropomorphic robotic arm is a bio-inspired mechanical structure which mimics (most of) the form and movement of a human arm. The anthropomorphic arm which is sufficiently compact, powerful and flexible to be functional is to be designed. The number of degrees of freedom endowed in a robotic arm is what makes it capable of executing a wide range of tasks, including dexterous hand movements. A well developed robotic arm is capable of sensing and interpreting incoming external force to perform a plethora of tasks that are used in most of the industries today. To accomplish a desired task by the manipulator, we need to study the robots kinematics structure, so that hand is able emulate the capability of the human hand by providing similar degrees of freedom , ranges of motion, link and sizes.

Anthropomorphic hand, on the other hand, often needs complicated joint mechanism such as hinges, gimbals, springs linkages, or gears and belts in order to achieve the right number of DOFs and to mimic the kinematic characteristics of the human hand. The significance of designing the hands likely originates from the expectation of using motorized prosthetic hand to achieve optimal dexterity.

## 1.1 The PARLOMA project



Figure 1.1: PARLOMA

I have worked within PARLOMA, which is a research environment that aims to design and develop a remote communication system between:

- Deaf-blind people
- Deaf person and deaf-blind person
- Deaf-blind person and a person with impaired hearing who has a prior knowledge of a Tactile Sign language

PARLOMA, is a project which provides a novel system to aid communication between deaf-blind people remotely. PARLOMA essentially works as a "telephone" for deaf-blind people. This technology will dramatically improve the quality of life of deaf-blind people. PARLOMA has been presented and supported by the main Italian deaf-blind association, Lega del Filo d'Oro. End users are also involved in the design phase.

### 1.1.1 Tactile sign language

Humans are social beings. It is this trait of "social" that makes them interact and communicate with one another. Communication keeps them bonded with each other, help the fellow beings and also keeps them informed. The art of communication has been an integral part of the human society, the origin of which can never be traced previously.

Being able to communicate was probably the main reason that people came together and gradually a society was formed. However, the crucial link in the society or rather the human race was language. It was the ability to precisely convey thoughts and feelings to one another verbally. But then language was conveyed in an acoustic-vocal way. And more importantly, this way of interaction worked for the "normal" people who had the ability to talk, hear and also see. However, deaf people cannot communicate through this way. Hence, the gestural way comes into play. This means that the communication is codified in poses made by hands using gestures and facial expressions. This is then received through the optical apparatus(or eyes). Sign Languages are similar to vocal languages, in the sense that they have complete grammars and can convey every possible meaning, but at the same time, they are totally independent from the latter. A sign language is a language which typically involves conveying messages using hand gestures along with facial expressions and body movements. In sign language , the speaker, to expresses his/her thoughts may use one hand or both to produce gestures as a means of communication. These gestures, in sign language have specific meaning and are codified and assimilated. Each and every gesture has a specific meaning. Languages of sign use visual gestational channel, so the message is expressed by the body and perceived with sight. It is interesting to note that similar to the fact that any language has variants from region to region in any given country, the sign language also differs from place to place.

Deaf-blind people can use neither vocal nor sign language, in their communication. This is because, by using sign language, they lack a way to receive the communication expressed by the original "signer", as the optical apparatus is absent. Due to this, their communication is different from the conventional sense. The "receiver's" hands are placed on the hands of the "signer" in order to follow the signs made. Since the communication is still based on sign language, this variant is called Tactile Sign Language (t-SL). While it is possible for two "seeing" speakers or two "signers" to communicate in presence of one another or remotely (either through phone calls or video-calling systems), at the moment, there is no way for two deaf-blind people to communicate with each other as the "touch" receiver is absent.

As of now, no solutions exist that enables two (or more) deaf-blind people to communicate remotely. Using tactile Sign Language (t-SL), deaf-blind people can communicate with each other. But for this, a physical presence is essential, as they need to touch the hands of the people they are communicating with, to exchange responses.

The aim of the PARLOMA project is to enable a remote communication, to develop an interface that would be able to reproduce of Tactile Sign Language that would allow the communication to be "normal" and instant for deaf-blind people.



Figure 1.2: Alphabets of the Italian Sign Language

The PARLOMA project, has targeted the Italian Tactile Sign Language, a variant used by Italian deaf-blind people and the ones involved with this community. The aim is to extended this support to other tactile languages spoken in other parts of the world. Italian sign language is like a real language with its structure and syntax. Though it is different from the Italian spoken language, it shares certain similarities with others spoken languages. As illustrated in Fig. 1.2, a variety of different sign combinations positioning the individual fingers and the orientation of the hand in a three dimensional space are depicted. This means that the final design of the robotic hand will have

to try to move in all three degrees of freedom similar to the human arm.

### 1.1.2 The architecture of the system

The communication system used by PARLOMA, is to be composed of different blocks: the "signer" makes his/her signs in front of a sign acquisition and recognition module, which codifies the information before sending it over the internet and to robotic arms. These robotic arm are then able to reproduce the signs that are received. The "receiver" is the able to touch these arms and understand what the "signer" is implying. To realize this system, the team made an analysis on the possible alternatives for each component. Then, after the research activity, the most suitable solution was decided and studied.

The architectural pattern supports the development of Hand Gesture Recognition-based Human Robot Interaction systems leveraging on all the functionalities offered by Robot Operating System (ROS). The pipeline is formed in three blocks:

1. Input Module which consists of acquisition node, for acquiring the frames from the input camera, a tacking node that tracks the hands' motion and recognizes their gestures. This module aims at gathering data from a depth camera and at recognizing gestures performed by the user. Fast and reliable remote transmission is ensured by the ROS framework.
2. Gesture conversion and transmission, wherein the gesture converter node offers the desired abstraction, since it generates specific commands to proper control various and several robotic interfaces.
3. Output Module which is responsible for gesture synthesis through an haptic interface. This module aims at controlling the output robotic and haptic interface(s).

In addition, since ROS provides hardware abstraction for machines executing nodes and it implements a distributed paradigm, remote communication is achieved in a simple manner, and is capable of controlling different robotic interfaces and input cameras, even in cases where multiple I/O devices are connected. The input device is composed of a computer and a Kinect, which is tasked with recognizing the signs of

the tactile sign language. These signs, then appropriately encoded by the gesture conversion module, are transmitted via the internet network to the output node. The output node, in turn consists of anthropomorphic arms that are designed to reproduce the signs in a way such that they are understood by the user.

### 1.1.3 Input Module

The Input Module is in charge of extrapolating and processing in real-time 3D information captured by the input camera to recognize the gesture performed by the user. It consists of three ROS nodes, namely:

- the Hand Tracker node that computes the hand joints' positions from depth images and can be composed of different sub modules depending on the implemented computer vision technique.
- the Gesture/Handshape Classifier node which classifies and recognizes gestures that the "signer" performs in front of the input camera that are sent to this node from the hand tracker node.
- The Camera Driver node reads the stream of images from the input camera and embeds it within a ROS topic. This node supports the video streams coming from a wide range of sensors (e.g, a single RGB camera, a depth camera, a multi-camera system, etc.) and of different types (e.g., 3D information, RGB, grayscale, etc.), even if successive nodes use only depth information for reliability.

As mentioned earlier the input device consists of a Kinect and a computer. The Kinect is a device which is a proprietary of Microsoft technologies, that combines several systems of vision. A normal camera is infringed with an infrared emitter Matrix and a reliable receiver which allows the input node to get a map of the depth of the environment. This kind of Technology was first seen in gaming market as a console controller and then it was used in research as a low-cost and high-tech innovative performance tool. The depth map is provided by Asus Xtion that obtains the gesture recognition performed by the Kinect. These gestures are then processed through complex computer vision algorithms and then translated into packets. The algorithms must be fast and should precisely recognize the gestures made by the "signer". Once it has been recognized, with all the difficulties just illustrated, the computer then deals with the sending packets, optionally coded, via the web to the output device.

#### 1.1.4 Output Module

The Output Module is in charge of reproducing the recognized hand-shape by using the output interface(s). This module is composed by two main ROS nodes, namely the Gesture Converter and the Haptic Interface Driver one. The PARLOMA architecture allows the system to have different output devices connected, enabling one user to simultaneously control many remote interfaces. In fact, several interfaces, if available, can receive the same inputs and perform the same gestures. Nodes are hereby detailed:

- the Gesture Converter node is in charge of converting recognized gestures in generic poses (i.e., intermediate representations that do not target a specific robotic interface). This is achieved through a pre-compiled dictionary of possible poses that associates gestures as received from the network with a corresponding set of commands;
- the Haptic Interface Driver node is devoted to control the robotic hand with specific commands depending upon the employed robotic interface. It receives as input a set of commands from the Gesture Converter node and performs specific algorithms of inverse kinematics and collision avoidance. Collision avoidance is of extreme importance for mission critical tasks, i.e., when there is the necessity that the robotic interface keeps working for a long time without errors and without getting entangled. Behavior of the robotic interface (e.g., allowed movements and working area) is described in a XML file that the driver takes as input and parses to produce proper commands for the given interface. Leveraging on such information, this node discards input stimuli related to tasks and movements that the robotic interface cannot accomplish due to mechanical constraints.

The first task of the output device is to receive packets from the web, process and reproduce the signs using the robotic hand. This task is designated to a small computer, based on ARM, to mount the system operative that is derived from a properly deployed Linux distribution. This Mini computer, Raspberry Pi, is an open Source, that was developed by a nonprofit foundation to make Computer science accessible even in the less industrialized areas of the planet. To pursue this goal, the Raspberry Pi is sold at the cost of about thirty dollars. This performance / price ratio, immediately made the device a top con-

tender for the PARLOMA project. As mentioned above, Raspberry Pi has the task of receiving information from Network and redirect them to an Arduino Mega 2560 r3 which, in turn, has the task to control different actuators and acquire sensors on the hand. Finally, the real anthropomorphic hand, by performing the signs, provides the data to the "receiver" for communication.

## 1.2 Aim of the thesis

In this thesis, we aim to find generalized solutions for the kinematics of 7 DOF anthropomorphic arm. We go in depth into the direct kinematics to find adaptable solution of the manipulator. Then we look at the inverse kinematics and using the fact that the manipulator in question is a redundant arm, we compute the inverse kinematics using redundancy parameterization. We further delve into the mechanics of an anthropomorphic hand and kind the novel solution of the same using the Lagrange method.

### 1.2.1 General findings

Since we will be focusing on an 7 DOF manipulator, redundant manipulators have an infinite number of configurations that lead to the same end-effector desired position and orientation. Forward kinematics is a straightforward problem. Therefore, the existing methods in the literature can be used to solve this problem. Denavit-Hartenberg method is used in this thesis work to derive the forward kinematics.

In the case of Inverse kinematics problems are tricky and indirect to solve. The existing literature on solving the inverse kinematics problems do not provide a general solution, as solving the inverse kinematics problem by analytical methods usually gives exact solutions that are unlike the numerical methods and are faster than the other methods. However, these methods become more complicated to apply to the manipulators that have higher degrees of freedom. Hence we use the redundancy parameterization method.

To simplify the computation of a finger joint, and to keep the fingers as lightweight as possible, we model a generalized robotic finger to find it's differential equation. We achieve this using the Lagrangian

method to find the differential equations. These equations can later be used to find the values of spring constants if need be.

### 1.2.2 **Research outline**

The structure of the thesis work is as following. Chapter 2 reviews the literature of how an anthropomorphic arm has changed over the years and the PARLOMA arm Chapter 3 deals with the overview of the modelling systems and the requirements we have to model our robotic arm and our anthropomorphic hand. Chapter 4 presents the methods of direct kinematics and the method we used in particular for our computations. Chapter , talks about the inverse kinematics, different methods available to solving them, the method used and the results obtained. In chapter 6, we deal with a Robotic fingers. We talk about ways in which we can model the robotic finger and the calculations done to model the finger. Finally, in chapter 7, we conclude the results and talk about the future work.

## Chapter 2

# Literature review

The term Anthropomorphism refers to the capability of a device to mimic the human hand general aspects, such as shape, size, consistency, colour, temperature and aesthetic factors. The word anthropomorphism originates from the Greek words Anthropos and Morphe, meaning human and form respectively. As per Liarokapis et al.[1], we can identify at least two dimensions of similarity between humans and robots: similarity in terms of motion and similarity in terms of morphology. Similarity in motion depends on the kinematic model (relative link lengths and joint positions) and the joint coordination patterns (encapsulated in the concept of synergies from Neuroscience). Similarity in morphology concerns the degree of correspondence for visually perceived characteristics such as shape and size for robotic artifacts or even facial expressions for humanoid robots.

These days anthropomorphism is becoming increasingly important for robotics applications for three main reasons:

- 1) it ensures likeability of robotic artifacts
- 2) increase a robot's capability for motion expressiveness, which can be critical in scenarios where robots must cooperate closely with humans to accomplish a task
- 3) it guarantees safety in Human Robot Interaction (HRI) applications as human-like motion can be intuitively understood by humans.

Humans are starting to interact and co-exist with robots in everyday life environments. Thus, the more human-like a robot is in terms of appearance, perceived intelligence and motion the more easily it will manage to establish solid social connections with humans. In this respect, anthropomorphic arms can generate predictable motion, avoid-

ing to surprise or confuse their human partner, or even generate sub-optimal legible motion that is more intent expressive, thus leading to increased overall safety and efficiency.

## 2.1 History

The robotic arm has become indispensable part of the industry - be it agriculture or automation. One can say that 1495 was a historical year in robotics. It was in this year that the first robotic arm was designed. This newly designed arm provided four degrees of freedom and also included an analog controller. The analog controller did two things, firstly supplied power and secondly provided programmability. The robotic arm from then on slowly and steadily moved towards sophistication and by the year 1993, Arid, the first Android with a robotic arm was seen.

### 2.1.1 Leonardo Da Vinci's Robotic Arm

In the 1950s, researchers at the University of California, examined the comprehensive renderings of Leonardo da Vinci's notebooks which exceeded over 1,119 pages. His work which dated from 1480 to 1518, would thereafter be referred to as the Codex Atlanticus, aptly named after the massive Atlantic Ocean. Leonardo da Vinci, during his lifetime, was heavily influenced by the classical Greek philosophers in the areas of arts and engineering. The investigations stated that, he gained the knowledge from ancient scholars like, Hero of Alexandria<sup>1</sup>, Philon<sup>2</sup> and Ctesibius<sup>3</sup>, who were said to be interested in the mechanical motion and human anatomy. Da Vinci thus began planning and developing these ideas to make a mechanical device. He designed his first "robot" in the December of 1478, when he was just 26 years old (Fig. 2.1).

Folio 812 of the Codex Atlanticus, consists of a front wheel driven, rack and pinion automobile. This was fully programmable and had the ability to control its direction and motion. It has now become

---

<sup>1</sup>Hero of Alexandria (c. 10 AD – c. 70 AD), was a Greek mathematician and engineer

<sup>2</sup>Philon, 4th century BC, well known for planning the architecture of; the portico of twelve Doric columns to the great Hall of the Mysteries at Eleusis and an arsenal at Athens.

<sup>3</sup>Ctesibius(285 BC – 222 BC), was a Greek inventor and mathematician in Alexandria, Ptolemaic Egypt



Figure 2.1: Leonardo da Vinci's automobile, courtesy of Biblioteca Ambrosiana in Milan

apparent that this “base” would eventually become the foundation of his fully functioning automation, according to Rosheim [2]. To make a simulated humanoid, he knew that he had to form a detailed description of human kinesiology. Da Vinci enforced this by learning about drafting, human anatomy, metal construction, learning to make tools, and designing armors on top of painting and sculpting.

In 1495, while he was working on his painting of the Last Supper, it is said that, Da Vinci developed his first programmable robot. The research being done at the Museo Galileo<sup>4</sup> at Florence, it is now known that this mechanical machine had the capability to open and shut its mouth, it could wave its arms and move its head about its neck, automatically according to Capello et al. [3]. This robot consisted of two parts as shown in (Fig. 2.1). The base had three degrees of freedom due to the legs, ankles, knees and hips. The body consisted of four degrees of freedom from the arms with articulated shoulders, elbows, wrists and hands. The positioning of the arms show that moving of the joints in unison could imitate the grasping motion. The robot also had an “on board” controller fitted within its body to provide power and to control the arms. The legs of the robot were simulated by an external crank system. The Museo Galileo has developed computer animations using this design. it is said that Da Vinci re-evaluated this design to impress his former royal benefactor, Francis I of France. From Gian

---

<sup>4</sup>Museo Galileo, the former *Istituto e Museo di Storia della Scienza*, contains a wide range of scientific instruments, which prove the crucial role that the Medici and Lorraine Grand Dukes attached to science and scientists.

Paolo Lomazzo's<sup>5</sup> description of Leonardo, Francesco Melzi<sup>6</sup> describes that Da Vinci, developed many "robots", as writted in his works as "birds, of certain material that few through the air and a lion that could walk...the lion, constructed with marvellous artifice, to walk from its place in a room and then stop, opening its chest, which was full of lilies and different flowers." Rosheim [2] cites that the cart, which were worked by springs, could power the "lion". Da Vinci's multiple degrees of freedom automation is considered as an approximate starting point of technical interest of man in reviewing automation as, da Vinci inspired attention to detail, has been a recurrent theme throughout history.

### 2.1.2 Automata to the Industrial Revolution

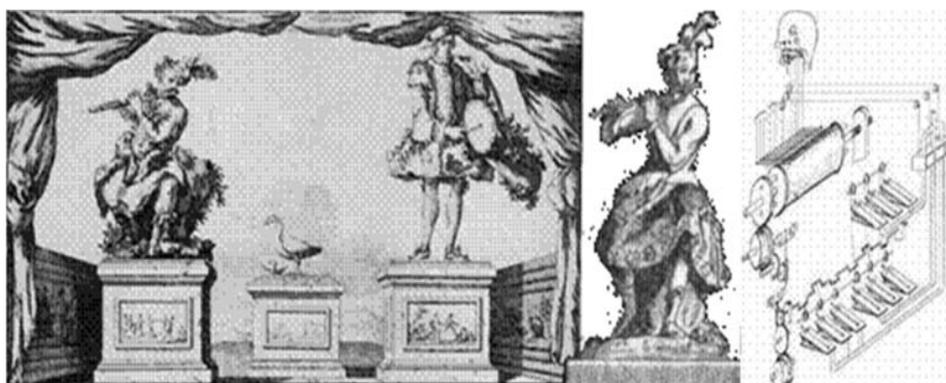


Figure 2.2: De Vaucanson's *Androide*

The son of a glove-maker, might as well, have been responsible for the starting the Industrial Revolutions according to Wood G. [4]. Jacques de Vaucanson, was a mechanical designer who built of some of the most complex, clockwork of eighteenth century. Born in Grenoble in the year 1709, he was the youngest of ten children showing signs of his marvels at an early age. De Vaucanson was interested in the anatomy of the human body, and is said to have attended anatomy and medicine lectures at the Jardin du Roi, where he, met Claude-Nicolas Le Cat<sup>7</sup>. By the year 1738, the De Vaucanson had crafted an

---

<sup>5</sup>Gian Paolo Lomazzo (April 26, 1538 – January 27, 1592), *Trattato dell'arte della pittura, scoltura et architettura*, 1584, RVE.III.37

<sup>6</sup>Francesco Melzi (ca. 1491-1568/70), an Italian painter born into a family of the Milanese nobility in Lombardy.

<sup>7</sup>Claude-Nicolas Le Cat (September 6, 1700 – August 20, 1768) was a French surgeon known for Lithotomy and cataract surgery.

automated flute player (Fig. 2.2), called “*androïde*” and in just over a year, he had made two more automata to add to his exhibition, a drum and pipe player and a duck. According to Riskin J.[5],the duck was his most popular and famous mechanical contrivances. The entrance fee to Vaucanson’s exhibition *Salle des quatre saisons*<sup>8</sup> in Paris, approximately three livres (a week’s pay in those times). Abbe Desfontaines<sup>9</sup>, who was amazed by the human-like features of the “*androïde*”, described the insides of the “*androïde*”,to have contained an “*infinity of wires and steel chains...form the movement of the fingers, in the same way as in living man, by the dilation and contraction of muscles*”. De Vaucanson has written a detailed description of the “*androïde*” that has been published as illustrated works in his book [6].

Many other inventors followed Vaucanson, but the most well-known, were the Swiss clock-making family, called Jaquet-Droz. In 1774, Pierre along with his son Henri Louis, started to make three life size humanoids mimicking human capabilities. the village surgeon is said to have helped them to achieve their goal to develop the limbs and the arms of the automata. They finally succeeded in buiding an artist, a writer and a musician which played a clavichord. This was achieved by applying pressure to the keys by moving the musician’s fingertips (Fig. 2.3).

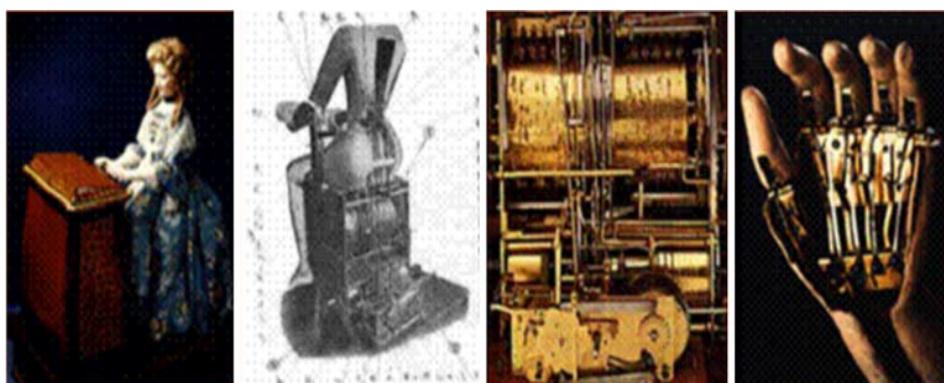


Figure 2.3: Jaquet-Droz’s musician

The final automation of interest is of that of Wolfgang Von Kempelen. He had made a chess player, called the *Turk* (Fig. 2.4), as described in the book by Standage T. [7]. It was developed in 1769 for the Empress Maria Therese. The *Turk* was an elaborate swindle

---

<sup>8</sup>Salle des quatre saisons in *Hôtel de luxe*,Paris

<sup>9</sup>Abbé Pierre François Guyot-Desfontaines (1685 in Rouen - 16 December 1745 in Paris),French journalist, translator and popular historian.

as it was worked by a human operator (Called as the "director", a name given by those who were aware of the automata being controlled by a human). This "director" was concealed inside a complex cabinet present under the chessboard. In spite of this, the automation was an ingenious system of mechanisms. The chess player was carved by wood and sat behind a wooden chest, dressed in Turkish clothing. The *Turk's* head could rotate on its head, its eyes could move in their sockets and the left hand were intricately designed. Kempelen had made a pantograph, an instrument which could enable the "director" to control the *Turk's* left arm from the inside of its body. The limb could be raised and then the hand could be moved over the desired chess piece to be moved. Then, the arm could be lowered to the piece, as the "director" moved a lever inside of the pantograph, making the *Turk's* fingers (each of which had a series of cables connected to the pantograph), grasp the chess piece. The *Turk's* fingers were wooden and during a match, the hand would be placed inside a glove so that it could grasp the chess pieces with more precision.

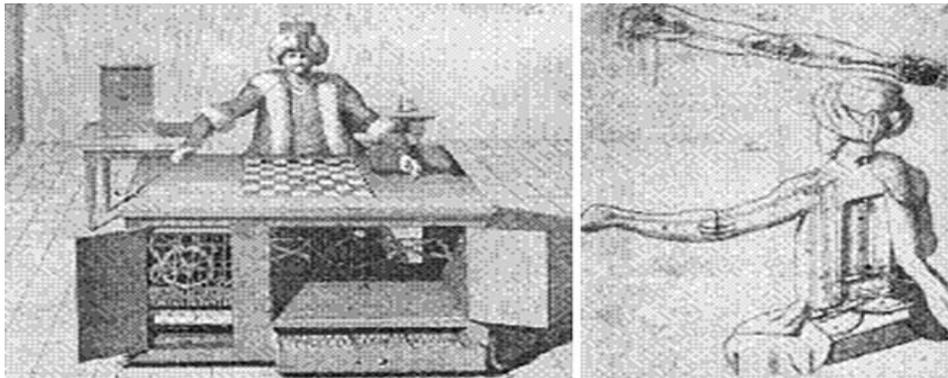


Figure 2.4: Kempelen's *Turk*, with sophisticated left arm mechanisms

### 2.1.3 Robots of Expositions and Industrial Robotic Arms

The first industrial "world fair" took place at Paris in 1798 and it permitted the public to see the progress of technology. With the progress in the field of electronics, where the vacuum tubes were replaced by solid-state transistors, the development in the microelectronics and faster computing systems, the revolution in the modern robotic arms was bound to happen. The first "position controlling apparatus" by Willard Pollard (Fig. 2.5) was patented in 1938. It was a spray painting industrial arm with five degrees of freedom. Although this robotic arm was purely theoretical and never built, Pollard[8]'s design is of interest

as this automated robotic arm would soon motivate others. Harold A. Roselund [9], who worked for DeVilbiss<sup>10</sup>, developed another spray finishing robotic arm, which was manufactured. Both of these arms were ahead of their time as each of them solved the problems related to the joint movements of the arms in unique ways. The control system, however, lacked the reliability inhibiting their popularity.

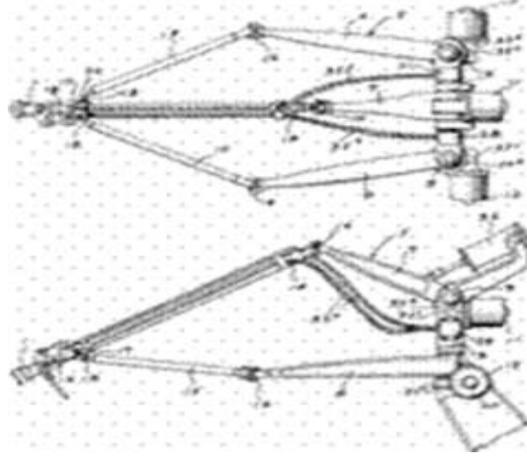


Figure 2.5: Early modern robotic arm-the sketch of Pollard's arm

It was at the San Diego Exposition, that, the "new age" automated humanoid would be exhibited. A little known 2,000-lb "mechanical man" was presented at the expo by its maker, Professor Harry May. *Alpha* (Fig. 2.6), was about 6'2" in height and could open and close its jaws, roll its eyes, sit, stand, could move its arms and could also fire a revolver. In 1939, a superior humanoid was displayed at the New York Expo by the electronics giant, Westinghouse<sup>11</sup>. *Elektro* stood on the stage, moving about with a sliding gait, high above the audience and could supposedly, react to English commands (Fig. 2.6). *Elektro* could perform far more complex tasks than *Alpha* being about 7 feet in height and was made in Mansfield, Ohio [10]. The company records show that, they produced eight more robots from 1931 to 1940 and all these robots had actuated arms and could walk. *Elektro* is said to have had a 78- rpm record player to simulate conversation and had a vocabulary of more than 700 words.

Interestingly, these mechanical humanoids were not called

<sup>10</sup>DeVilbiss Automotive Refinishing(founded in 1907), is an American manufacturer of spray guns, airbrushes and related products for paint and lacquer coating applications.

<sup>11</sup>Westinghouse Electric Corporation(founded on January 8, 1886, as Westinghouse Electric Company) was an American manufacturing company which ceased its operation in 1999

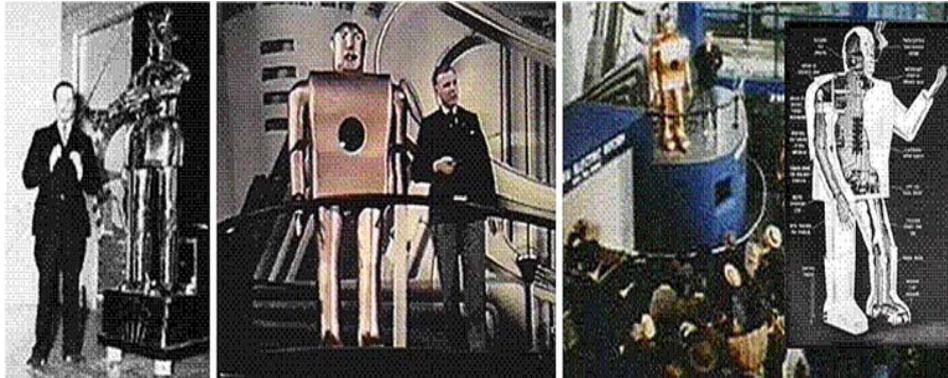


Figure 2.6: The Exposition robots: *far left*, Harry May's Alpha; *right*, three of the Westinghouse *Elektros*

"robots" at that time, as Čapek Kapek's play, "Rossum's Universal Robots"<sup>12</sup> had not gained fame yet. The electronics in those mechanical men was primitive, containing motor drivers that were very loud and they were made up of vacuum tube relays. These parts would be replaced by microelectronics and faster and more efficient mechanisms.

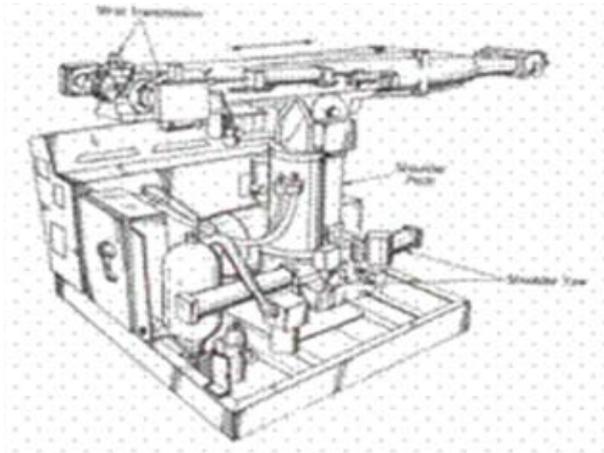


Figure 2.7: Unimate robotic arm

#### Milestones in the robotic industry

1954 — George Devol [11] designed the first programmable robot

1956 — Joseph Engelberger [12], a Columbia University physics student, bought the rights to Devol's robot and started a company called Unimation

1961 — *Unimate* (short for Universal Automation Robot), as shown in

<sup>12</sup>R.U.R. is a 1920 science fiction play by the Czech writer Karel Čapek. R.U.R. stands for Rossumovi Univerzální Roboti

figure 2.7, a robot was installed in Trenton, New Jersey for the first time to tend to a die casting machine

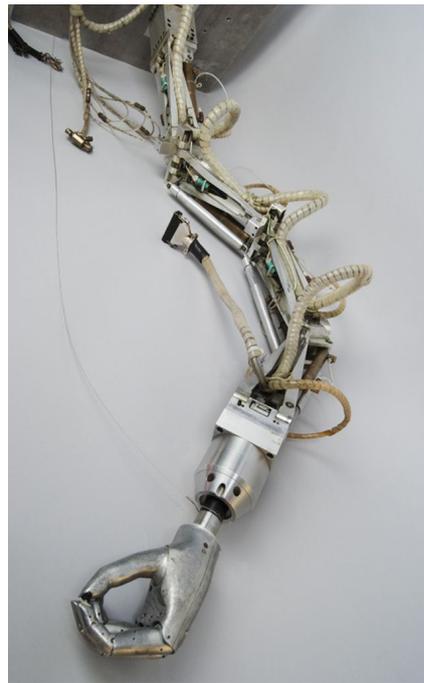
1961 — The force feedback incorporating a robot was developed for the first time

1963 — Robot vision system was established

1966 — Kawasaki<sup>13</sup> bought the license to manufacture industrial robot arms from Unimation.



(a) Stanford Arm



(b) Minsky arm controlled by a PDP-6 computer

Figure 2.8: Sophisticated Robotic Arms

1968 - The Minsky Arm 2.8 was created by Marvin Minsky<sup>14</sup> at Massachusetts Institute of Technology

1971 — The *Stanford arm* 2.8 was created at Stanford University by Victor Scheinman<sup>15</sup>

1973 — Robot programming language called WAVE was developed at

---

<sup>13</sup>Kawasaki Heavy Industries Ltd.(founded on 15th October 1896), is a Japanese public multinational corporation primarily known as a manufacturer of motorcycles, heavy equipment, aerospace and defense equipment, rolling stock and ships.

<sup>14</sup>Marvin Lee Minsky (August 9, 1927 – January 24, 2016) was an American cognitive scientist known for his research in the area of artificial intelligence (AI)

<sup>15</sup>Victor David Scheinman (December 28, 1942 – September 20, 2016), pioneer in the field of robotics, invented the Stanford arm,[1] an all-electric, 6-axis articulated robot designed to permit an arm solution in closed form

Stanford for the first time

1974 — Milacron<sup>16</sup> introduced the robot T3 with computer control

1976 — Draper Labs, develops a device called Remote Center Compliance (RCC) for insertion of parts in assembly was developed at Draper Labs in Boston

1976 — Robot arms were used on the Viking I and II space probes, which later on landed on Mars

1978 — PUMA robot, based on designs from General Motors study, was introduced by Unimation

1979 — SCARA robot design was introduced in Japan

1981 — Carnegie-Mellon University developed the first direct-drive robot

1982 — FANUC<sup>17</sup> and General Motors, formed GM FANUC to market robots in North America

1982 — Westinghouse bought the rights to Unimation, and secured its position as the head of the domestic robot industry

1983 — Adept Technology was founded and which successfully marketed the direct drive robot

1986 — Dr. Robert Barnard, who founded the Woods Hole Oceanographic Institute, introduced the underwater robot, Jason, which explored the wreck of the Titanic

1988 — Stäubli Group<sup>18</sup> purchased Unimation from Westinghouse

1988 — The IEEE Robotics and Automation Society was formed

1993 — ROTEX, an experimental robot, of the German Aerospace Agency (DLR) was flown aboard the space shuttle Columbia which then helped perform a variety of tasks like tele-operated programs and sensor-based offline programs

1996 — Honda unveiled its humanoid robot, the development of which had begun in secret in 1986

1997 — RoboCup-97, the first robot soccer competition was held in Nagoya, Japan and which drew 40 teams from across the world

1997 — Sojourner mobile robot travelled to Mars aboard NASA's Mars PathFinder mission

2000 — Honda<sup>19</sup> created a Humanoid Robot called ASIMO

---

<sup>16</sup>Milacron is an American limited liability company that manufactures and distributes plastic processing equipment for fields such as injection molding, extrusion molding, and metal injection molding

<sup>17</sup>Fuji Automatic Numerical Control,(founded in 1958),provide automation products and services such as robotics and computer numerical control systems.

<sup>18</sup>Stäubli (founded in 1892), is a Swiss Mechatronics company, primarily known for its textile machinery, connectors and robotics products

<sup>19</sup>Honda Motor Co., Ltd. (founded in 1946), is a manufacturer of automobiles, aircraft, motorcycles, and power equipment

2001 — Sony<sup>20</sup> began to mass produce the first household robot, Aibo, which was a robot dog.

2001 — Space Station Remote Manipulation System (SSRMS) was put on the space shuttle, Endeavor, to help facilitate the continued construction of the space station

2001 — Laparoscopic gall bladder removal via the first tele-surgery was performed on a woman in Strasbourg, France by surgeons in New York, USA

2001 — robots were used to search for victims at the World Trade Center site after the September 11th tragedy

2003 — The *Unimate* was inducted into the Robot Hall of Fame



Figure 2.9: Aichi Expo, Japan in 2005 with the robot exhibits

World's Expo of 2005 was held at Aichi, Japan which was attended by over 22,000,000 people. The theme of the expo was “Nature’s Wisdom”, however, technology was the main focus. The humanoid robots played the important role having the tag line “we live in a robot age”. Working robots roved around the expo and participated in the routine chores like garbage collection, sanitation, security, childcare and to aid the handicapped people (Fig. 2.9). Multiple prototypes of robots were displayed for 11 days in the month of June at a “Robot Station”, where visitors could interact with the hosts of those robots, which were basically robot-based industrial manufacturers like Toyota<sup>21</sup>, Mitsubishi<sup>22</sup>, and Brother Industries<sup>23</sup> as they presented their

---

<sup>20</sup>Sony Corporation(founded on 7 May 1946, formerly known as Tokyo Tsushin Kogyo KK), is one of the leading manufacturers of electronics, gaming, entertainment, and financial services

<sup>21</sup>Toyota Motor Corporation(founded on August 28, 1937), is a Japanese automotive manufacturer headquartered in Toyota, Aichi, Japan.

<sup>22</sup>The Mitsubishi Group(founded in 1870), also known as the Mitsubishi Group of Companies or Mitsubishi Companies, is a group of autonomous Japanese multinational companies in a variety of industries.

<sup>23</sup>Brother Industries, Ltd.(founded in 1908, as Yasui Sewing Machine Co.),is a Japanese multinational electronics and electrical equipment company headquartered in Nagoya,

future technology according to Belarmino J. et al. [13].

## 2.2 Anatomy of a human hand

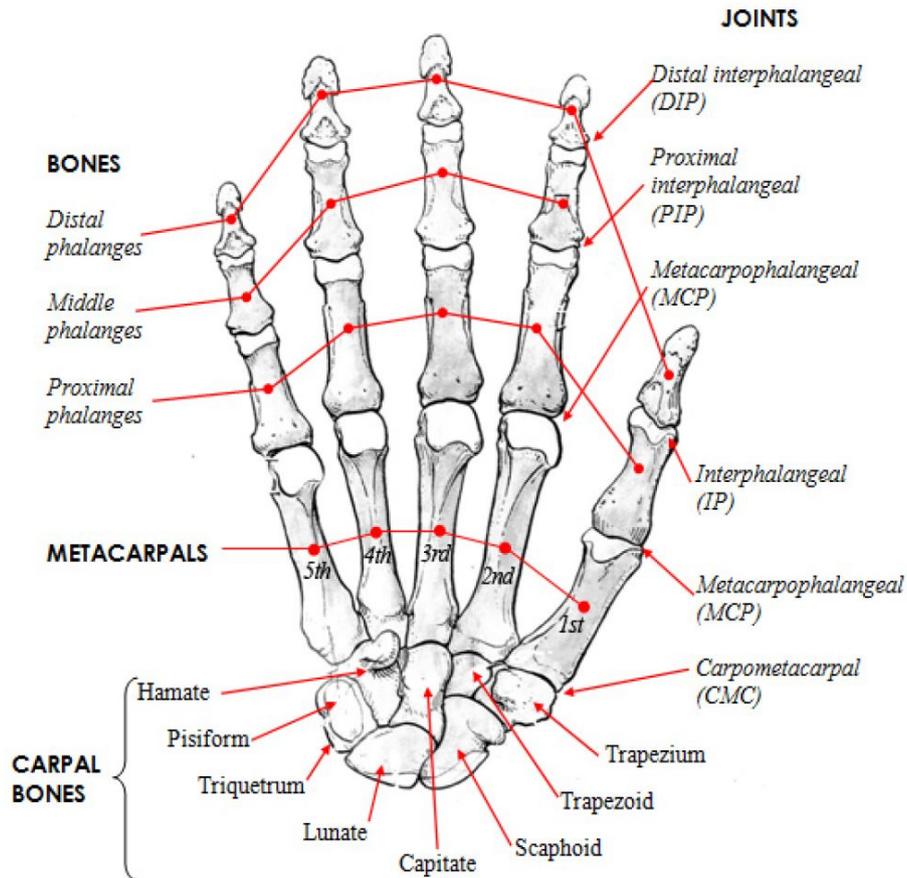


Figure 2.10: Anatomy of a human hand

Understanding the basic anatomy of the hand and fingers is necessary if we want to mimic the functioning of a finger like flexion and extension. There are no muscles in the fingers and fingers move due to the pull of forearm muscles with the help of the tendons. The three bones in each finger are named according to their relationship to the palm of

Japan.

the hand as shown in figure 2.10. The first bone, closest to the palm, is called the proximal phalange; the second bone is the middle phalange; and the bone which is the smallest and the farthest from the hand is called the distal phalange. The thumb does not have a middle phalange. In the same way, the first and largest knuckle is the interlink between the palm and the fingers, which is called the metacarpophalangeal joint (MCP). The next knuckle toward the fingernail is the proximal interphalangeal joint (PIP). The farthest joint of the finger closest to the fingertips is the distal inter-phalangeal joint (DIP).

## 2.3 Progress in the field of robotic hand

A lot of progress has been made in the field of anthropomorphic hand to find innovative ways of mimicking the human hand. Some of these are as follows:

### 2.3.1 EthoHand

There is a dexterous robotic hand called **EthoHand**[20], shown in figure 2.11, comprises of finger joints that are consisting of Ball and socket joint at the thumb.

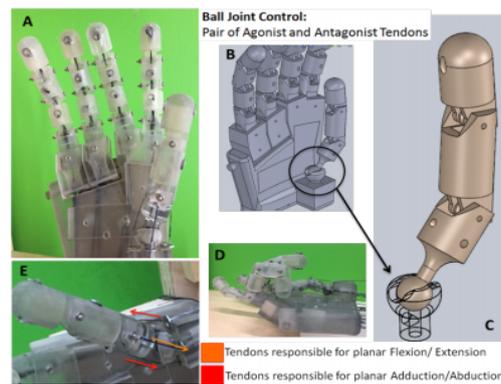


Figure 2.11: EthoHand

### 2.3.2 SmartHand

SmartHand, shown in figure 2.12[21], is a highly innovative, interdisciplinary project, combining forefront research by The BioRobotics Institute, Scuola Superiore Sant'Anna, Pontedera, Italy. It is a five fingered self-contained robotic hand, with 16 degrees of freedom, actuated by 4 motors. It is able to perform everyday grasps, count and independently point the index. Underactuated fingers and differential mechanisms were designed and exploited in order to fit all mechatronic components in the size and weight of a natural human hand. Its sensory system was designed with the aim of delivering significant afferent information to the user through adequate interfaces.

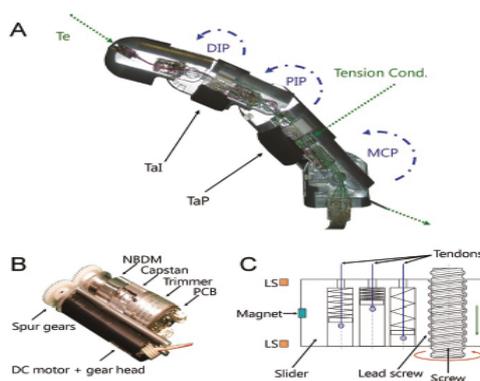


Figure 2.12: SmartHand

### 2.3.3 Spring Hand

Spring Hand, shown in figure 2.13[22], is under actuated, as it has traditional actuators that are replaced with passive elastic elements and mechanical stops. This under actuated hand can be used to obtain an adaptive grasp that resembles human grasping more easily than a hand with completely independent DOFs could achieve. This is similar to the human hand which is also under actuated, as the distal interphalangeal (DIP) joints of the fingers are not independently controllable.

### 2.3. Progress in the field of robotic hand

---

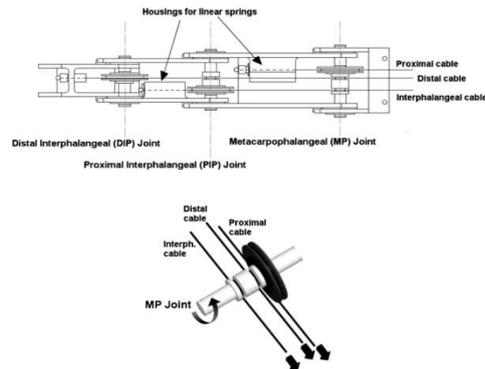


Figure 2.13: SpringHand

#### 2.3.4 Blackfinger

Blackfinger as shown in figure 2.14[23] hand uses a hybrid actuation with cylinders both using air and liquid. The two actuators are used in integration as the pneumatic cylinder is used to set the pressure but difficult to regulate the position of the cylinder and in case of an hydraulic cylinders it is easy to fix the position but difficult to set pressure. Hence the integration of both can offers a good solution. The extensor tendon is connected to the pneumatic single effect cylinder, the flexor tendon to the double effect cylinder, with one chamber filled with high pressure air and the other with low pressure oil. The high pressure of air is regulated by an electro-valve, specially designed.

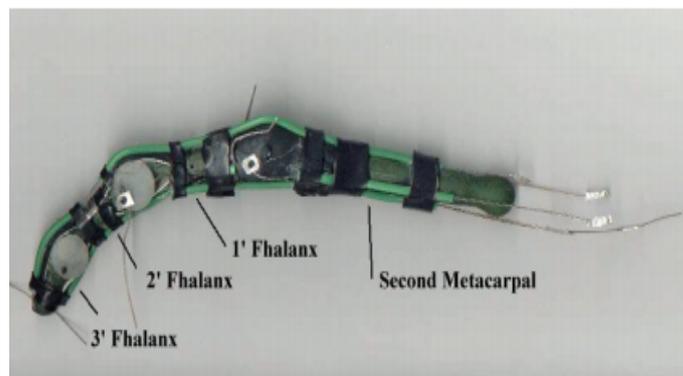


Figure 2.14: Blackfinger

## 2.4 The PARLOMA hand

PARLOMA took inspiration from the InMoov project, which in particular, has produced a novel robotic hand specifically targeting the tactile sign language reproduction. A major concern was also to develop an entirely 3D-printable architecture, to keep costs as low as possible. Secondly, it improved the hand using a spherical parallel three DOF joint as a wrist. Parallel manipulators provided, less inertia and higher stiffness. In addition, to a great architecture, it was also compact and consisted of simple mechanics.

Robotic hands have also been investigated for haptics-based interaction. Haptics can be a complementary communication means for HRI, in addition to vision and hearing. PARLOMA aims to fill this lack by using a haptic interface (a robot hand) that mimics movements of a remote "signer" captured through Computer Vision techniques. Hence, PARLOMA needs a dexterous anthropomorphic robotic hand with a large number of DOFs to replicate the complex movements that are typical of human hands. In addition, such a hand should be low-cost in order to be accessible to a large community of users.

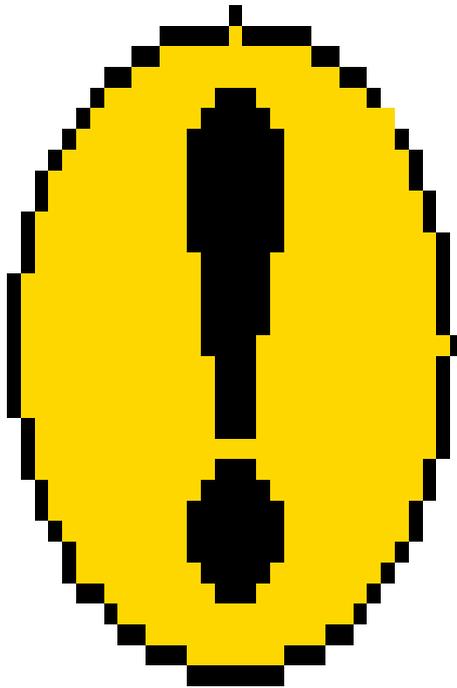


Figure 2.15: PARLOMA hand

The proposed solution consists of a 3D-printable anthropo-

morphic robotic hand designed to reach a high degree of dexterity. With respect to the InMoov's structure, the fingers actuated by the motors were moved and placed in the forearm: bending/extension of each finger is obtained by means of a tendon (for bending) and a spring (extension), leading to three under-actuated joints. The ring and little fingers present an additional bending/extension joint in their base (i.e., contact point with the palm), placed at 45 degrees with respect to the main finger axis to emulate the carpal-metacarpal bending. Abduction/adduction has been implemented for the thumb, index and middle fingers by means of tendons. Motion transmission has been improved by introducing nylon sheaths to reduce friction and provide greater flexibility in the positioning of the actuators, and consequently greater freedom in the design.

In this thesis, we use the hand, shown in figure 2.15. The idea is that this hand would at a later stage would be mounted on a commercial arm. This gives us constraints on the commercial arm like cost, degrees of freedom, weight and payload. We will talk about these in depth in the next chapter. We also look at the possibility of improving the current hand by modelling it's fingers to get optima results.

## Chapter 3

# Overview and requirements of the modelling system

To understand the thesis better, we need to divulge into some basic necessities required for modelling. These representations will be used later on to understand the manipulator in a better way. Herein, we list the factors that form the constraints that we have put on our system to obtain a cheaper and a more robust manipulator. The constraints are applied on both the arm and the hand.

### 3.1 Basics for the modelling system

Kinematics allows to represent positions, velocities and accelerations of specified points in a multi-body structure. These representations can be independent from the causes that may have generated the motion like forces and torques. As we venture into the kinematics of this manipulator, we define some basic concepts which we will be using later on.

#### 3.1.1 End effector and Cartesian/Euclidean space

End effector is a structure which is situated at the end of the last link of the robotic arm, which performs the required task or can hold a tool. The TCP (Tool Center Point) is an ultimate point on the end effector related to which all other robotic points are defined. The TCP

moves in a 3D Cartesian/Euclidean space called Task Space. The task space is the subset of the Cartesian space, that can be reached by the TCP. Joint Space is the mathematical space (vector space) which has elements that are the joint values.

### 3.1.2 Degrees of freedom and redundancy

Joints on a robotic manipulator define the degree of motion (DOM)

$$RobotDOM = n$$

. The number of independent variables that describe the Tool Center Point reference frame is called the TCP degree of freedom (DOF).

$$ToolCenterPointDOF = n' \leq 6$$

. The number of independent variables that characterize or are required by the task reference frame is called the Task DOF.

$$TaskDOF = m \leq 6$$

. We can consider the following cases:

$$n = n' = m$$

$$n = n' > m$$

$$n = n' < m$$

$$n > n' = m$$

Case 1. The robot is called *non-redundant*, it has as many TCP DOF as required by the task.

Case 2. Is impossible for  $m = 6$ , but is possible for  $m \neq 6$ ; in this case the robot is *redundant*.

Case 3. The robot TCP has less DOF than those required by the task, therefore it is a useless robot (for that task).

Case 4. the robot is called *redundant*.

### 3.1.3 Rotation matrix and reference frame

A kinematic chain is a series of ideal arms/links that are connected by ideal joints and consists of a number of:

1. Arms/links (rigid and ideal);
2. Joints (rigid and ideal);

The joints are considered to be of two types:

1. Revolute (or rotational) joints; allows a rotation between two connected links;
2. Prismatic (or translation) joints; allows a translation between two connected links;

Orthogonal Reference Frame (ORF) in 3D space is defined as a set consisting of an origin  $O$  and three mutually orthogonal unit vectors  $(i, j, k)$ :

$$\mathbb{R}(O, i, j, k)$$

Given two right handed orthogonal reference frames  $\mathbb{R}_a$  and  $\mathbb{R}_b$ , with a common origin  $\mathbf{O}$ , but with different orientation, their reciprocal relation can be expressed in two ways:

1.  $\mathbb{R}_b$  represented in  $\mathbb{R}_a$
2.  $\mathbb{R}_a$  represented in  $\mathbb{R}_b$

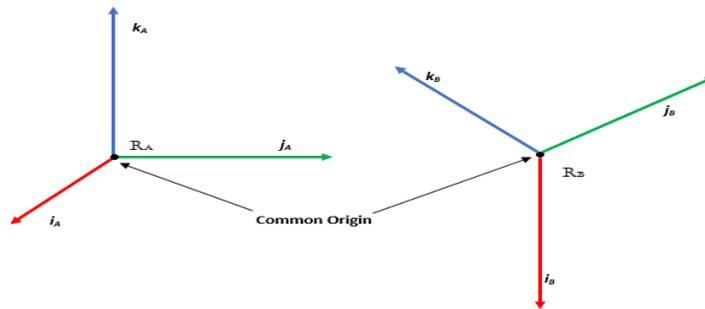


Figure 3.1: Two RF's  $\mathbb{R}_a$  and  $\mathbb{R}_b$  with a common origin but different orientation

$\mathbb{R}_b$  represent the three unit vectors ( $\mathbf{i}_b, \mathbf{j}_b, \mathbf{k}_b$ ) with components in  $\mathbb{R}_a$ :

$$\mathbf{R}_b^a = \left( \begin{array}{c} \left[ \begin{array}{c} \mathbf{i}_b \\ \mathbf{j}_b \\ \mathbf{k}_b \end{array} \right]_a \end{array} \right) \quad (3.1)$$

$\mathbb{R}_a$  represent the three unit vectors ( $\mathbf{i}_a, \mathbf{j}_a, \mathbf{k}_a$ ) with components in  $\mathbb{R}_b$ :

$$\mathbf{R}_a^b = \left( \begin{array}{c} \left[ \begin{array}{c} \mathbf{i}_a \\ \mathbf{j}_a \\ \mathbf{k}_a \end{array} \right]_b \end{array} \right) \quad (3.2)$$

These are rotation matrices that satisfy the relations:

$$\begin{aligned} R_a^b &= (R_b^a)^T \\ R_b^a &= (R_a^b)^T \end{aligned}$$

We use elementary rotation matrices where, there are rotations around the unit vectors ( $i, j, k$ ) of a generic Reference Frame:

$$Rot(x, \alpha) = Rot(i, \alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \quad (3.3)$$

$$Rot(x, \alpha) = Rot(i, \alpha) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} = \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \quad (3.4)$$

$$Rot(x, \alpha) = Rot(i, \alpha) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c\gamma & -s\gamma & 0 \\ s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

Every rotation can be obtained combining these three elementary rotation matrices, to obtain rotations, that are composed of a number of elementary rotations. To achieve this, it is necessary to:

- define the elementary rotation which is applied, i.e., define the rotation axis;
- define the rotation angle;
- establish if the rotation is performed around a fixed reference frame or a mobile reference frame;
- if the rotation takes place with respect to the fixed reference frame, then pre-multiply the matrix with the previous rotation matrix.
- if the rotation takes place with regard to the mobile RF, then post-multiply rotation matrix with the previous rotation matrix;

### 3.1.4 Homogeneous Transformation matrix

Translations are rigid displacements, that keep the axes of the new reference frame, parallel to the old axes, and just translating the origin.

$$\mathbf{T}_b^a : \text{Re}_a(O, i, j, k) \rightarrow \text{Re}_b(O', i, j, k) \quad (3.6)$$

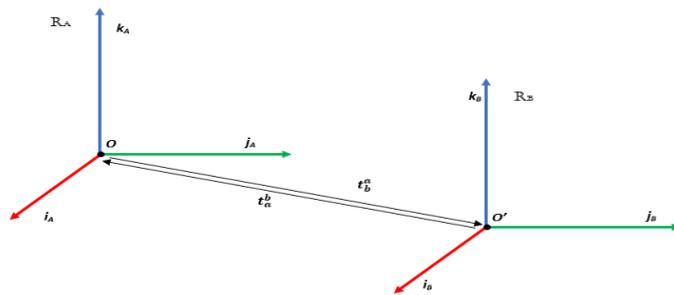


Figure 3.2: Translation with respect to a frame

where  $\overrightarrow{OO'} = t_b^a$  is the translation vector represented in  $\mathbb{R}_a$

Rotations are represented by matrices and translations of vectors by using the homogeneous representation of vectors:

$$v = \begin{bmatrix} p1 \\ p2 \\ p3 \end{bmatrix} \rightarrow \bar{v} = \begin{bmatrix} p1 \\ p2 \\ p3 \\ 1 \end{bmatrix} \quad (3.7)$$

It is possible to use a unique representation for roto-translations, called homogeneous transformation matrix.

$$\mathbf{T}_b^a = \begin{bmatrix} \mathbf{R}_b^a & \mathbf{t}_b^a \\ 0^t & 1 \end{bmatrix} \quad (3.8)$$

where  $\mathbf{R}_b^a$  is the rotation and  $\mathbf{t}_b^a$  the translation and

$$0^t = [0 \ 0 \ 0]$$

## 3.2 Requirements of Robotic Arm

To find the best robots available which would efficiently replicate the movements of a human arm, led me to research the possible articulated robotic arms in the commercial sector today. Having analyzed the requirements posed by PARLOMA, we notice that, any robotic arm, to fit in our constraints would require:

1. The arm to be light weight: This is so that the final product would be portable if need be.
2. The arm should be capable of having a payload of 3 to 10kg: This is because though the arm would be fitted with servo motors, springs and other actuating devices which would eventually add up to the final weight. Another reason is that when "receiver" comprehends the tactile signs, by touching the arms, he/she would exert a certain amount of their body weight on to the arm.
3. A manipulator with higher DOF is preferred: We know that human arm consists of 7 degrees of freedom and since we are trying to replicate the same, the manipulator chosen should be as close

to the human arm as possible as higher DOF means higher dexterity.

Keeping the above constraints in mind, we made the following observations:

The table shows different arms with various degrees of freedom, payloads and weights. This would eventually lead to us making a trade-off so as to prioritize our constraints. Since we put more importance on the Degrees of freedom, we see that Yaskawa SIA5F manipulator has 7 degrees of freedom. Looking at its other specifications like weight and payload, we see that it lies well within our specified constraints. Hence we considered this manipulator for our further calculations.

### 3.3 Requirements for the robotic finger

As seen in the last chapter, we need to improve up on the current PARLOMA hand, to achieve a low cost hand with better dexterity. To achieve this, we need to look into the spring of our under actuated hand. The making of our spring should be such that it is of a low cost, as customized springs are more expensive to be manufactured. We look at the following constraints:

1. The hand should be bio-mimetic : we see that the current arm in question already satisfies the requirements for this
2. Low cost : we see that the cost goes up when there is a need to manufacture customized springs. If we are able to define the constraints on the spring, we can save up on additional costs.
3. Light weight : Though the current hand satisfies this condition moderately, we see that by reducing the elements on the arm will lead to lesser weight.

We will further apply the constraints on the arm and the hand in chapter 4, 5 and 6 respectively.

Table 3.1: Comparison between robotic manipulators by manufacturers

Manufacturer	Model	DOF	Max. Payload(Kg)	Weight(Kg)
Staubli	TX60	6	9	51
	TX60L	6	5	52
Universal Robots	UR5	6	5	18.4
	UR3	6	3	11
	UR10	6	10	28.9
Denso	VS-S2 H2O2/UV SERIES	6	4	34
	NEW VS SERIES - NEW VS-050	6	4	29
	NEW VS SERIES - NEW VS-060	6	4	30
Yaskawa	MH3BM	6	3	27
	MH3F	6	3	27
	MH5LS II	6	5	29
	GP7	6	7	34
	GP8	6	8	32
	SIA5F	7	5	30
Kawasaki Robotics	RS003N	6	3	30
	RS005N	6	5	34
	RS005L	6	5	37
	RC005I	6	5	37

## Chapter 4

# Direct Kinematics

Kinematics describes the motion of points, bodies and systems of bodies (like a manipulator) without considerations on the causes of motion. In particular, robot kinematics studies the mathematical relationships between positions, velocities and accelerations of specific points of these bodies. The study of these relationships allows to define some functions, called kinematic functions.

This chapter discusses and develops the kinematic equations of a 7-DOF articulated robot manipulator. The manipulator considered in this project has seven revolute joints that rotate around fixed axes of their previous links. A 7-DOF manipulator is considered as it has a similar structure to that of a human arm. The seven revolute joints are so arranged, so as to form the shoulder, elbow, and wrist as shown in Fig. 4.1. Direct kinematics problem is straight forward issue as compared to the inverse kinematics. Denavit-Hartenberg method is the most common method for describing the manipulator kinematics to derive the direct kinematic equations.

The direct kinematic functions form the direct kinematics which transform the variables  $\mathbf{q}(t)$ <sup>1</sup> and  $\dot{\mathbf{q}}(t)$ <sup>2</sup> of joint space into the variables  $\mathbf{p}(t)$  and  $\dot{\mathbf{p}}(t)$  of the task space. In opposite way, the inverse kinematic functions form the inverse kinematics, which transform the variables  $\mathbf{p}(t)$  and  $\dot{\mathbf{p}}(t)$  of the task space into the variables  $\mathbf{q}(t)$  and  $\dot{\mathbf{q}}(t)$  of joint space.

Assigned two reference systems  $\mathbf{R}_0$  and  $\mathbf{R}_{TCP}$ , the base and

---

<sup>1</sup> bold letter represents a vector or a matrix

<sup>2</sup>the dot above the letter represents first derivative

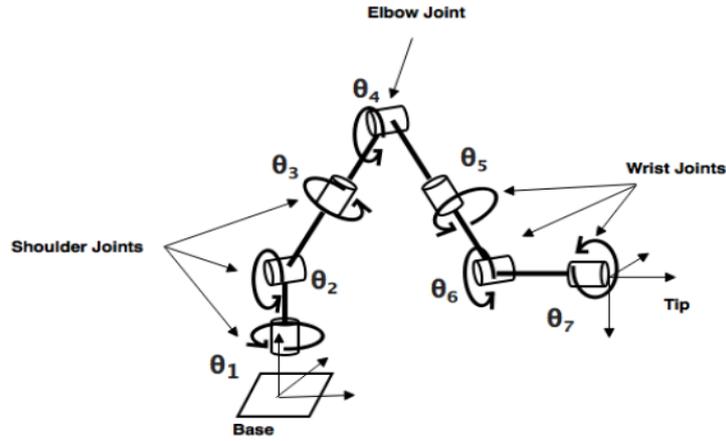


Figure 4.1: 7 DOF manipulator

the manipulator's tool centre point, respectively, it is possible to define the vector  $\mathbf{p}(t)$  for position and  $\mathbf{R}_{TCP}$  for orientation, represented in the reference  $\mathbf{R}_0$ .

$$p(t) = \begin{bmatrix} \mathbf{x}(t) \\ \boldsymbol{\alpha}(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \alpha_1(t) \\ \alpha_2(t) \\ \alpha_3(t) \end{bmatrix}$$

The vector  $\mathbf{q}(t)$ , instead, is the vector of joint variables

$$\mathbf{q}(t) = \begin{bmatrix} q_1(t) \\ \vdots \\ q_n(t) \end{bmatrix}$$

where  $n$ , is the number of joints.  
Similarly,  $\dot{\mathbf{p}}(t)$ , is the vector of Cartesian velocities

---


$$\dot{\mathbf{p}}(t) = \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\boldsymbol{\alpha}}(t) \end{bmatrix} = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \\ \dot{\alpha}_1(t) \\ \dot{\alpha}_2(t) \\ \dot{\alpha}_3(t) \end{bmatrix}$$

Here  $\dot{\mathbf{x}}(t)$  is a vector of linear velocities and  $\dot{\boldsymbol{\alpha}}(t)$  is a vector of angular velocities of  $R_{TCP}$  represented in the reference  $R_0$ . Similarly  $\dot{\mathbf{q}}(t)$  is the vector of joints velocities

$$\dot{\mathbf{q}}(t) = \begin{bmatrix} \dot{q}_1(t) \\ \vdots \\ \dot{q}_n(t) \end{bmatrix}$$

For the simplicity of this thesis, we will concentrate on just the Direct position kinematics, which refers to the use of the kinematic equations of a robot to compute the position of the end-effector from specified values for the joint parameters.

It, mathematically, works to calculate the transformation matrix (as seen in section 3.1.4)  $\mathbf{f}(\mathbf{q}(t))$  (with which the vector  $\mathbf{p}(t)$  is built), such that  $\mathbf{p}(t) = \mathbf{f}(\mathbf{q}(t))$ .

$$\mathbf{p}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \boldsymbol{\alpha}(t) \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{q}) \\ \vdots \\ f_n(\mathbf{q}(t)) \end{bmatrix}$$

$\mathbf{f}(\mathbf{q}(t))$  is generally pointed out with  $\mathbf{T}_n^0$ , where the reference system  $n$  is represented in the reference system 0.  $\mathbf{T}_n^0$  is gotten as composition of the  $n$  elementary matrices  $\mathbf{T}_i^{i-1}(q_i(t))$ , with  $i=1, \dots, n$ .

Form of the transformation matrix  $\mathbf{T}_n^0(\mathbf{q}(t))$  is shown below

$$\mathbf{T}_n^0 = \mathbf{T}_1^0 \cdot \mathbf{T}_2^1 \cdot \dots \cdot \mathbf{T}_n^{n-1} = \begin{bmatrix} \mathbf{R}_n^0 & \mathbf{t}_n^0 \\ \mathbf{0}^T & 1 \end{bmatrix}$$

## 4.1 Methods available

### 4.1.1 Denavit Hartenberg convention

In 1955, Jacques Denavit and Richard Hartenberg introduced this convention [24] in order to standardize the coordinate frames for spatial linkages. The Denavit-Hartenberg convention is a particular convention attach reference frames to the links of a spatial kinematic chain, or robot manipulator, which defines four parameters with which to build the matrix  $\mathbf{T}_n^0$ . the following rules are used in the Denavit Hartenberg convention:

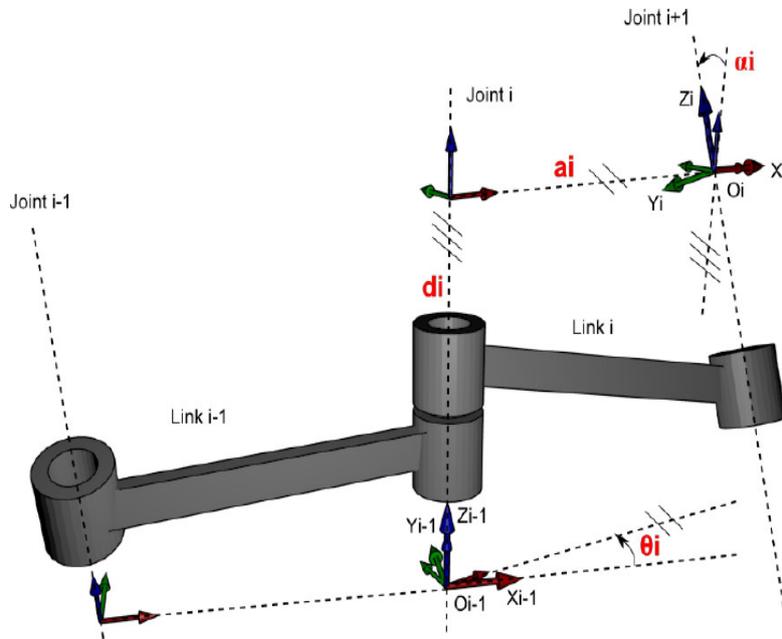


Figure 4.2: Denavit-Hartenberg convention

1. The origin of  $R_i$  is located on the axis of joint  $g_{i+1}$  at the intersection between this axis and the common normal between this motion axis  $g_{i+1}$  and the previous motion axis  $g_i$ . If the two axes intersect, the origin is located in the intersection point. If axes are parallel, the origin can be located in a point of choice, usually on the arm;
2. Versor  $k_i$  is aligned with the motion axis  $g_{i+1}$  with the verse indicating positive motion;

h!

Table 4.1: Denavit Hartenberg parameters

Parameters	Description
$d_i$	it defines the translation along the motion axis $k_{i-1}$ , between the origin of $R_{i-1}$ and the intersection of the axis defined by $k_{i-1}$ and the axis defined by $i_i$ .
$\theta_i$	it defines the rotation around axis $k_{i-1}$ , such that $i_{i-1}$ overlaps $i_i$ .
$a_i$	it defines the minimum signed distance between axis $k_{i-1}$ and $k_i$ along the common normal, measured along $i_i$
$\alpha_i$	it defines the rotation angle around motion axis $i_i$ such that $k_{i-1}$ overlaps $k_i$ .

3. Versor  $i_i$  is set orthogonal to  $k_{i-1}$ . Since  $i_i$  must be also orthogonal to  $k_i$ , it is normal to the plane defined by  $k_{i-1}$  and  $k_i$ . If those two vectors are parallel, axis  $i_i$  lies on the plane normal to them with direction and verse chosen by the user;
4. Finally, versor  $j_i$  completes the right-hand RF;

When fixing the reference frame  $R_0$  of the base, the origin cannot be uni-vocally defined. One solves the ambiguity setting only the direction of  $k_0$ , so that it is aligned with  $g_1$  and the origin, the verse of  $k_0$ ,  $i_0$  and  $j_0$  are chosen by the user. Moreover when fixing reference frame on the TCP, versor  $i_i$  is normal to  $k_{i-1}$  and the remaining versors are chosen by the user.

So the DH homogeneous rotational-translation matrix defines the transformation between two consequently reference frame.

$$\mathbf{T} = \mathbf{T}(\mathbf{R}, \mathbf{t}) = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0^T & 1 \end{bmatrix} \quad (4.1)$$

$$\mathbf{T}_i^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

$$\mathbf{T}_i^{i-1} = \mathbf{T}(I, d)\mathbf{T}(R_{k,\theta}, 0)\mathbf{T}(I, \alpha)\mathbf{T}(R_{i,\alpha}, 0) \quad (4.3)$$

Finally we can compute the Direct kinematics function with these step:

1. Select and identify links and joints;
2. Define and place the RFs using DH conventions;
3. Define the constant DH parameters;
4. Define the variable DH parameters and the generalized coordinate;
5. Compute the homogeneous transformation  $\mathbf{T}_i^{i-1}$  and  $\mathbf{T}_{TCP}^0$ ;
6. Extract the direct position KF from  $\mathbf{T}_{TCP}^0$

#### 4.1.2 Neural Network analysis

One of the features of the neural networks[14] is it's ability to approximate nonlinear maps or functions. Furthermore neural network schemes are independent of the system structure resulting in a robust approach with respect to environmental changes. Neural networks are said to be the ideal choice to compute the direct kinematics of parallel manipulators. Two most popular neural network models are the Multi-layer Feed Forward and Radial Basis networks.

#### 4.1.3 Quasi-closed solution method

In this method of analytical solution[15] of the direct kinematics, numerical approximation is used as far as possible to solve the equations. By this means, we basically start with 9 unknown parameters,

then using analytical manipulation, the problem is reduced to determine 2 further parameters. Once we find these two parameters, others can easily be calculated.

#### 4.1.4 Taylor series expansion

This direct kinematic model[15] uses nonlinear equations with actuator lengths as the input and orientation angles of the end effector as the outputs . The number of the coefficients in the expansion is determined by the required degree of accuracy. Solving the direct kinematics problem, will hence be equal to computing these coefficients.

## 4.2 Method applied in the case of a 7 DOF robotic arm

In the case of direct kinematics, we face no complexity in terms of deriving the equations, as it is a straightforward problem. So, we use the DH parameter convention.

## 4.3 Results achieved

Direct kinematics equations are obtained by assigning the coordinate frames and establishing the link parameters  $d_i$ ,  $a_i$ ,  $\alpha_i$  and  $\theta_i$  that form the Denavit Hartenberg table. The link length  $a_i$  is distance along  $x_i$  from the origin  $o_i$  to the intersection of the axis of  $x_i$  and the axis of  $z_{i-1}$ . With the help of the assigned frames, shown in Figure 4.3, we see that there is no distance along any x axes from the origins to the point of intersection between the  $x_i$  and  $z_{i-1}$  so that the link length is zero in all the frames in the D-H table.

The link offset  $d_i$  is the distance along the axis  $z_{i-1}$  from the origin  $o_{i-1}$  to the intersection of the axis  $x_i$  and the axis of  $z_{i-1}$ . The assigned frames in Figure4.3 shows that there are  $l_s^b$  along  $z_0$ ,  $l_e^s$  along  $z_2$ ,  $l_w^e$  along  $z_4$  and  $l_t^w$  along  $z_6$  while there is no any link offsets along  $z_1$ ,  $z_3$  and  $z_5$ . The link twist  $\alpha_i$  is the angle between  $z_{i-1}$  and  $z_i$  and is measured about the  $x_i$  axis. Figure 4.3 shows, about  $x_1$ , there is an angle of  $-\pi/2$ (rad) between the axes of  $z_0$  and  $z_1$ , there is a same angle

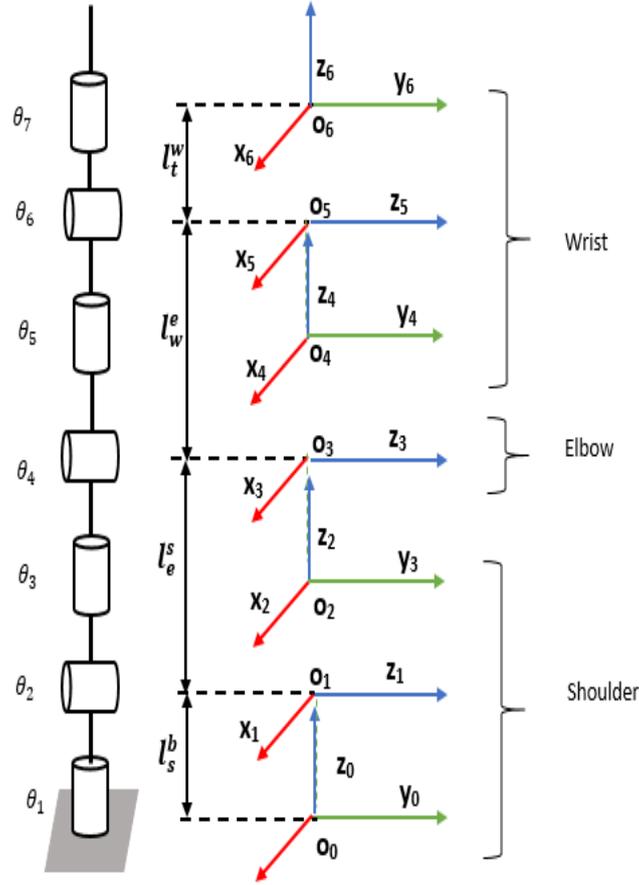


Figure 4.3: The coordinate frames assigned to the manipulator

is around the axes of  $x_3$  and  $x_5$ . The angle about  $x_2$ , between the axes of  $z_1$  and  $z_2$  is  $\pi/2$ (rad), which is the same angle around the axes of  $x_4$  and  $x_6$ . The figure also illustrates that there is no angle around  $x_7$ , that is, between the axes of  $z_6$  and  $z_7$ .

The Link angle  $\theta_i$  is the angle between  $x_{i-1}$  and  $x_i$ , and is measured about  $z_{i-1}$  axis. Since all the joints are revolute joints, the joint angles from  $\theta_1$  through  $\theta_7$  are all variables. The Denavit-Hartenberg parameters are shown in Table 4.2.

The homogeneous transformation matrices  $\mathbf{T}_{i+1}^i$  are calculated by substituting the above parameters in the table 4.2, into the matrix equation for each joint. The resulting transformation matrices are given by equation (4.4). For the simplifications of the notations,

4.3. Results achieved

Table 4.2: Comparison between robotic manipulators by manufacturers

Joint	Link angle $\theta_i$ (Rad)	Link twist $\alpha_i$ (rad)	Link length $a_i$ (mm)	Link offset $d_i$ (m)
1	$\theta_1$	$-\frac{\pi}{2}$	0	$l_s^b$
2	$\theta_2$	$\frac{\pi}{2}$	0	0
3	$\theta_3$	$-\frac{\pi}{2}$	0	$l_e^s$
4	$\theta_4$	$\frac{\pi}{2}$	0	0
5	$\theta_5$	$-\frac{\pi}{2}$	0	$l_w^e$
6	$\theta_6$	$\frac{\pi}{2}$	0	0
7	$\theta_7$	0	0	$l_t^w$

the  $\cos(\theta_i)$  written as  $C_i$  and  $\sin(\theta_i)$  written as  $S_i$ .

$$\begin{aligned}
 \mathbf{T}_1^0 &= \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & l_s^b \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{T}_2^1 &= \begin{bmatrix} C_2 & 0 & S_2 & 0 \\ S_2 & 0 & -C_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 \mathbf{T}_3^2 &= \begin{bmatrix} C_3 & 0 & -S_3 & 0 \\ S_3 & 0 & C_3 & 0 \\ 0 & -1 & 0 & l_e^s \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{T}_4^3 &= \begin{bmatrix} C_4 & 0 & S_4 & 0 \\ S_4 & 0 & -C_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 \mathbf{T}_5^4 &= \begin{bmatrix} C_5 & 0 & -S_5 & 0 \\ S_5 & 0 & C_5 & 0 \\ 0 & -1 & 0 & l_w^e \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{T}_6^5 &= \begin{bmatrix} C_6 & 0 & S_6 & 0 \\ S_6 & 0 & -C_6 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 \mathbf{T}_7^6 &= \begin{bmatrix} C_7 & -S_7 & 0 & 0 \\ S_7 & C_7 & 0 & 0 \\ 0 & 0 & 1 & l_t^w \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{4.4}$$

The multiplication of the equations shown in (4.4) are performed using Symbolic Math Toolbox in MATLAB to obtain a symbolic form of the direct kinematic function  $\mathbf{T}_7^6$ . This gives the position and orientation of the end-effector as a function of the joint angles  $\theta_i$ . The position and orientation transformation matrix of the end-effector

relative to basic coordinate system is shown in equation (4.5).

$$\mathbf{T}_7^0 = \mathbf{T}_1^0 \cdot \mathbf{T}_2^1 \cdot \mathbf{T}_3^2 \cdot \mathbf{T}_4^3 \cdot \mathbf{T}_5^4 \cdot \mathbf{T}_6^5 \cdot \mathbf{T}_7^6 = \begin{bmatrix} R_{11} & R_{12} & R_{13} & p_x \\ R_{21} & R_{22} & R_{23} & p_y \\ R_{31} & R_{32} & R_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.5)$$

Here  $R_{11}$ ,  $R_{12}$ ,  $R_{13}$ ,  $R_{21}$ ,  $R_{22}$ ,  $R_{23}$ ,  $R_{31}$ ,  $R_{32}$ ,  $R_{33}$ ,  $p_x$ ,  $p_y$  and  $p_z$  are given by the equations from (4.6) through (4.17) respectively.

$$\begin{aligned} R_{11} = & C_7 * (S_6 * (S_4 * (S_1 * S_3 - C_1 * C_2 * C_3) \\ & - C_1 * C_4 * S_2) - C_6 * (S_5 * (C_3 * S_1 + C_1 * C_2 * S_3) \\ & + C_5 * (C_4 * (S_1 * S_3 - C_1 * C_2 * C_3) + C_1 * S_2 * S_4))) \\ & - S_7 * (C_5 * (C_3 * S_1 + C_1 * C_2 * S_3) - S_5 * (C_4 * (S_1 * S_3 \\ & - C_1 * C_2 * C_3) + C_1 * S_2 * S_4)) \end{aligned} \quad (4.6)$$

$$\begin{aligned} R_{12} = & -S_7 * (S_6 * (S_4 * (S_1 * S_3 - C_1 * C_2 * C_3) \\ & - C_1 * C_4 * S_2) - C_6 * (S_5 * (C_3 * S_1 + C_1 * C_2 * S_3) \\ & + C_5 * (C_4 * (S_1 * S_3 - C_1 * C_2 * C_3) + C_1 * S_2 * S_4))) \\ & - C_7 * (C_5 * (C_3 * S_1 + C_1 * C_2 * S_3) - S_5 * (C_4 * (S_1 * S_3 \\ & - C_1 * C_2 * C_3) + C_1 * S_2 * S_4)) \end{aligned} \quad (4.7)$$

$$\begin{aligned} R_{13} = & -C_6 * (S_4 * (S_1 * S_3 - C_1 * C_2 * C_3) - C_1 * C_4 * S_2) \\ & - S_6 * (S_5 * (C_3 * S_1 + C_1 * C_2 * S_3) + C_5 * (C_4 * (S_1 * S_3 \\ & - C_1 * C_2 * C_3) + C_1 * S_2 * S_4)) \end{aligned} \quad (4.8)$$

$$\begin{aligned}
 R_{21} = & S_7 * (C_5 * (C_1 * C_3 - C_2 * S_1 * S_3) - S_5 * (C_4 * (C_1 * S_3 \\
 & + C_2 * C_3 * S_1) - S_1 * S_2 * S_4)) - C_7 * (S_6 * (S_4 * \\
 & (C_1 * S_3 + C_2 * C_3 * S_1) + C_4 * S_1 * S_2) - C_6 * (S_5 * (C_1 * C_3 \\
 & - C_2 * S_1 * S_3) + C_5 * (C_4 * (C_1 * S_3 + C_2 * C_3 * S_1) \\
 & - S_1 * S_2 * S_4)))
 \end{aligned} \tag{4.9}$$

$$\begin{aligned}
 R_{22} = & S_7 * (S_6 * (S_4 * (C_1 * S_3 + C_2 * C_3 * S_1) + C_4 * S_1 * S_2) \\
 & - C_6 * (S_5 * (C_1 * C_3 - C_2 * S_1 * S_3) + C_5 * (C_4 * (C_1 * S_3 \\
 & + C_2 * C_3 * S_1) - S_1 * S_2 * S_4))) + C_7 * (C_5 * (C_1 * C_3 \\
 & - C_2 * S_1 * S_3) - S_5 * (C_4 * (C_1 * S_3 + C_2 * C_3 * S_1) \\
 & - S_1 * S_2 * S_4))
 \end{aligned} \tag{4.10}$$

$$\begin{aligned}
 R_{23} = & C_6 * (S_4 * (C_1 * S_3 + C_2 * C_3 * S_1) \\
 & + C_4 * S_1 * S_2) + S_6 * (S_5 * (C_1 * C_3 - C_2 * S_1 * S_3) \\
 & + C_5 * (C_4 * (C_1 * S_3 + C_2 * C_3 * S_1) - S_1 * S_2 * S_4))
 \end{aligned} \tag{4.11}$$

$$\begin{aligned}
 R_{31} = & S_7 * (S_5 * (C_2 * S_4 + C_3 * C_4 * S_2) + C_5 * S_2 * S_3) \\
 & - C_7 * (S_6 * (C_2 * C_4 - C_3 * S_2 * S_4) + C_6 * (C_5 * \\
 & (C_2 * S_4 + C_3 * C_4 * S_2) - S_2 * S_3 * S_5))
 \end{aligned} \tag{4.12}$$

$$\begin{aligned}
 R_{32} = & C_7 * (S_5 * (C_2 * S_4 + C_3 * C_4 * S_2) \\
 & + C_5 * S_2 * S_3) + S_7 * (S_6 * (C_2 * C_4 - C_3 * S_2 * S_4) \\
 & + C_6 * (C_5 * (C_2 * S_4 + C_3 * C_4 * S_2) - S_2 * S_3 * S_5))
 \end{aligned} \tag{4.13}$$

$$\begin{aligned}
 R_{32} = & C_6 * (C_2 * C_4 - C_3 * S_2 * S_4) \\
 & - S_6 * (C_5 * (C_2 * S_4 + C_3 * C_4 * S_2) - S_2 * S_3 * S_5)
 \end{aligned} \tag{4.14}$$

$$\begin{aligned}
 p_x = & C_1 * S_2 * l_e^s - l_w^e * (S_4 * (S_1 * S_3 \\
 & - C_1 * C_2 * C_3) - C_1 * C_4 * S_2) - l_t^w * (C_6 * \\
 & (S_4 * (S_1 * S_3 - C_1 * C_2 * C_3) - C_1 * C_4 * S_2) \\
 & + S_6 * (S_5 * (C_3 * S_1 + C_1 * C_2 * S_3) \\
 & + C_5 * (C_4 * (S_1 * S_3 - C_1 * C_2 * C_3) + C_1 * S_2 * S_4))
 \end{aligned} \tag{4.15}$$

$$\begin{aligned}
 p_y = & l_t^w * (C_6 * (S_4 * (C_1 * S_3 + C_2 * C_3 * S_1) \\
 & + C_4 * S_1 * S_2) + S_6 * (S_5 * (C_1 * C_3 - C_2 * S_1 * S_3) \\
 & + C_5 * (C_4 * (C_1 * S_3 + C_2 * C_3 * S_1) - S_1 * S_2 * S_4))) \\
 & + l_w^e * (S_4 * (C_1 * S_3 + C_2 * C_3 * S_1) + C_4 * S_1 * S_2) \\
 & + S_1 * S_2 * l_e^s
 \end{aligned} \tag{4.16}$$

$$\begin{aligned}
 p_z = & l_s^b + C_2 * l_e^s + l_w^e * (C_2 * C_4 - C_3 * S_2 * S_4) \\
 & + l_t^w * (C_6 * (C_2 * C_4 - C_3 * S_2 * S_4) - S_6 * (C_5 * (C_2 * S_4 \\
 & + C_3 * C_4 * S_2) - S_2 * S_3 * S_5))
 \end{aligned} \tag{4.17}$$

where  $l_s^b$ ,  $l_e^s$ ,  $l_w^e$  and  $l_t^w$  are the given link offsets. After deriving these equations it is possible now to obtain the end-effector position and orientation from the individual joint angles. Substituting the values of all the variables, will aid PARLOMA in finding the direct kinematics for any given specification.

## Chapter 5

# Inverse Kinematics

The inverse kinematics is widely used for controlling the robot manipulators. It has a finite number of solutions, provided, the number of degrees of freedom of the arm are enough to perform a task in its reachable work space. For a redundant manipulator it is possible to generate an infinite number of joint trajectories that lead to the same end-effector desired trajectory. From these infinite number of solutions only one solution should be selected in order to control the manipulator.

In this chapter, we discuss the different methods of computing the inverse kinematics, the second section talks about redundancy parameterization method at the position level. Then the joint angles of the manipulator are calculated in term of a new defined arm angle.

The inverse kinematic functions form the inverse kinematics, which transform the variables  $\mathbf{p}(t)$  and  $\dot{\mathbf{p}}(t)$  of the task space into the variables  $\mathbf{q}(t)$  and  $\dot{\mathbf{q}}(t)$  of joint space.

### 5.1 Methods available

#### 5.1.1 Redundancy parameterization

The redundancy parametrization[18] can be represented by an arm angle between a reference plane and arm plane. The arm plane defined as the plane spanned by the shoulder, elbow and wrist.

### 5.1.2 Jacobians

This method is very powerful, but also has the potential to be computationally expensive. The end effector has a start position with a goal position. Jacobian methods have three main steps from a top-down perspective:

1. Find the joint configurations:  $T$
2. Compute the change in rotations:  $d_O$
3. Compute the Jacobian:  $J$

### 5.1.3 Cyclic coordinate descent

The Cyclic Coordinate Descent[19] is based on minimization applied to each joint separately. The steps in one pass are ordered from the most distant segment to the base segment. Here only one joint variable is changing along the minimization process. That significantly speeds up the minimization problem.

## 5.2 Method applied in the case of a 7 DOF robotic arm

Since we are using an arm with with redundant degrees of freedom, we can possibly get many solutions for a particular condition. This is when Jacobian method fails as it needs more computations for obtaining all the possible solutions. Hence, redundancy parameterization can be used effectively.

## 5.3 Method applied

There are two different main methods for inverse kinematics, one solves the problem analytically at the position level and the other one solves it iteratively at the velocity level by approximating the solutions over time. Solving the inverse kinematics and exploiting the redundancy at the position level will be used in this work of thesis.

One of the published papers is [16], which looked at a method to compute all the solutions of the inverse kinematics solutions anyti-

cally for a 7-DOF redundant manipulators in the position domain. This is obtained by having constraints on the joint limits. The method proposed requires the use of the arm angle parameterization methods as proposed by [17] for the redundancy resolution. The arm angle parameterization tells that the motion of the manipulator can be represented by the arm angle geometry. The arm angle is defined as the angle between the reference plane and the arm plane, which is made up of by the shoulder, elbow and wrist. In [17] the reference plane is determined by a fixed vector. However, if this arbitrarily chosen fixed vector and the axis connecting the shoulder and wrist are co-linear, then the reference plane is considered to be indeterminate. Hence, [16] talks about an alternate definition for the reference plane by fixing joint angle 3 in the manipulator to zero and considering the plane spanned by the shoulder, elbow and wrist as the reference plane.

## 5.4 Results achieved

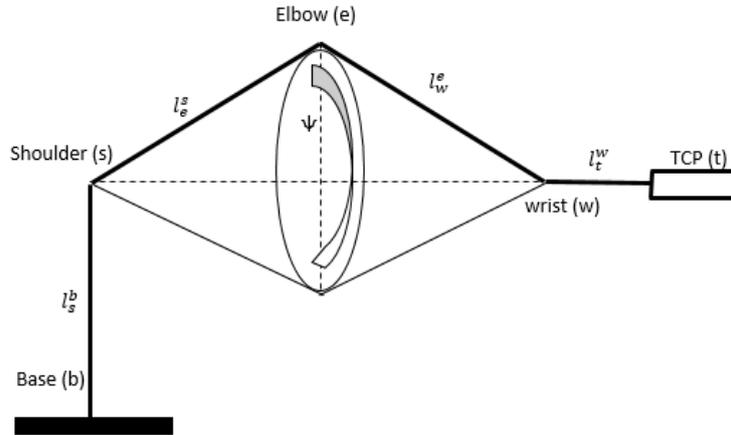


Figure 5.1: Arm angle

The redundancy parameterization can be represented by an arm angle between a reference plane and arm plane. The arm plane is the plane spanned by the shoulder, elbow and wrist as shown in figure 5.1. The reference plane is determined by the redundant manipulator because the non redundant that we assume, joint 3 angle, is equal to zero. So the axis of the rotation in joint 2 and 4 are parallel. Thus, the plane spanned by the shoulder, elbow and wrist in this non redundant arm can be regarded as a reference plane.

We use the following notations, the same as the ones we used in chapter 4:

- $l_s^b$  is the length from the base to the shoulder.
- $l_e^s$  is the length from the shoulder to the elbow.
- $l_w^e$  is the length from the elbow to the wrist.
- $l_t^w$  is the length from the wrist to the TCP rotating around z-axis.
- $\mathbf{P}_7^0$  is the position of the TCP.
- $\mathbf{R}_7^0$  is the matrix defining the orientation of the TCP.

The redundant degree of freedom represents the motion of the manipulator and is defined as the arm angle  $\psi$ . The axis is defined as the rotation of the arm is the motion around the axis between the shoulder and wrist. Now we consider  $\mathbf{L}_w^s$  be the vector between the centers of the shoulder and wrist. The calculation of the wrist position  $w$  is given by:

$$w = \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} = \mathbf{P}_7^0 - l_t^w \mathbf{R}_7^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (5.1)$$

Here the  $\mathbf{P}_7^0$  and  $\mathbf{R}_7^0$  are defined as the given position and orientation of the tool center point respectively. We know that the equation (5.1) showed the wrist position with respect to the base and now, the wrist position with respect to the shoulder can be computed as shown in equation (5.2).

$$\mathbf{L}_w^s = w - \mathbf{L}_s^b \quad (5.2)$$

Here,  $w$  is taken from (5.1) and  $\mathbf{L}_s^b$  is the vector between the centers of the base and the shoulder links, rotating around z axis. This is shown as in equation (5.3).

$$\mathbf{L}_s^b = \begin{bmatrix} 0 \\ 0 \\ l_s^b \end{bmatrix} \quad (5.3)$$

If for a given pose, the end-effector is fixed, then the wrist position also remains fixed. Hence, the vector link between shoulder and wrist,  $\mathbf{L}_w^s$  is considered to be constant as, all the variables on the right hand side of the equation (5.2) are constants for a stationary end-effector. Here, a rotation matrix has to be derived to represent the arm angle rotation around the link that connects the shoulder and wrist axes. This is as shown in figure 5.1. An algorithm to represent a rotation vector in space is the vector  $\mathbf{L}_w^s$  and angle of rotation.

$$\mathbf{R}_\psi = I = (1 - \cos(\psi))(\mathbf{K}_w^s)^2 + \sin(\psi)\mathbf{K}_w^s \quad (5.4)$$

Here  $I$  is the  $3 \times 3$  identity matrix and  $\mathbf{K}_w^s$  is a skew symmetric matrix of  $u_w^s$ . This is given as shown in equation (5.6)

$$\mathbf{K}_w^s = \begin{bmatrix} 0 & -u_w^s(z) & u_w^s(y) \\ u_w^s(z) & 0 & -u_w^s(x) \\ -u_w^s(y) & u_w^s(x) & 0 \end{bmatrix} \quad (5.5)$$

Here  $u_w^s(x), u_w^s(y)$  and  $u_w^s(z)$  can be defined as the x, y and z coordinates of the vector  $u_w^s$ .

$u_w^s$  is a unit vector that defines the direction of  $\mathbf{L}_w^s$ . This is given by (5.20).

$$u_w^s = \frac{\mathbf{L}_w^s}{\|\mathbf{L}_w^s\|} \quad (5.6)$$

Here  $\|\mathbf{L}_w^s\|$  represents the norm of  $\mathbf{L}_w^s$ .

As mentioned above, arm angle (whichever chosen) will not effect the wrist position. But, we can see that the spherical joint orientation of the wrist, as seen from the base of the arm, changes with the change in the arm angle. The fourth joint angle is independent independent of the arm angle that we have defined. Again however, a change in the arm angle would lead to a change in the spherical joint orientation of the shoulder. This change can be calculated as follows:

$$\mathbf{R}_3^0 = \mathbf{R}_\psi \cdot \mathbf{R}_3'^0 \quad (5.7)$$

Here  $\mathbf{R}_3^0$  represents the rotation matrix of the shoulder when the third joint angle is assumed to be zero (with respect to the reference plane). The arm angle also considered to be zero if and only if the plane of the arm coincides with the reference plane.

### 5.4.1 Joint angles

We now compute the seven joint angles of the arm angle that we have defined. This is based on the redundancy parameterization method discussed in the previous section.

#### Elbow joint angle $\theta_4$

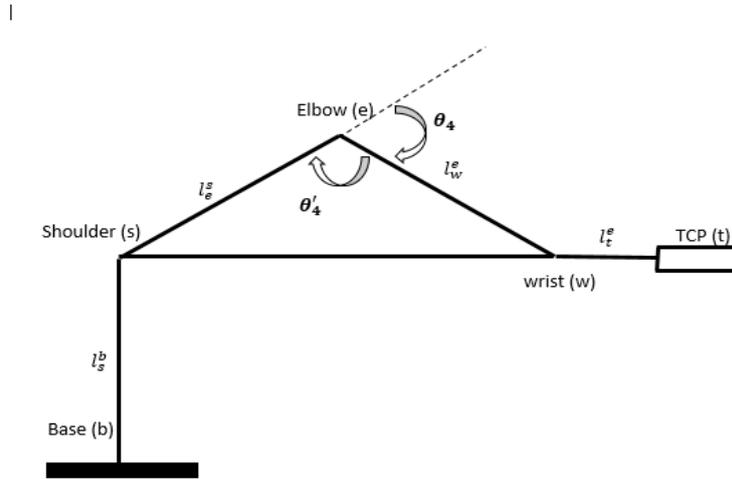


Figure 5.2: Elbow joint angle

Figure 5.2 shows that choosing any arm angle does not affect the elbow joint angle. For any given fixed end-effector, the elbow joint angle can be calculated. In figure 5.2, we apply the cosine law for further calculations.

We calculate the elbow joint prime angle( $\theta_0'$ ), as shown in equation (5.8).

$$\|\mathbf{L}_w^s\|^2 = l_e^{s2} + l_w^{e2} - 2 \cdot l_e^s \cdot l_w^e \cdot \cos(\theta_0') \quad (5.8)$$

From the above equation  $\cos(\theta'_0)$  can be derived as:

$$\cos(\theta'_0) = \frac{l_e^{s2} + l_w^e - \|\mathbf{L}_w^s\|^2}{2 \cdot l_e^s \cdot l_w^e} \quad (5.9)$$

Here  $\|\mathbf{L}_w^s\|$  shows the length of the vector we had calculated in the equation (5.2). Now, from this equation, we can easily calculate  $\theta_4$  as shown in equation (5.10).

$$\theta_4 = \pi - \theta'_0 \quad (5.10)$$

### Shoulder joint angles ( $\theta_1$ , $\theta_2$ and $\theta_3$ )

The shoulder joint angles depend on the arm angle that we had defined earlier. The representation of the shoulder joint angles is done using the equation (5.7). Varying the arm angle generates different values for  $\theta_1$ ,  $\theta_2$  and  $\theta_3$ , but with the same given pose of end-effector. Calculating the shoulder joint angles with the use of the defined arm angle, requires the need of a reference plane. We see from the definition of redundancy parameterization, that the reference plane is the plane that contains the should, elbow, wrist links when the third joint angle is equal to zero. Let  $\theta'_1$ ,  $\theta'_2$  and  $\theta'_3$  be the joint angles when the rotation matrix  $\mathbf{R}'_3{}^0$  for the associated reference plane. The redundant manipulator then becomes non-redundant when we assign the  $\theta_3'$  as 0.

Now we see that when the wrist position is projected onto the horizontal plane, we can calculate the  $\theta'_1$  as follows

$$\theta'_1 = \text{atan2}(w_y, w_x) \quad (5.11)$$

Here  $w_x$  and  $w_y$  are the x and y coordinates of the wrist vector seen in equation (5.1). The figure 5.3 shows that the  $\theta'_2$  can be calculated by the following equation

$$\theta'_2 = \frac{\pi}{2} - \alpha - \beta \quad (5.12)$$

Here  $\alpha$  is represented as in equation (5.13) and  $\beta$  is repre-

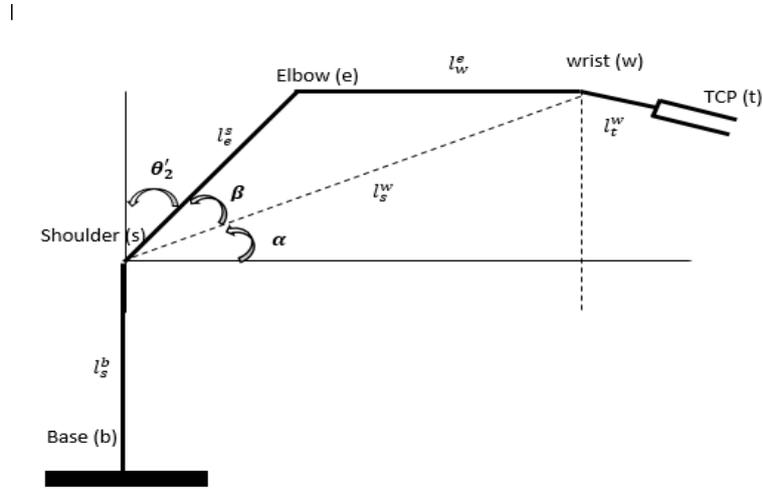


Figure 5.3: Shoulder joint angle

sented as given in equation (5.14)

$$\alpha = \sin^{-1} \left( \frac{w_z \cdot l_s^b}{\|\mathbf{L}_s^w\|} \right) \quad (5.13)$$

Here  $w_z$  is the vector in the z coordinate which is calculated as shown in equation (5.1).

$$\beta = \cos^{-1} \left( \frac{l_e^{s^2} + \|\mathbf{L}_w^s\|^2 - l_w^{e^2}}{2 \cdot l_s^w \cdot \|\mathbf{L}_w^s\|} \right) \quad (5.14)$$

Now, we calculate  $\theta_1'$  and  $\theta_2'$  using the equation (5.11) and equation (5.12) respectively. Now that we know that  $\theta_3'$  equals to zero, we can use the rotational matrix to calculate  $\mathbf{R}_3'^0$ .

$$\mathbf{R}_i'^{i-1} = \begin{bmatrix} \cos(\theta_i') & -\sin(\theta_i')\cos(\alpha_i) & \sin(\theta_i')\cos(\alpha_i) \\ \sin(\theta_i') & \cos(\theta_i')\sin(\alpha_i) & -\cos(\theta_i')\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \quad (5.15)$$

By substituting the values of  $\theta_1'$ ,  $\theta_2'$  and  $\theta_3' = 0$  into the equation (5.15) and then using the parameters as in the Denavit Hartenberg

table, we obtain the following equation (5.16).

$$\mathbf{R}'_1{}^0 \mathbf{R}'_2{}^1 \mathbf{R}'_3{}^2 = \begin{bmatrix} C_{\theta'_1} & 0 & -S_{\theta'_1} \\ S_{\theta'_1} & 0 & C_{\theta'_1} \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} C_{\theta'_2} & 0 & S_{\theta'_2} \\ S_{\theta'_2} & 0 & -C_{\theta'_2} \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad (5.16)$$

Here  $\mathbf{R}'_1{}^0$ ,  $\mathbf{R}'_2{}^1$  and  $\mathbf{R}'_3{}^2$  are the rotation matrices of the reference joint angles.

Now, the matrix  $\mathbf{R}'_3{}^0$  can be computed as shown in equation (5.17):

$$\mathbf{R}'_3{}^0 = \mathbf{R}'_1{}^0 \mathbf{R}'_2{}^1 \mathbf{R}'_3{}^2 = \begin{bmatrix} C_{\theta'_1} C_{\theta'_2} & -C_{\theta'_1} S_{\theta'_2} & -S_{\theta'_1} \\ C_{\theta'_2} S_{\theta'_1} & -S_{\theta'_1} S_{\theta'_2} & C_{\theta'_1} \\ -S_{\theta'_2} & -C_{\theta'_2} & 0 \end{bmatrix} \quad (5.17)$$

The matrix in the equation (5.17) is constant for any given pose. It is now possible to compute the equations for the shoulder joint angles with the use of the arm angle by substituting the equations (5.4) and (5.17) into (5.7), which will result in the following equation

$$\mathbf{R}_3^0 = \sin(\psi) \mathbf{K}_w^s \mathbf{R}'_3{}^0 - \cos(\psi) (\mathbf{K}_w^s)^2 \mathbf{R}'_3{}^0 + (I + (\mathbf{K}_w^s)^2) \mathbf{R}'_3{}^0 \quad (5.18)$$

The right-hand side of the equation (4.18) shows the  $\mathbf{K}_w^s$ , which is given as in equation (5.5) and  $\mathbf{R}_3^0$  is as given as in equation (5.17). So now, the shoulder joint angles  $\theta_1$ ,  $\theta_2$  and  $\theta_3$  which are represented by the arm angle  $\psi$  are given in equation (5.18). For simplification, the equation (5.18) is rewritten as follows

$$\mathbf{R}_3^0 = \sin(\psi) \cdot \mathbf{X}_s + \cos(\psi) \cdot \mathbf{Y}_s + \mathbf{Z}_s \quad (5.19)$$

Here the matrices  $\mathbf{X}_s$ ,  $\mathbf{Y}_s$  and  $\mathbf{Z}_s$  are given constant matrices as defined by equations (5.20), (5.21) and (5.22) respectively.

$$\mathbf{X}_s = \mathbf{K}_w^s \mathbf{R}'_3{}^0 = \begin{bmatrix} X_{s11} & X_{s12} & X_{s13} \\ X_{s21} & X_{s22} & X_{s23} \\ X_{s31} & X_{s32} & X_{s33} \end{bmatrix} \quad (5.20)$$

$$\mathbf{Y}_s = -(\mathbf{K}_w^s)^2 \mathbf{R}'_3 \mathbf{0}_3 = \begin{bmatrix} Y_{s11} & Y_{s12} & Y_{s13} \\ Y_{s21} & Y_{s22} & Y_{s23} \\ Y_{s31} & Y_{s32} & Y_{s33} \end{bmatrix} \quad (5.21)$$

$$\mathbf{Z}_s = (1 + (\mathbf{K}_w^s)^2) \mathbf{R}_3^0 = \begin{bmatrix} Z_{s11} & Z_{s12} & Z_{s13} \\ Z_{s21} & Z_{s22} & Z_{s23} \\ Z_{s31} & Z_{s32} & Z_{s33} \end{bmatrix} \quad (5.22)$$

Equation (5.19) can now be re-written in matrix form to compute the shoulder joint angles. This is done by comparing the elements of the matrices in the right-hand side and left-hand side of this equation.  $\mathbf{R}_3^0$  matrix is computed by substituting the parameters of the shoulder joints used in the Denavit Hartenberg table into the equation (5.15) to enable the computation of the matrix form in (5.23).

$$\mathbf{R}_3^0 = \begin{bmatrix} C_1 C_2 C_3 - S_1 S_3 & -C_1 S_2 & -C_3 S_1 - C_1 C_2 S_3 \\ C_1 S_3 + C_2 C_3 S_1 & -S_1 S_2 & C_1 C_3 - C_2 C_1 S_3 \\ -S_2 C_3 & -C_2 & S_2 S_3 \end{bmatrix} \quad (5.23)$$

The right hand-side of the equation (5.19) is re-written in the form of matrix as shown in equation (5.24).

$$\mathbf{R}_3^0 = \begin{bmatrix} S_\psi X_{s11} + C_\psi Y_{s11} + Z_{s11} & S_\psi X_{s12} + C_\psi Y_{s12} + Z_{s12} & S_\psi X_{s13} + C_\psi Y_{s13} + Z_{s13} \\ S_\psi X_{s21} + C_\psi Y_{s21} + Z_{s21} & S_\psi X_{s22} + C_\psi Y_{s22} + Z_{s22} & S_\psi X_{s23} + C_\psi Y_{s23} + Z_{s23} \\ S_\psi X_{s31} + C_\psi Y_{s31} + Z_{s31} & S_\psi X_{s32} + C_\psi Y_{s32} + Z_{s32} & S_\psi X_{s33} + C_\psi Y_{s33} + Z_{s33} \end{bmatrix} \quad (5.24)$$

Here  $S_\psi$  and  $C_\psi$  are used in the place of  $\sin(\psi)$  and  $\cos(\psi)$  respectively. Equating the elements given in the matrices of the equations (5.23) and (5.24), it is how possible to calculate the shoulder joint angles. The joint angle  $\theta_1$  can be calculated by comparing  $\mathbf{R}_3^0(1, 2)$  and  $\mathbf{R}_3^0(2, 2)$ . This is illustrated in the equation (5.25)

$$\frac{-S_{\theta_1} S_{\theta_2}}{-C_{\theta_1} S_{\theta_2}} = \frac{S_\psi X_{s22} + C_\psi Y_{s22} + Z_{s22}}{S_\psi X_{s12} + C_\psi Y_{s12} + Z_{s12}} \quad (5.25)$$

The above equation can be rewritten as the following equation (5.26). This equation shows the joint angle  $\theta_1$  which is represented by the arm angle  $\psi$ .

$$\theta_1 = \text{atan} \left( \frac{-S_\psi X_{s22} - C_\psi Y_{s22} - Z_{s22}}{-S_\psi X_{s12} - C_\psi Y_{s12} - Z_{s12}} \right) \quad (5.26)$$

Similarly we compare  $\mathbf{R}_3^0(3, 2)$  in the equations (5.37) and (5.38), to calculate the shoulder joint angle  $\theta_2$  as given by the equation (5.27)

$$-C_{\theta_2} = S_\psi X_{s32} + C_\psi Y_{s32} + Z_{s32} \quad (5.27)$$

This above equation can be rewritten as:

$$\theta_2 = \text{acos}(-S_\psi X_{s32} - C_\psi Y_{s32} - Z_{s32}) \quad (5.28)$$

Again, in same way, we compare  $\mathbf{R}_3^0(3, 1)$  and  $\mathbf{R}_3^0(3, 3)$  given by the equations (5.23) and (5.24) to obtain the following equation.

$$\frac{-S_{\theta_2} S_{\theta_3}}{-C_{\theta_2} S_{\theta_3}} = \frac{S_\psi X_{s33} + C_\psi Y_{s33} + Z_{s33}}{S_\psi X_{s31} + C_\psi Y_{s31} + Z_{s31}} \quad (5.29)$$

By simplifying the equation (5.29), we can calculate the shoulder joint angle  $\theta_3$  as shown in equation (5.30).

$$\theta_3 = \text{atan} \left( \frac{S_\psi X_{s33} + C_\psi Y_{s33} + Z_{s33}}{-S_\psi X_{s31} - C_\psi Y_{s31} - Z_{s31}} \right) \quad (5.30)$$

So now, the equations (5.26), (5.28) and (5.30) represent the shoulder joint angles  $\theta_1$ ,  $\theta_2$  and  $\theta_3$  which also happen to be represented in the arm angle  $\psi$ .

### Wrist joint angles( $\theta_5$ , $\theta_6$ and $\theta_7$ )

When the orientation of the end-effector is given, that is,  $\mathbf{R}_7^0$  which is the rotation matrix with respect to the base, we can calculate the

following equation.

$$\mathbf{R}_7^0 = \mathbf{R}_3^0 \mathbf{R}_4^3 \mathbf{R}_7^4 \quad (5.31)$$

The above equation can now be rewritten representing the arm angle as shown in the equation (5.32)

$$\mathbf{R}_7^4 = (\mathbf{R}_4^3)^T (\mathbf{R}_3^0)^T \mathbf{R}_7^0 \quad (5.32)$$

Here,  $\mathbf{R}_3^0$  is given by the equation that we saw in (5.7) and  $\mathbf{R}_4^3$  gotten by substituting the elbow joint  $\theta_4$  that we computed in equation (5.10). The link parameters of the fourth joint are taken from the denavit hartenberg table, and substituted in the equation (5.15). Now,  $\mathbf{R}_7^4$  can be computed as in equation (5.33).

$$\mathbf{R}_7^4 = \sin(\psi)(X)_w + \cos(\psi)(Y)_w + (Z)_w \quad (5.33)$$

Here  $\mathbf{X}_w$ ,  $\mathbf{Y}_w$  and  $\mathbf{Z}_w$  represent the constant matrices for a given pose. These constant matrices are shown by the equations (5.34), (5.35) and (5.36).

$$\mathbf{X}_s = (\mathbf{R}_4^3)^T (\mathbf{K}_w^s)^T (\mathbf{R}_3^0)^T = \begin{bmatrix} X_{w11} & X_{w12} & X_{w13} \\ X_{w21} & X_{w22} & X_{w23} \\ X_{w31} & X_{w32} & X_{w33} \end{bmatrix} \quad (5.34)$$

$$\mathbf{Y}_s = -((\mathbf{R}_4^3)^T)^T ((\mathbf{K}_w^s)^2)^T (\mathbf{R}_3^0)^T = \begin{bmatrix} Y_{w11} & Y_{w12} & Y_{w13} \\ Y_{w21} & Y_{w22} & Y_{w23} \\ Y_{w31} & Y_{w32} & Y_{w33} \end{bmatrix} \quad (5.35)$$

$$\mathbf{Z}_s = ((\mathbf{R}_4^3)^T)^T (1 + (\mathbf{K}_w^s)^2)^T (\mathbf{R}_3^0)^T = \begin{bmatrix} Z_{w11} & Z_{w12} & Z_{w13} \\ Z_{w21} & Z_{w22} & Z_{w23} \\ Z_{w31} & Z_{w32} & Z_{w33} \end{bmatrix} \quad (5.36)$$

Just like we calculated the shoulder joint angles, the elbow joint angles can now be calculated. Equation (5.33) is re-written in

matrix form to compute the wrist joint angles easily. This is achieved by comparing the matrices in the right hand side and the left hand side of the equations.  $\mathbf{R}_7^4$  matrix can be computed by substituting the parameters of the wrist joints as shown in the Denavit Hartenberg table in the( quation e5.15) which leads us to achieve the following equation.

$$\mathbf{R}_7^4 = \begin{bmatrix} C_5 C_6 C_7 - S_5 S_7 & -C_7 S_5 - C_5 C_6 S_7 & -C_5 S_6 \\ C_5 S_7 + C_6 C_7 S_5 & C_5 C_7 - C_6 C_5 S_7 & -S_5 S_6 \\ -S_6 C_7 & S_6 S_7 & -C_6 \end{bmatrix} \quad (5.37)$$

The right-hand side of the equation (5.33) can now be rewritten as follows

$$\mathbf{R}_7^4 = \begin{bmatrix} S_\psi X_{w11} + C_\psi Y_{w11} + Z_{w11} & S_\psi X_{w12} + C_\psi Y_{w12} + Z_{w12} & S_\psi X_{w13} + C_\psi Y_{w13} + Z_{w13} \\ S_\psi X_{w21} + C_\psi Y_{w21} + Z_{w21} & S_\psi X_{w22} + C_\psi Y_{w22} + Z_{w22} & S_\psi X_{w23} + C_\psi Y_{w23} + Z_{w23} \\ S_\psi X_{w31} + C_\psi Y_{w31} + Z_{w31} & S_\psi X_{w32} + C_\psi Y_{w32} + Z_{w32} & S_\psi X_{w33} + C_\psi Y_{w33} + Z_{w33} \end{bmatrix} \quad (5.38)$$

Using  $S_\psi$  and  $C_\psi$  are used to represent  $\sin(\psi)$  and  $\cos(\psi)$  respectively. Comparing the elements of the matrix given in the equations (5.37) and (5.38), it is now possible to compute the wrist joint angles. The joint angle  $\theta_5$  can be calculated by comparing  $\mathbf{R}_7^4(1, 3)$  and  $\mathbf{R}_7^4(2, 3)$  as shown in the following equation (5.39)

$$\frac{S_{\theta_5} S_{\theta_6}}{-C_{\theta_5} S_{\theta_6}} = \frac{S_\psi X_{w23} + C_\psi Y_{w23} + Z_{w23}}{S_\psi X_{w13} + C_\psi Y_{w13} + Z_{w13}} \quad (5.39)$$

The above equation can be now be rewritten as the following equation (5.40), which shows the joint angle  $\theta_5$  which is represented in term of the arm angle  $\psi$ .

$$\theta_5 = \tan^{-1} \left( \frac{S_\psi X_{w23} + C_\psi Y_{w23} + Z_{w23}}{S_\psi X_{w13} + C_\psi Y_{w13} + Z_{w13}} \right) \quad (5.40)$$

We can now compare  $\mathbf{R}_7^4(3, 3)$  given in the equations (5.37) and (5.38), to compute the joint angle  $\theta_6$ . This is given by the equation

(5.41)

$$C_{\theta_6} = S_{\psi}X_{w33} + C_{\psi}Y_{w33} + Z_{w33} \quad (5.41)$$

The above equation can be rewritten to obtain the sixth joint angle equation:

$$\theta_6 = (\cos)^T(S_{\psi}X_{w33} + C_{\psi}Y_{w33} + Z_{w33}) \quad (5.42)$$

Again by comparing  $\mathbf{R}_7^4(3, 2)$  and  $\mathbf{R}_7^4(3, 1)$  given in the equations (5.37) and (5.38) we obtain:

$$\frac{S_{\theta_6}S_{\theta_7}}{-C_{\theta_7}S_{\theta_6}} = \frac{S_{\psi}X_{w32} + C_{\psi}Y_{w32} + Z_{w32}}{S_{\psi}X_{w31} + C_{\psi}Y_{w31} + Z_{w31}} \quad (5.43)$$

By simplifying the equation (5.43), we can compute the joint angle  $\theta_7$  as shown by the equation (5.44).

$$\theta_7 = \text{atan} \left( -\frac{S_{\psi}X_{w32} + C_{\psi}Y_{w32} + Z_{w32}}{S_{\psi}X_{w31} + C_{\psi}Y_{w31} + Z_{w31}} \right) \quad (5.44)$$

Now we have calculated the generic equations for all seven joint angles of the manipulator in term of the new arm angle parameter  $\psi$ .

## Chapter 6

# Robotic finger mechanism

Due to the inherent complexity of a real human hand, there are a lot of conciliations which are fundamentally put on the anthropomorphic robotic hands to make them as dexterous as a human hand. In this chapter we look at the mathematical model of the robotic finger mechanism.

### 6.1 Approaches

Since our aim is to find the possible spring constant for a finger joint, we look at the different methods available under rigid body mechanics.

#### 6.1.1 Bond-Graph method

Bond graph method can be used to model the model the hand in such a way that the it is easier to visualize the energy exchange. It is similar to a block diagram or signal-flow graph, with the major difference that the arcs in bond graphs represent bi-directional exchange of physical energy, while those in block diagrams and signal-flow graphs represent uni-directional flow of information.

### 6.1.2 Free body diagram

Free-body diagrams are diagrams used to show the relative magnitude and direction of all forces acting upon an object in a given situation. In this method, all the physical attributions of the object are removed. The object, henceforth, is represented by a simple single line. Every connection is represented with a unique value, or is replaced by a set of forces and moments, that represent the action due to that particular connection. Internal forces found at a connection is replaced by representational external forces. The point at which it connects, then connect to other objects in the FBD.

### 6.1.3 Lagrange method

Lagrangian formulation of modeling derives from the basic work–energy principle and Newton’s laws of motion.

1. Find the inputs and outputs of the system to be modelled
2. Find the generalized coordinates and compute the Kinetic energy(K) and the potential energy(U)
3. Compute the Lagrange function  $L = K - U$
4. Compute the differential equations

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = F_{nc} \quad (6.1)$$

where  $F_{nc}$  is the sum of all non-conservative forces acting in  $q_j$  direction

## 6.2 Method used for modelling

As we use these approaches to find the spring constant, we see that the Bond-graph method at some point gets more difficult to be solved and so does free body diagram. So we follow the Lagrange method to achieve the desired results.

We use the following list of symbols to understand the mathematical modelling better.

Table 6.1: List of symbols

symbol	description	unit
$i$	i-th Phalange	
$\theta_i$	angle between phalange and y-axis	rad
$\dot{\theta}_i$	angular velocity between phalange and y-axis	rad/s
$\ddot{\theta}_i$	angular acceleration between phalange and y-axis	rad/s <sup>2</sup>
$L_i$	length of the i-th phalange	m
$m_i$	mass of the i-th phalange	kg
$k_i$	spring stiffness coefficient of i-th joint	Nm/rad
$c_i$	damping coefficient of i-th joint	Nm · s/rad
$I_i$	moment of inertia of i-th phalange	kg · m <sup>2</sup>
$x_i$	displacement along x-axis	m
$\dot{x}_i$	velocity along x-axis	m/s
$\ddot{x}_i$	acceleration along x-axis	m/s <sup>2</sup>
$y_i$	displacement along y-axis	m
$\dot{y}_i$	velocity along y-axis	m/s
$\ddot{y}_i$	acceleration along y-axis	m/s <sup>2</sup>

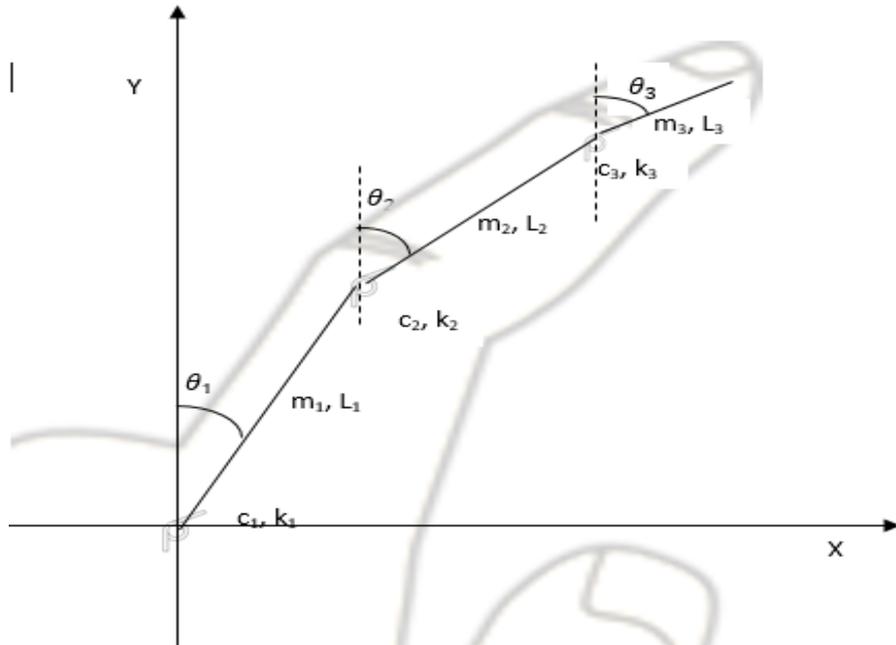


Figure 6.1: Spring loaded finger

Figure 6.1 shows the forces, mass, and angles involved in the movement of the phalanges of a finger. The angle between the y-axis and the phalange is  $\theta_i$ ,  $L_i$ , length of the phalanx,  $m_i$  is the center of mass of the i-th phalange, and  $k_i$  is the spring stiffness coefficient of the spring of i-th joint.  $c_i$  is the damping coefficient of i-th joint and  $I_i$  is the moment of inertia of the i-th phalange. An idealized mechanical model considered for our modelling wherein each phalange is assumed to be infinitely rigid with its mass concentrated at the centre. Herein we also assume that all the joints are friction-less. The first phalange considered as a rigid body and is assumed to be a rotational link as the first joint is assumed to be stationary. Looking at the motion of the masses  $m_2$  and  $m_3$ , we see that they are more complicated as they involve rotational motion about the joint while the joints are also moving and not stationary.

### 6.3 Results achieved

The Lagrangian method is used to get the equations of motion.

Firstly, we start by computing the kinetic energy of the system. The equation of the kinetic energy is given by the equation (6.2)

$$K = \frac{1}{2}I_1\dot{\theta}_1^2 + \frac{1}{2}m_2(\dot{x}_2^2 + \dot{y}_2^2) + \frac{1}{2}I_2\dot{\theta}_2^2 + \frac{1}{2}m_3(\dot{x}_3^2 + \dot{y}_3^2) + \frac{1}{2}I_3\dot{\theta}_3^2 \quad (6.2)$$

Then we calculate the potential energy of the system. This potential energy equation is given by equation (6.3).

$$U = m_1g \cdot \frac{1}{2}L_1\cos\theta_1 + m_2g[L_1\cos\theta_1 + \frac{1}{2}L_2\cos\theta_2] + m_3g[L_1\cos\theta_1 + L_2\cos\theta_2 + \frac{1}{2}L_3\cos\theta_3] + \frac{1}{2}k_1\theta_1^2 + \frac{1}{2}k_2(\theta_1 - \theta_2)^2 + \frac{1}{2}k_3(\theta_3 - \theta_2)^2 \quad (6.3)$$

Here, we find the positions, velocities and the accelerations of all three phalanges as shown in the equations (6.4) through (6.11).

$$x_2 = L_1\sin\theta_1 + \frac{1}{2}L_2\sin\theta_2 \quad (6.4)$$

$$\dot{x}_2 = L_1\dot{\theta}_1\cos\theta_1 + \frac{1}{2}\dot{\theta}_2L_2\cos\theta_2 \quad (6.5)$$

$$y_2 = L_1\cos\theta_1 + \frac{1}{2}L_2\cos\theta_2 \quad (6.6)$$

$$\dot{y}_2 = -L_1\dot{\theta}_1\sin\theta_1 - \frac{1}{2}L_2\dot{\theta}_2\sin\theta_2 \quad (6.7)$$

$$x_3 = L_1\sin\theta_1 + L_2\sin\theta_2 + \frac{1}{2}L_3\sin\theta_3 \quad (6.8)$$

$$\dot{x}_3 = L_1\dot{\theta}_1\cos\theta_1 + L_2\dot{\theta}_2\cos\theta_1 + \frac{1}{2}L_3\dot{\theta}_3\cos\theta_3 \quad (6.9)$$

$$y_3 = L_1\cos\theta_1 + L_2\cos\theta_2 + \frac{1}{2}L_3\cos\theta_3 \quad (6.10)$$

$$\dot{y}_3 = -L_1\dot{\theta}_1\sin\theta_1 - L_2\dot{\theta}_2\sin\theta_1 - \frac{1}{2}L_3\dot{\theta}_3\sin\theta_3 \quad (6.11)$$

Substituting (6.4) to (6.11),in (6.2), we get,

$$\begin{aligned} K = & \left[ \frac{1}{2}I_1 + \frac{1}{2}m_2L_1^2 + \frac{1}{2}m_3L_1^2 \right] \dot{\theta}_1^2 + \left[ \frac{1}{2}I_2 + \frac{1}{8}m_2L_2^2 + \frac{1}{2}m_3L_2^2 \right] \dot{\theta}_2^2 \\ & + \left[ \frac{1}{2}I_3 + \frac{1}{8}m_3L_3^2 \right] \dot{\theta}_3^2 + \left[ \frac{1}{2}m_2L_1L_2 + m_3L_1L_2 \right] \dot{\theta}_1\dot{\theta}_2\cos(\theta_2 - \theta_1) \\ & + \left[ \frac{1}{2}m_3L_1L_3 \right] \dot{\theta}_1\dot{\theta}_3\cos(\theta_3 - \theta_1) + \left[ \frac{1}{2}m_3L_2L_3 \right] \dot{\theta}_2\dot{\theta}_3\cos(\theta_3 - \theta_2) \end{aligned} \quad (6.12)$$

Now for simplification, we make the following substitutions. We club all the constant terms together and assign them to a common constant.

$$X_1 = \left[ \frac{1}{2}I_1 + \frac{1}{2}m_2L_1^2 + \frac{1}{2}m_3L_1^2 \right] \quad (6.13)$$

$$X_2 = \left[ \frac{1}{2}I_2 + \frac{1}{8}m_2L_2^2 + \frac{1}{2}m_3L_2^2 \right] \quad (6.14)$$

$$X_3 = \left[ \frac{1}{2}I_3 + \frac{1}{8}m_3L_3^2 \right] \quad (6.15)$$

$$X_4 = \left[ \frac{1}{2}m_2L_1L_2 + m_3L_1L_2 \right] \quad (6.16)$$

$$X_5 = \left[ \frac{1}{2}m_3L_1L_3 \right] \quad (6.17)$$

$$X_6 = \left[ \frac{1}{2}m_3L_2L_3 \right] \quad (6.18)$$

Substituting (6.4) to (6.11),in (6.3), we get,

$$\begin{aligned} U = & \left[ \frac{1}{2}m_1 + m_2 + m_3 \right] gL_1 \cos\theta_1 + \left[ \frac{1}{2}m_2 + m_3 \right] gL_2 \cos\theta_2 \\ & + \left[ \frac{1}{2}m_3 \right] gL_3 \cos\theta_3 + \frac{1}{2} [k_1 + k_2] \theta_1^2 + \frac{1}{2} [k_2 + k_3] \theta_2^2 \\ & + \frac{1}{2} k_3 \theta_3^2 - K_2 \theta_1 \theta_2 - k_3 \theta_2 \theta_3 \end{aligned} \quad (6.19)$$

Again, for simplification, we make the following substitutions by replacing all the constants elements in a term to a common constant.

$$Y_1 = \left[ \frac{1}{2}m_1 + m_2 + m_3 \right] gL_1 \quad (6.20)$$

$$Y_2 = \left[ \frac{1}{2}m_2 + m_3 \right] gL_2 \quad (6.21)$$

$$Y_3 = \left[ \frac{1}{2}m_3 \right] gL_3 \quad (6.22)$$

$$Y_4 = \frac{1}{2} [k_1 + k_2] \quad (6.23)$$

$$Y_5 = \frac{1}{2} [k_2 + k_3] \quad (6.24)$$

$$Y_6 = \frac{1}{2} k_3 \quad (6.25)$$

$$Y_7 = k_2 \quad (6.26)$$

$$Y_8 = k_3 \quad (6.27)$$

Now we calculate the Lagrange equation using the formula  $L= K-U$

$$\begin{aligned} L = & X_1\dot{\theta}_1^2 + X_2\dot{\theta}_2^2 + X_3\dot{\theta}_3^2 + X_4\dot{\theta}_1\dot{\theta}_2\cos(\theta_2 - \theta_1) + X_5\dot{\theta}_1\dot{\theta}_3\cos(\theta_3 - \theta_1) \\ & + X_6\dot{\theta}_2\dot{\theta}_3\cos(\theta_3 - \theta_2) - Y_1\cos\theta_1 - Y_2\cos\theta_2 - Y_3\cos\theta_3 - Y_4\theta_1^2 \\ & - Y_5\theta_2^2 - Y_6\theta_3^2 + Y_7\theta_1\theta_2 + Y_8\theta_2\theta_3 \end{aligned} \quad (6.28)$$

By calculating the three differential equations we get,

First differential equation,

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_1} - \frac{\partial L}{\partial \theta_1} = -c_1\dot{\theta}_1 - c_2(\dot{\theta}_1 - \dot{\theta}_2) \quad (6.29)$$

$$\begin{aligned} \ddot{\theta}_1 = & \frac{X_4}{2X_1}\dot{\theta}_2^2\sin(\theta_2 - \theta_1) - \frac{X_4}{2X_1}\ddot{\theta}_2\cos(\theta_2 - \theta_1) + \frac{X_5}{2X_1}\dot{\theta}_3^2\sin(\theta_3 - \theta_1) \\ & - \frac{X_5}{2X_1}\ddot{\theta}_3\cos(\theta_3 - \theta_1) + \frac{Y_1}{2X_1}\sin\theta_1 - \frac{Y_4}{2X_1}\theta_1 + \frac{Y_7}{2X_1}\theta_2 \\ & - \frac{c_1 - c_2}{2X_1}\dot{\theta}_1 + \frac{c_2}{2X_1}\dot{\theta}_2 \end{aligned} \quad (6.30)$$

Second differential equation

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_2} - \frac{\partial L}{\partial \theta_2} = -c_2(\dot{\theta}_2 - \dot{\theta}_1) - c_3(\dot{\theta}_2 - \dot{\theta}_3) \quad (6.31)$$

$$\begin{aligned}
 \ddot{\theta}_2 = & -\frac{X_4}{2X_2}\dot{\theta}_1^2 \sin(\theta_2 - \theta_1) - \frac{X_4}{2X_2}\ddot{\theta}_1 \cos(\theta_2 - \theta_1) + \frac{X_6}{2X_2}\dot{\theta}_3^2 \sin(\theta_3 - \theta_2) \\
 & - \frac{X_6}{2X_2}\ddot{\theta}_3 \cos(\theta_3 - \theta_2) + \frac{Y_2}{2X_2} \sin\theta_2 - \frac{Y_5}{2X_2}\theta_2 + \frac{Y_7}{2X_2}\theta_1 \\
 & + \frac{Y_8}{2X_2}\theta_3 - \frac{c_2 + c_3}{2X_2}\dot{\theta}_2 + \frac{c_2}{2X_2}\dot{\theta}_1 + \frac{c_3}{2X_2}\dot{\theta}_3
 \end{aligned} \tag{6.32}$$

Third Differential equation

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_3} - \frac{\partial L}{\partial \theta_3} = -c_3(\dot{\theta}_3 - \dot{\theta}_2) + F \tag{6.33}$$

$$\begin{aligned}
 \ddot{\theta}_3 = & -\frac{X_5}{2X_3}\dot{\theta}_1^2 \sin(\theta_3 - \theta_1) - \frac{X_5}{2X_3}\ddot{\theta}_1 \cos(\theta_3 - \theta_1) - \frac{X_6}{2X_3}\dot{\theta}_2^2 \sin(\theta_3 - \theta_2) \\
 & - \frac{X_6}{2X_3}\ddot{\theta}_2 \cos(\theta_3 - \theta_2) + \frac{Y_3}{2X_3} \sin\theta_3 - \frac{Y_6}{2X_3}\theta_3 + \frac{Y_8}{2X_3}\theta_2 \\
 & - \frac{c_3}{2X_2}(\dot{\theta}_3 - \dot{\theta}_2) + \frac{1}{2X_3}F
 \end{aligned} \tag{6.34}$$

We can use all these three differential equations to calculate either the motion of the anthropomorphic hand or to find the values of the constants used for modelling the same provided all other values of the system are known.

## Chapter 7

# Conclusion

The thesis presents the kinematics involved in the development of a 7 DOF anthropomorphic arm. Further, it discusses the robotic fingers that are modelled with the Lagrangian Method.

We start by looking at the PARLOMA project to figure out a novel way of moving further ahead with the project. When we look at the anthropomorphic arm, we are faced with a few constraints which are essential for the development of the project, namely weight of the arm, payload and degrees of freedom. These constraints are adhered to, and we find that Yaskawa SIA5F manipulator, which fulfills all our constraints.

After figuring out the optimal manipulator, we look at the direct kinematics of the system. Since the manipulator is a 7 DOF system, we see that the complexity of calculations increases exponentially. We solve the direct kinematics problem using the Denavit Hartenberg convention to compute the homogeneous transformation. Since in our case, there is no distance along any x axes, from the origins to the point of intersection between  $x_i$  and  $z_{i-1}$ , the link length is taken to be zero in all the frames. Now, this homogeneous transformation can be further used to extract the direct position kinematic functions and later, to find the direct velocity kinematic functions depending on what would be required.

Now we look at the inverse kinematics of manipulator and see that it is not as straight-forward as the direct kinematics. This is because, since our manipulator is a redundant manipulator, we get infinite number of solutions of the joint trajectories that lead to the same desired end-effector trajectory. From these infinite solutions, then, one solution must be selected to control the manipulator. Choosing that unique solution

in this case becomes very difficult. Hence, we choose the redundancy parameterization to get this unique solution. The redundancy parameterization is represented by an arm angle between the reference plane and the arm plane(plane spanned by the shoulder, elbow and wrist). Using this arm angle, we then get the values of all the joint angles. Moving on from the anthropomorphic arm, we look into the anthropomorphic hand. We see that the hand too, has constraints such as the hand should be bio-mimetic, should be low cost and should be light in weight. This system that get, can be modelled in different ways. Modelling it with bondgraph, makes it very complicated. Free-body diagram gives similar uncertain results. Finally using the Lagrangian method, we compute three differential equations. These equations can further be used to calculate the spring constants, provided all other values are known.

## 7.1 Future work

The development of the manipulator requires further work along with the motion planning. Singularities and redundancies can be calculated with more information on the manipulator. We can also potentially see the effects of the arm angle on the joint angle with more manipulations. In the case of the robotic finger, simulations can be carried out to check their effects on the spring loaded joints.

# Bibliography

- [1] M. Liarokapis, P. Artemiadis, and K. Kyriakopoulos, *Functional anthropomorphism for human to robot motion mapping*, IEEE International Symposium on Robot and Human Interactive Communication, pp. 31–36, sept. 2012.
- [2] M.E. Rosheim, *Leonardo's programmable automaton. A reconstruction*.  
[http://www.anthrobot.com/press/article\\_leo\\_programmable.html](http://www.anthrobot.com/press/article_leo_programmable.html)
- [3] S. Capello, M.E. Moran, J. Belarmino, F. Firoozi, E. Kolios, M. Perrotti, *The da Vinci robot*, 23rd world congress on endourology. J Endourol 19(1).A133, 2005.
- [4] G. Wood, *Edison's eve. A magical history of the quest for mechanical life*, Anchor Books, New York, 2002.
- [5] J. Riskin, *The defecating duck or, the ambiguous origins of artificial life*, Critic Inq 29(4):599–633, 2003.
- [6] J. Vaucanson *Le Mechanisme du Xuteur automate* J. Guerin, Paris, 1738.
- [7] T. Standage, *The turk. The life and times of the famous eighteenth-century chess-playing machine*, Berkeley Books, New York, 2002
- [8] W. Pollard, U.S. Patent 2,286,571. *Position controlling apparatus*, Filed 22 April 1938 and issued 16 June, 1942.
- [9] H. Roselund, U.S. Patent 4,344,108. *Means for moving spray guns or other devices through predetermined paths*, Filed 17 August 1939 and issued 14 March, 1944.
- [10] D. Szondy, <http://www.davidszondy.com/future/robot/elektro1.htm>

- [11] G. Devol U.S. Patent 2,988,237. *Programmed article transfer*, Filed December 10,1954 and issued June 13, 1961.
- [12] J. Engelberger *Robots in service*, MIT Press, Cambridge, 1989.
- [13] J. Belarmino, M.E. Moran, F. Firoozi, S. Capello, E. Kolios, M. Perrotti *An oriental culture of robotics—the coming maelstrom*, Society of Urology and Engineering. J Endourol 19(7):A119, 906–907, 2005.
- [14] H. Sadjadian , H.D. Taghirad; *Neural Networks Approaches for Computing the Forward Kinematics of a Redundant Parallel Manipulator*, International Journal of Computational Intelligence Volume 2 Number 1
- [15] Hamid D. Taghirad *Parallel Robots: Mechanics and Control*
- [16] Shimizu, M., Kakuya, H., Yoon, W. K., Kitagaki, K., and Kosuge, K. (2008). *Analytical inverse kinematic computation for 7-DOF redundant manipulators with joint limits and its application to redundancy resolution*, Robotics, IEEE Transactions on, 24(5), 1131-1142.
- [17] Kreutz-Delgado, K., Long, M., and Seraji, H. (1992). *Kinematic analysis of 7- DOF manipulators*, The International journal of robotics research, 11(5), 469- 481.
- [18] Z.X Xu, J.T. Zhang, L. Yan and Z. Y. Wang *Parameterized inverse kinematics resolution method for a redundant space manipulator with link offset*, Yuhang Xuebao/Journal of Astronautics
- [19] B. Kenwright *Inverse Kinematics – Cyclic Coordinate Descent*
- [20] Charalambos Konnaris, Constantinos Gavriel, Andreas A.C. Thomik, A. Aldo Faisal *EthoHand: A dexterous robotic hand with ball-joint thumb enables complex in-hand object manipulation*, Biomedical Robotics and Biomechatronics (BioRob), 2016 6th IEEE International Conference.
- [21] <http://sssa.bioroboticsinstitute.it/projects/SmartHand>
- [22] M.C. CarrozzaC. SuppoF. SebastianiB. MassaF. VecchiR. LazariniM.R. CutkoskyP. Dario *The SPRING Hand: Development of a Self-Adaptive Prosthesis for Restoring Natural Grasping*, Autonomous Robots, March 2004, Volume 16, Issue 2, pp 125–141.

*Bibliography*

---

- [23] Michele Folgheraiter, Giuseppina Gini *Blackfingers: an Artificial Hand that Copies Human Hand in Structure, Size, and Functions.*, Humanoids 2000
- [24] Denavit Hartenberg convention [https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg\\_parameters](https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg_parameters)