# Virtual Human Hand: Grasping Strategy and Simulation

**Esteban Peña Pitarch**

Academic Dissertation for the Degree of PhD

PhD thesis submitted to

Universitat Politècnica de Catalunya (UPC)


October 2007

Director: Karim Abdel-Malek
Director, US Army Virtual Soldier Research (VSR) program
Director, Center for Computer Aided Design and the National Advanced Driving Simulator
Professor, Departments of Biomedical and Mechanical Engineering
The University of Iowa

Co-Director: Jingzhou Yang
Research Engineer, Center for Computed Aided Design
The University of Iowa

Co-Director: Anas Al Omar
Professor Department of Mechanical Engineering
Universitat Politécnica de Catalunya

*Dedicated to*

My wife and my sons, Joel and Eduard, for giving me your time.

# Virtual Human Hand: Grasping Strategy and Simulation

**Esteban Peña Pitarch**

Submitted for the degree of Doctor of Philosophy
October 2007

## Abstract

The human hand is the most complete tool, able to adapt to different surfaces and shapes and to touch and grasp. It is a direct connection between the exterior world and the brain. **I. Kant** (german philosopher) defined how the hand is an extension of the brain.

In this dissertation, we built a virtual human hand to simulate the human hand as realistically as possible. Based on the anatomy of the hand, we designed a hand with 25 degrees of freedom (DOF), with four of these degrees located in the carpometacarpal joint for the ring and small fingers. These four degrees permit the simulation of the human hand when it is arched. The thumb was designed with 5 DOF, the index and middle fingers have 4 DOF, in the metacarpophalangeal joint has two, and in the proximal interphalangeal joint and in the distal interphalangeal joint each have one. For the ring and small fingers, the 4 DOF are in similar joints plus as the four described above.

The Denavit-Hartenberg (D-H) method was applied because each finger was considered a ray, i.e., an open chain, with joints approximated to revolute joints. The D-H tables for each finger were shown, and the application

of forward and inverse kinematics permit the calculation of all angles for each joint $[q_1 \dots q_{25}]^T$.

Before grasping any object, our system checks the reachability of the object with workspace analysis.

Semi-intelligent task-oriented object grasping was implemented for making a decision once the user chooses the object and the task inherent to the object. The grasping algorithm was implemented in a virtual environment.

# Declaration

The work in this thesis is based on research carried out at The University of Iowa and Universitat Politècnica de Catalunya (UPC), the Departments of Mechanical Engineering and Enginyeria Minera i Recursos Naturals, and the Virtual Soldier Research program (VSR), USA. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

# Acknowledgements

Space is short, and there are so many people to acknowledge. Thanks to my advisors, Karim Abdel-Malek and Jingzhou Yang and Anas Al Omar, for your suggestions and comments.

# Contents

# List of Figures

# List of Tables

Chapter

# 1 Introduction

## 1.1 Motivation

Hand is the end-effector for the humans, using the terminology of robotics. Position and orientation, touch or grasp give to the human the contact with the exterior. It is a direct connection between the exterior world and the brain. **I. Kant** (german philosopher) defined how the hand is an extension of the brain.

Although robot hand grasp has been extensively studied [68], little attention has been paid to virtual human hand grasp. Virtual humans are an important tool for digital prototyping to save time and cost. Virtual human hand grasp is crucial for the virtual human to interact with the virtual world. When some joints have injuries (arthritis, carpal tunnel syndrome, etc.) the mobility of different joints is limited. A virtual hand model can help us to understand a suitable range of mobility when the virtual human hands reach for objects.

The principal motives for researching and designing a virtual human hand are:

- Performing research in a system with a high number of degrees of freedom.
- Simulating the human hand as realistically as possible and implementing it in a virtual environment, which leads to saving time and money.
- Provide new tools for building and implementing prosthetics hands. The hook in amputees will substituted by a sophisticated bionic hand.
- Develop a new hand model with 25 degrees of freedom.
- Provide a new algorithm and mathematical model for implementing and simulating the hand model.

## 1.2 State of Art

At this moment, hand research is a hot topic and there are a lot of groups working on the hand. There are different lines of research for the hand. We classify this research into five big groups, but the groups are related and research converges in the same way. The principal groups are robotics/mathematics, simulation/modelling, robotic hands, neural network, and medical studies.

### Robotics/Mathematics

The robot grasp has different configuration tools (e.g., parallel jaw gripper, Barrett hand, DLR hand, Shadow hand, MA-I hand), and each configuration tool has a special solution for grasping. Jia in this sense [57] proposed that parametric curves applied for grasping are antipodal points on

a curved shape. Application of stable forces was applied by Yu et al. [101], Howard and Kumar [55], Zhu et al. [108], Ponce and Faverjon [85], and Svinin et al. [93]. Kang and Ikeuchi [60] proposed programming the robot by direct human demonstration. Liu et al. [67] used the Barrett hand for grasping and manipulation. Kaneko and Tanie [59] presented fingers of a multifingered robot hand that touches an object. Yu et al. [101] modeled the hand with spherical sensors.

A grasping algorithm for grasping in robotics was presented by Bicchi et al. in [13,14] and [41] addresses the problem of planning grasps. To solve the grasping problem [22], Buss et al. proposed an optimization method for a multi-fingered robot hand [23] and Cheng and Orin [28] proposed a compact-dual linear programming (LP) method. Linear matrix inequality problems for grasp analysis was proposed by Han et al. in [50, 51].

Another area of research is a force-closure grasp [37, 107, 109] to address the problems of computing form-closure regions as well as computing optimal fingertip locations yielding $n$-finger form-closure grasps of three-dimensional (3-D) objects. Omata et al. [80, 81] analyzed the indeterminate grasp force in power grasps with a rigid-body model. Power grasp was studied by Zhang et al. [103–106].

MA-I hand was developed in Institut d'Organitzaci i Control de Sistemes Industrials (IOC), hosted in Universitat Polécnica de Catalunya (UPC) and was described and shown in several papers [29–31, 87, 92].

Feasible analysis for grasp was studied by Guan and Zhang [47] and Hagan et al. [46].

Dynamic manipulation/grasping controller multifingered robot hands was proposed by Nagai et al. [75, 76], and Nagashima et al. [77].

## Simulation/Modelling

Complete analysis and classification of the human hand was presented by Cutkosky [32] for the design of hands for manufacturing tasks. Albrecht et al. [7] presented a design hand without grasp simulation, and hand grasping can found the work by Aydin and Nakajima [10], based on a database. For simulating human fingers, Barbagli et al. [11] present a deformable model based on Hertzian and similar theories. Analysis of finger motion coordination in the hand manipulative and gestic acts for the fingers except the thumb was studied by Braido and Zhang [19].

For hand and muscle simulation, a sample surfaces method, was presented by Huang and Xiao [56]. Simulation for several robot hands was shown in the algorithm called GraspIt, built and presented in papers by Miller et al. [63, 68].

Another group that works in virtual hand has presented in several papers the DaibaHand [69–72]. This group models the hand based on an optical motion capture, that method is an expensive method that we use to validate our mathematical model.

For the virtual human called MAN3D, Savescu et al. [90] designed a virtual hand with 25 DOF include the wrist.

Based on visual recognition, Nolker and Ritter [79] reconstructed a hand posture in the articulated hand model with 20 DOF.

High degrees of freedom are very complex for controling and performing activities; to avoid these difficulties, Rijpkema and Girard [86] proposed some coupled movements of joints and reduced the number of degrees of freedom.

Based on three functions of grasping surfaces, namely the object-supporting,

pressing, and wrapping functions, Saito and Nagata [88] presented a study to assist robot designers in producing innovative robotic hand systems.

The Sharmes simulator [27] is the development of a highly realistic human hand and forearm model. The model contains 38 muscles and 24 DOF representing the joints of the system. Two of these DOF are for the wrist, two are for the arm, and 20 degrees of freedom are for the hand.

For a commercial CAD product, Davidoff and Freivalds [35] developed a virtual human for CATIA (Dassault System).

## Robotics Hand

In several works, Borst et al. [15–17] described and implemented grasp theories in the DLR Hand II; other contributions to this hand model were provided by Liu et al. [65] and Butterfab et al. [24].

Another hand model is the DIST-Hand from the University of Genova [25, 26, 45].

Fukaya et al. [43] designed the TUAT/Karlsruhe Humanoid Hand with a weight of 125 g, a length of 175 mm, and a width of 130 mm.

The AMADEUS Dextrous Subsea Hand: Design, Modeling, and Sensor Processing was described by Lane et al. [64].

There are other hands and several more contributions for these hands. For conciseness, we cite two robotic hands that have been built and implemented: the Shadow hand and the MA-I hand [92]; the latter was built at the Universitat Politècnica de Catalunya (UPC).

## Neuronal Network

Basically, in the context of hand model, these tools were applied in robotic hands. Early Liu et al. [66] proposed a neural network architecture

for robot hand control, and Hanes et al. [52] used neural network control of force distribution for power grasp.

Neuseirat and Abu-Zitar [6] used a special neural network to find minimal finger forces in the first stage. In the second stage, the results obtained in the first stage were used as a static mapping in training another neural network.

A system was presented by Bernardin et al. [12] that uses hand shape and contact-point information obtained from a data glove and tactile sensors to recognize continuous human-grasp sequences. The sensor fusion, grasp classification, and task segmentation are made by a hidden Markov model recognizer.

Bowers and Lumia [18] investigated intelligent grasping schemes using a fuzzy logic rule based expert system and used a vision system, robot arm, and mechanical hand to locate and manipulate unmodeled, randomly placed objects of various sizes and shapes.

In the work by Molina-Vilaplana and Lopez-Coronado [73] the proposal involves the design and development of a library of hand gestures consisting of motor primitives for finger pre-shaping of an anthropomorphic dextrous hand.

Moussa [74] ran experiments in a simulated environment using a 28-object database to show how the algorithm dynamically combined and expanded a mixture of neural networks to achieve the learning task.

### Medical Studies

In this section, we present state-of-the-art for medical experiments related to the human hand, because in order to design a virtual human hand,

we cannot avoid the anatomy of the hand. These experiments help us to design a more realistic hand model.

For the index finger movements, Darling and Cole [33, 34] determined whether dynamic interaction torques are significant for control of digit movements and investigated whether such torques are compensated by specific muscle activation patterns.

Measurement of the ability of humans to discriminate the orientations of cylinders passively contacting the fingerpad was discussed by Dodson et al. [38].

Dong et al. [39] theoretically demonstrated that the mechanical impedance (MI) in a hand power grip, as a measure of the biodynamic response of the system, can be divided into finger MI and palm MI.

Forces acting on the fingertips of 10 subjects were measured with three different handle diameters and five grip force levels by Freund et al. [42] for estimating the forces acting on the fingertips as functions of the total grip force, the diameter of the handle, and hand size.

To determine whether other digits move when normal humans attempt to move just one digit, Hager-Ross and Schieber [49] asked 10 right-handed subjects to move one finger at a time while they recorded the motion of all five digits simultaneously with both a video motion analysis system and an instrumented glove.

Kamojima et al. [58] proposed a system that identifies the position and orientation of hand bones from magnetic resonance (MR) volume images by registration of a bone model.

A study to determine a distribution of the forces produced by motor unit activity in the human flexor digitorum profundus was conducted by Kilbreath et al. [62].

Park et al. [82] studied the application of the Fibonacci sequence to the anatomy of the human hand, and the conclusion was that it is a relationship that is not supported mathematically.

A review of visual interpretation of hand gestures for human-computer interaction was presented by Pavlovic et al. [83].

For study of muscles, we found several papers Valero-Cuevas and colleagues, [84, 96–98].

In Yokogawa and Hara [100] investigated whether distribution patterns of the maximum fingertip force in all directions from 0 deg to 360 deg around the index fingertip was the same among subjects.

## 1.3 Contributions

Above we presented a state-of-the-art in hand research in all of ways. The contributions of this research are:

- A 25-DOF hand model; compared with other models, our hand model has two DOFs in the base of the ring finger and two more in the base of small finger in the carpometacarpal (CMC) joint. These four new degrees contributed to simulation of the palm arch, as seen in real like the humans.

- Adaptation and application of tools from robotics for forward and inverse kinematics. Application of forward and inverse kinematics is for determining the position of the fingers with respect to the object.

- Creation a virtual environment for grasping with precision, power, pinch, push, pull, or touch. Based in geometric finger position and posture prediction.

- A new tool for designing new hand models in robotics. Our virtual hand provides each angle of joints for configuring the hand shape.
- Connection with the upper body of the virtual human $Santos^{TM}$ created at The University of Iowa.
- A semi-intelligent grasping framework based on object-task-oriented grasping. As a function of task inherent to the object, shape, etc., the virtual hand can do these tasks. It can also grasp among objects.

## 1.4 Outline of the Thesis

This dissertation is divided into seven chapters and references. The chapters are summarized as follows:

*Chapter 2:* In this chapter, we present a few concepts of hand anatomy. The objective of these concepts is to focus and justify the number of degrees of freedom, the mobility of each joint for deciding the number of degrees of freedom for each joint, the plane of movement of each finger, and the range of motion for the joints of the fingers.

*Chapter 3:* A hand model with 25 DOF is shown in this chapter, the Denavit-Hartenberg method [36] is applied and the formulation for forward and inverse kinematics is presented. Some results are shown in the last part of this chapter.

*Chapter 4:* Representation of workspace for each finger based on the Jacobian rank deficiency method is introduced in this chapter and some examples are discussed. The objective of this chapter is that, when the virtual human makes a decision to grasp any object, it is able to validate the reachability of this object on the first try, i.e., the object is in the workspace of each finger that grasps the object.

*Chapter 5:* A task-oriented object grasping method is proposed in this chapter. This method permits the virtual human to grasp objects semi-intelligently when these objects enter the virtual environment. We define all the parameters and the relationship between the object and the virtual hand. A new mathematical model based on the support vector machine (SVM)is presented, and an algorithm for grasping strategy is implemented.

*Chapter 6:* This chapter is dedicated to implementing with examples the theory described in Chapter 5. Several examples are presented and discussed.

*Chapter 7:* In this chapter, we present conclusions and propose future work for this research.

Chapter

# 2 Anatomy of the Hand

## 2.1 Introduction

The objective of this chapter is to describe the anatomy of the hand; this is the basis for designing our hand model. We do not attempt to provide an exhaustive description for the hand similar at the level of the physicians, but we present the knowledge necessary for building our virtual hand model. First, we describe the physiology of the hand and show the muscles and connections with bones. The muscles are beyond the scope of this dissertation. For the articular system, we use tables and short descriptions of the connections, similar way to that used for the bones. We present the range of motion for the wrist and the joints of the hand. Principal connections among muscles and bones are shown in two figures in the last pages of this chapter.

## 2.2 Physiology of Hand

The human hand is fascinating and has been studied for many years; consequently, there are several books. The physiology of the hand not is the principal topic of this dissertation, but we need a little bit of knowledge in order to understand the next chapters. In this chapter, following the references Kapandji [61], and Tubiana et al. [94, 95], we show the muscles and bones that act in the hand. Of course, in the hands we have veins, sinovial fluid, etc., but, this does not affect our study. The hand is the terminal part of the forelimb in primates. The human hand consists of the wrist, palm, four fingers, and thumb. In humans and other primates, the thumb is opposable, i.e., it can be moved into a position opposite to the other four digits. Opposable thumbs make possible precise movements such as grasping small objects. In vertebrates other than humans, the primary function of the hand is locomotion; the human hand, due to the evolutionary development of bipedalism, is freed for manipulative tasks. There are 27 bones in the human hand. The wrist, which joins the hand to the forearm, contains eight cubelike bones arranged in two rows of four bones each. The metacarpus, or palm, is composed of five long metacarpal bones. Fourteen phalangeal bones constitute the four fingers and thumb (three in each finger, two in the thumb). Ligaments interconnect the bones of the hand. The bones of the digits are anchored to muscles in the hand and to muscles in the arms and shoulders, through connections to tendons, permitting a wide range of movements. Among humans, the undersides of the fingers and palms have distinctive ridges, which improve grip and can be used as identification marks.

## 2.3 The Articular System

This section provides a brief description of the articular system of the human hand. This section is not an anatomy book, but we need some knowledge about the articular system. For designing and using our virtual hand and how to use, the most relevant part is the structure of the hand, i.e., the bones, because we apply the forward and inverse kinematics to the virtual hand. First, we begin with the description of the bones and their relatives movements, then we introduce the range of motions for each finger, and finally we provide a brief description of muscles.

### Joint Unions

In this section, we describe the joint unions. There are joint unions for the wrist and joint unions for the rays of the hand. Figure 2.1 shows the bones of wrist. They are distributed in two rows of small bones; each row has four bones, and the articulations among bones as follows:

The scaphoid has articulations with the scapholunate, lunotriquetal, capitolunate, and radiolunate bones, and are attached to the abductor pollicis brevis muscle.

The lunate has articulations with the scapholunate, lunotriquetal, and capitolunate bones and the radiolunate joint.

The triquetum has articulations with the lunotriquetral, pisotriquetral, and triquetriohamate bones.

The trapezoid has articulations with the scaphotrapezoidal, trapeziotrapezoidal, and index carpometacarpal bones, and is attached to the abductor pollicis muscle.

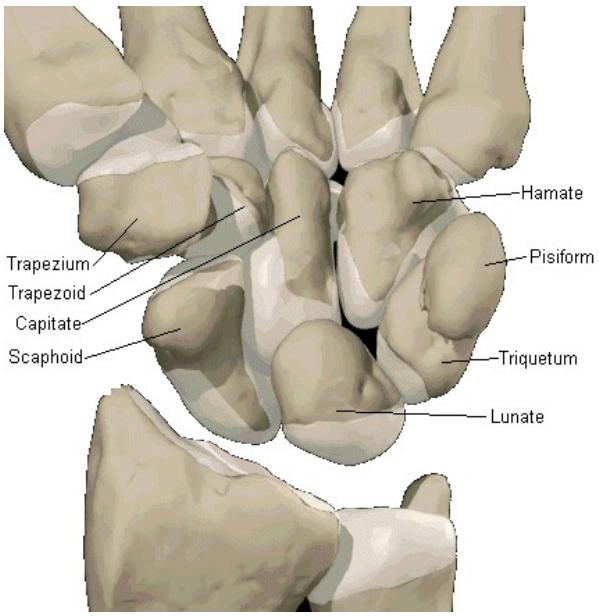


Figure 2.1: Bones of the wrist. Dorsal view

The trapezium has articulations with the scaphotrapezoidal, trapezio-trapezoidal, and thumb basal bones, and is attached to a five muscles.

The capitate has articulations with the middle carpometacarpal, and midcarpal bones, and is attached to the abductor pollicis muscle.

The hamate has articulations with the capitohamate, triquetriohamate, ring carpometacarpal, and small carpometacarpal, and is attached to the opponens digiti minimi and flexor carpi ulnaris muscles.

The pisiform has articulations with the pisotriquetral bones and is attached to the opponens carpi ulnaris, and abductor digiti minimi muscles.

The relative movements among the wrist bones have been studied by Neu et al. [78] and Sonenbluma et al. [91], and they considered the motion of the scaphotrapezio-trapezoidal (STT) joint. However, we consider this movement as shared in common with the different rays in each finger.

A similar process as described for the wrist bones is followed for the

hand bones. Table 2.1 depicts bones and articulations, where CMC is the carpometacarpal joint, MCP mecapophalangeal joint, and MC is the metacarpal joint. Figure 2.2 shows the skeleton of the hand with all the bones. We will use the subscript $I$ for the thumb, $II$ for the index, $III$ for the middle, $IV$ for the ring, and $V$ for the small finger.

| Hand bones. Metacarpals | |
|---|---|
| Bone | Articulations |
| Thumb CMC | Thumb CMC Joint and Thumb MCP Joint |
| Index MC | Index CMC Joint and Index MCP Joint |
| Middle MC | Middle CMC Joint and Middle MCP Joint |
| Ring MC | Ring CMC Joint and Ring MCP Joint |
| Small MC | Small CMC Joint and Small MCP Joint |

Table 2.1: Metacarpal bones

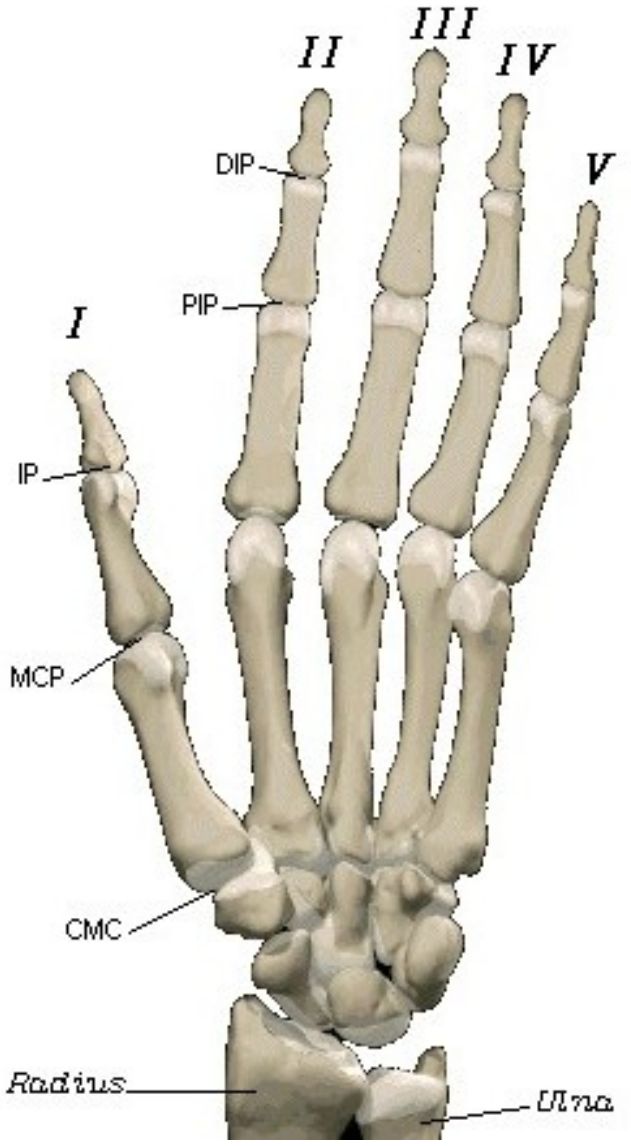


Figure 2.2: The hand, bones and joints; dorsal view

There are three phalanges: proximal, middle, and distal. The next table shows each phalanx with its corresponding articulations.

| Hand Bones, Phalanges | |
|---|---|
| Phalanx | Articulations |
| Thumb Proximal | Thumb MCP Joint and Thumb IP Joint |
| Index Proximal | Index MCP Joint and Index Proximal IP Joint |
| Middle Proximal | Middle MCP Joint and Middle Proximal IP Joint |
| Ring Proximal | Ring Proximal IP Joint and Ring MCP Joint |
| Small Proximal | Small MCP Joint and Small Proximal IP Joint |
| Index Middle | Index Proximal IP Joint (PIP) and Index Distal IP Joint (DIP) |
| Middle Middle | Middle Proximal IP Joint(PIP) and Middle Distal IP Joint(DIP) |
| Ring Middle | Ring Proximal IP Joint(PIP) and Ring Distal IP Joint(DIP) |
| Small Middle | Small Proximal IP Joint(PIP) and Small Distal IP Joint(DIP) |
| Thumb Distal | Thumb IP Joint |
| Index Distal | Index Distal IP Joint(DIP) |
| Middle Distal | Middle Distal IP Joint(DIP) |
| Ring Distal | Ring Distal IP Joint(DIP) |

| | |
|---|---|
| Small Distal | Small Distal IP Joint(DIP) |

Table 2.2: Metacarpal bones and phalanges

IP is the interphalangeal joint, DIP is the distal interphalangeal joint, and PIP is the proximal interphalangeal joint.

## Normal Range of Motion Reference Values

Movements of wrist for flexion/extension (F/E) and are shown in Figure 2.3, and the Figure 2.4 shows the radial/ulnar deviation for the wrist. The range of motion for these movements are shown in Table 2.3; these movements and range of motions come from the literature [21, 89].

| Joint | Action | Normal |
|---|---|---|
| Wrist | Extension/Flexion | 70/75 |
| Wrist | Radial/Ulnar | 20/35 |

Table 2.3: Range of motion for the wrist(in degrees)

Figure 2.5 shows the definition of movements for each finger in terms of F/E and abduction/adduction (Ab/Ad) and the visual concept of hyper-extension, which that is, when the finger overpasses the line of the finger, and the finger is aligned with arm.

Tables 2.4 and 2.5 show the range of motion for each finger, where H means hyper-extension. Not all the fingers have these movements, and these tables do not cover all the populations. Figure 2.6 shows the movements for the thumb; for our model, the movement of opposition and retropposition are simulated by the sum of the movements of the CMC and MCP joints.

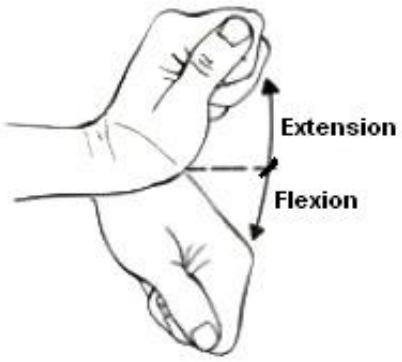


Figure 2.3: Flexion/extension of the wrist

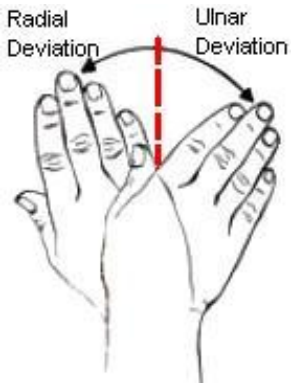


Figure 2.4: Radial/ulnar deviation of the wrist

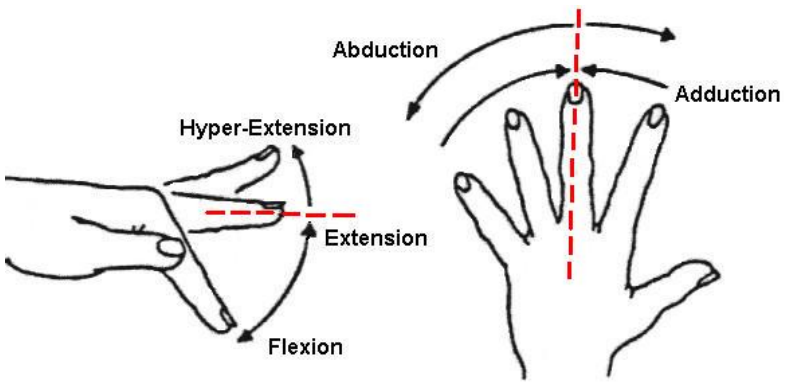


Figure 2.5: Abduction/adduction(Ab/Ad), flexion/extension (F/E) for one finger

| Thumb | Action | Normal |
|---|---|---|
| CMC | Adduction/Abduction | 0(contact)/60 |
| CMC | Extension/Flexion | 25/35 |
| MCP | Extension/Flexion | 10H/55 |
| MCP | Adduction/Abduction | 0/60 |
| IP | Extension/Flexion | 15H/80 |

Table 2.4: Range of movements for the joints of the thumb(in degrees)

| Finger | MCP(E/F) | PIP(E/F) | DIP(E/F) | MCP(Ab/Ad) |
|---|---|---|---|---|
| Index | 0/80 | 0/100 | 10H/90 | 13/42 |
| Middle | 0/80 | 0/100 | 10H/90 | 8/35 |
| Ring | 0/80 | 0/100 | 20H/90 | 14/20 |
| Small | 0/80 | 0/100 | 30H/90 | 19/33 |

Table 2.5: Range of movements for the joints of the fingers; H is hyper-extension(in degrees)

## Muscles

Table 2.6 shows the type of muscles and their actions using the following abbreviations: FDP: flexor digitorum profundis, AMAd: adductor muscles adductor, AMAb: adductor muscles abduction, FRP: flexor retinaculum posteriorly, FM: flexor muscles, LM: liumbrical muscles.

Figures 2.7 and 2.8 show an arm and hand skeleton and the attached position for the muscles schematically. In this thesis we do not use the muscles, because when we study movement, it is under the point of view of the kinematics, and are only interested in how the bones move, not

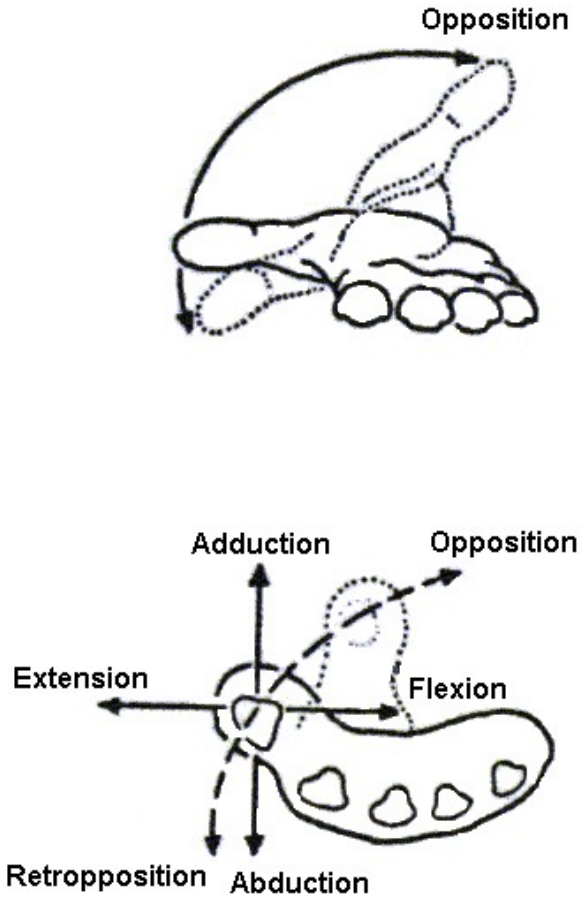


Figure 2.6: Abduction/adduction, flexion/extension for the thumb

| Muscle | Action fingers |
|---|---|
| FDP | Flexes the distal joints of the fingers |
| AMAd | Moves a digit toward the axis of a limb |
| AMAb | Moves a digit away from the axis of a limb |
| FRP | Flexor retinaculum in the wrist and palm of the hand |
| FM | Decreases the angle between two bones |
| LM | Each flexes the corresponding digital joint to extend the finger |

Table 2.6: Types of muscles

what cause of this causes the movements. For more information about the muscles, there are an excellent references; see, for example, Kapandji [61] and Tubiana et al. [94, 95].

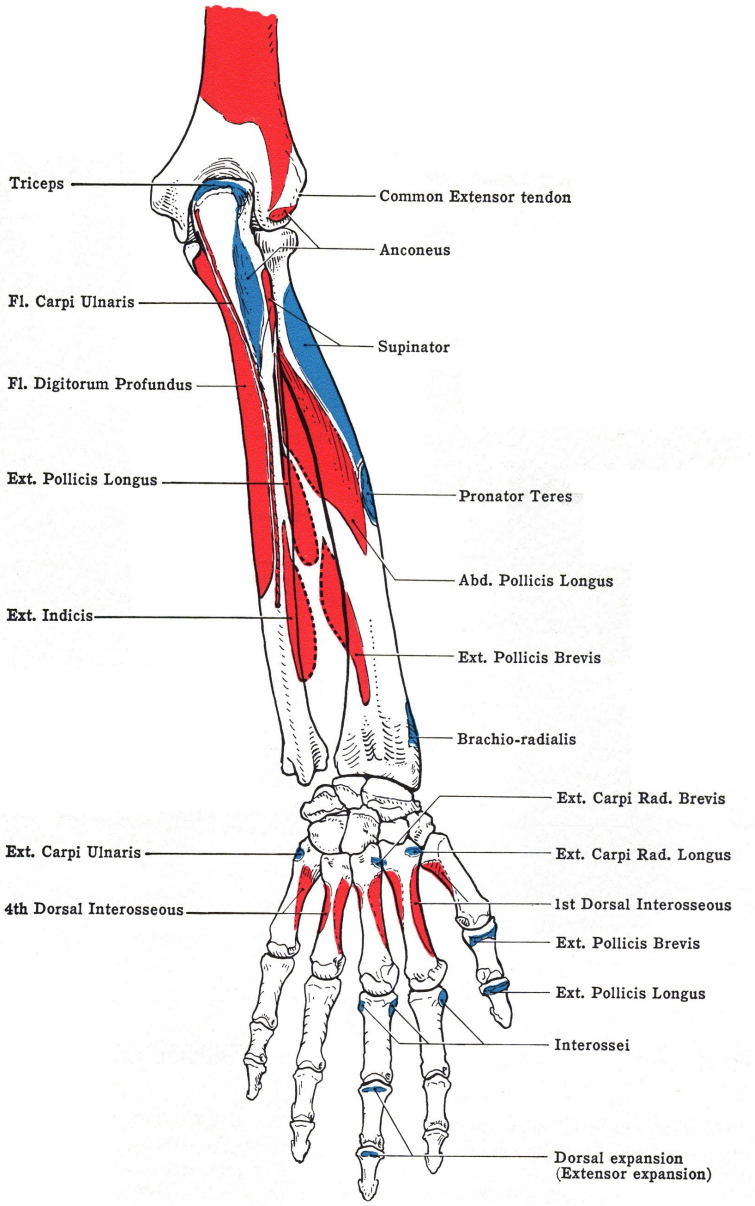


Figure 2.7: The muscles and their location; posterior view

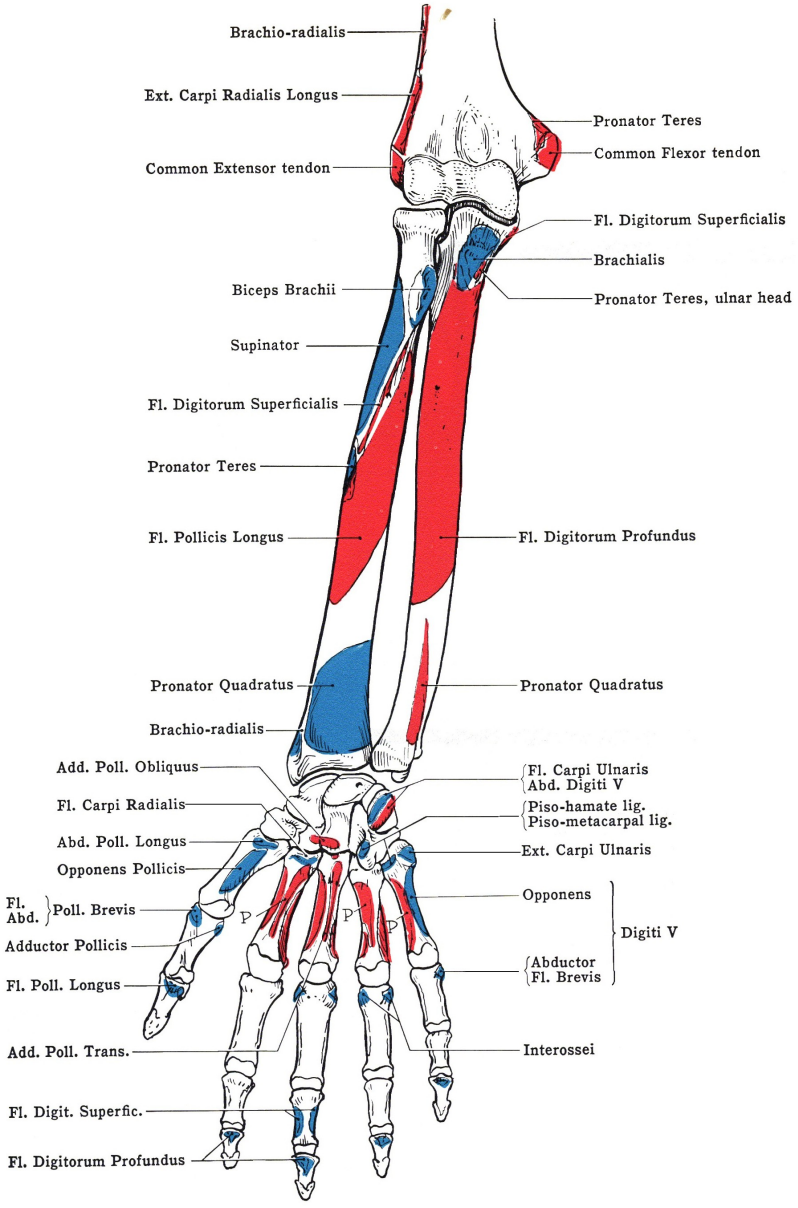


Figure 2.8: The muscles and their location; anterior view

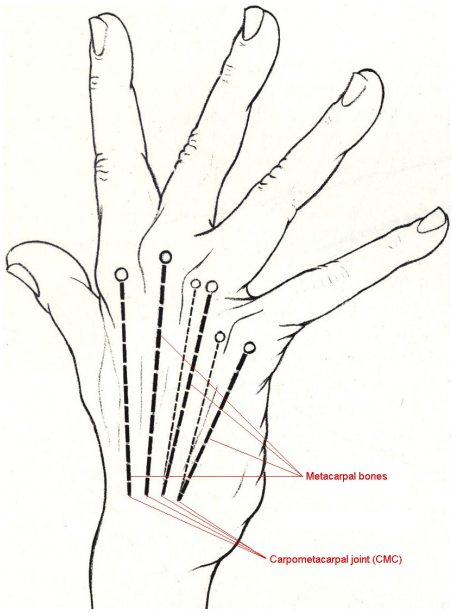


Figure 2.9: Palm arch

## 2.4 Palm Arch

From the literature [94], we can see that the human hand arches in the palm. For our next considerations, in Figure 2.9, the metacarpal bones for the index and the middle fingers are constant when the hand arches the palm, but for the ring and small fingers, the metacarpal bones rotate about their respecitve carpometacarpal joints (CMC). This movement decomposes into two, one in the direction of F/E and the other in the direction of Ab/Ad.

## 2.5 Conclusions

This chapter is the basis for modeling our virtual hand. The only intention for this chapter is to provide the hand anatomy. Because this not

is a book on hand anatomy, the hand anatomy is present only in tables, talking only about the joints, muscles, connections, and movements. Based on the references, we defined the movements of flexion/extension (F/E), abduction/adduction (Ab/Ad), and showed how the palm arch. Based on this arch, we add four more DOFs than other hand models. The novel 25 DOFs are justified for simulating as realistically as possible the movement of the virtual hand. In the next chapter, we apply and develop this model.

Chapter

# 3 Kinematic Hand Model

This chapter presents a kinematic hand model. The Denavit-Hartenberg (D-H) convention [36] becomes necessary because we have five open-loop kinematics structures and a high number of degrees of freedom (DOF). We show a novel model with 25 DOF; this number of DOF is justified in the hand anthropometry section. The next section presents a parametric model for each finger. The transformation from local to global coordinates is analyzed, and the chapter ends with a discussion about controlling the hand kinematics formulation for manipulating our model.

## 3.1 D-H Representation

Figure 3.1 shows the hand with the 25 DOF for parametrization using the D-H representation algorithm described in the next section. For application of D-H parameters, we begin with point O. It has a fixed base, is located close to the pulse, and all the movements for the five fingers are referred to this base. Figure 3.1 shows the associated movements such as *Abduction/Adduction (Ab/Ad)* and *Flexion/Extension (F/E)*. These pairs

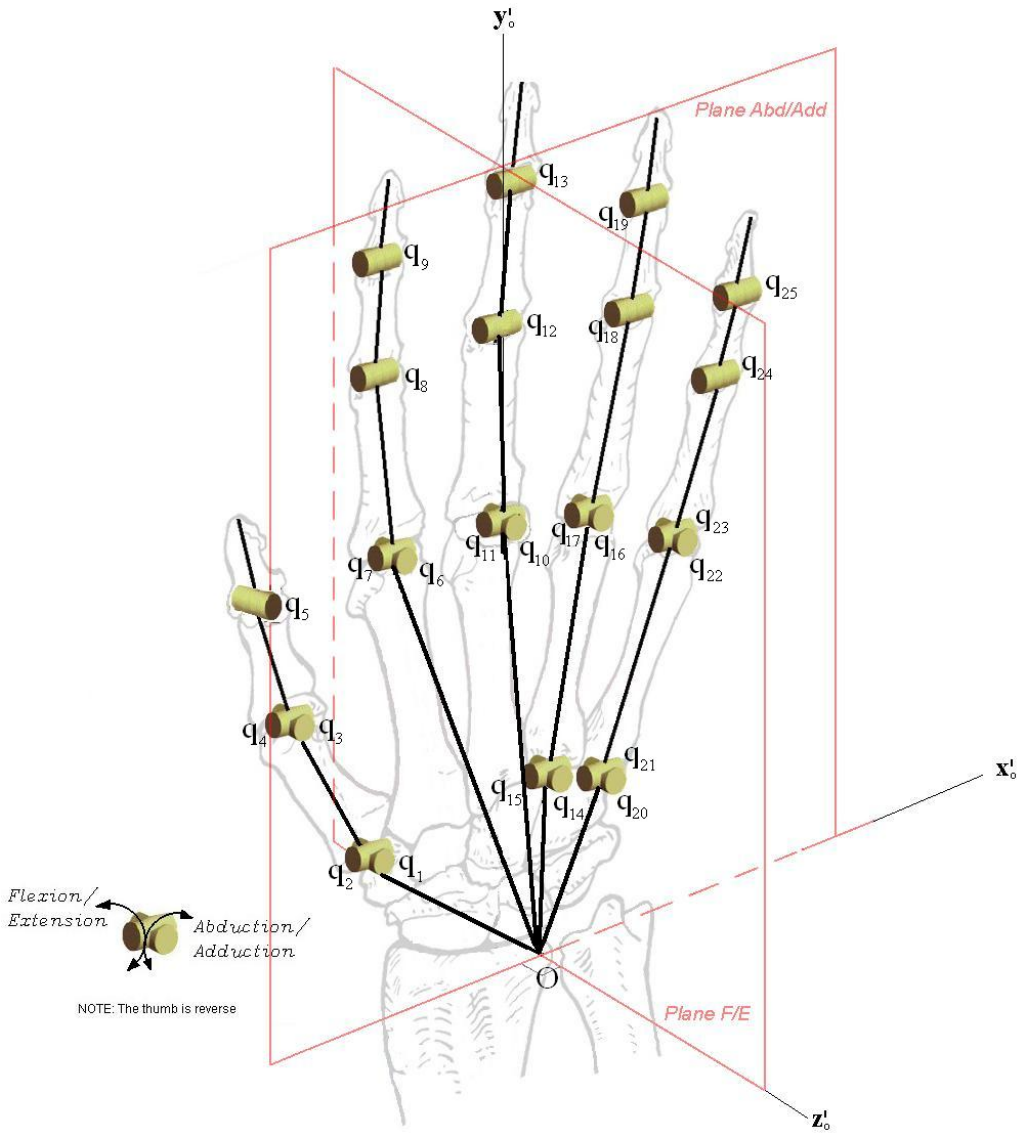


Figure 3.1: The 25-DOF hand. Posterior view of right hand.

of movements are in the same plane for the four fingers (II, III, IV and V) shows in Figure 3.1, but for the thumb the planes of movements for *Ab/Ad* and *F/E* are in planes perpendicular to the other fingers.

Also note that, in the picture the DOF are located in each finger, with 5 DOF for the thumb, 4 DOF for the index and middle fingers, because we considered the bottom of metacarpophalangeal joint fixed in the wrist, and 6 DOF for the ring and small fingers. The two new DOF for the last fingers appear because we considered the movement in the bottom of metacarpophalangeal joint with respect to the wrist to permit the hand to show the shape of the arc in the palm [61, 94, 95].

## Algorithm: D-H Representation

This is the algorithm proposed by Angeles [9] and adapted for the hand model.

1. Number the joints from 1 to $n$ starting with the base and ending with the tip yaw, pitch, and roll, in that order.

2. Assign a right-handed orthonormal coordinate frame $L_0$ to the finger base, making sure that $z^0$ aligns with the axis of joint 1. Set $k = 1$.

3. Align $z^k$ with the axis of joint $k + 1$.

4. Locate the origin of $L_k$ at the intersection of the $z^k$ and $z^{k-1}$ axes. If they do not intersect, use the intersection of $z^k$ with a common normal between $z^k$ and $z^{k-1}$.

5. Select $x^k$ to be orthogonal to both $z^k$ and $z^{k-1}$. If $z^k$ and $z^{k-1}$ are parallel, point $x^k$ away from $z^{k-1}$.

6. Select $y^k$ to form a right-handed orthonormal coordinate frame $L_k$. This axis normally is not shown to avoid cluttering the drawing.

7. Set $k = k + 1$. If $k < n$, go to step 3; otherwise, continue.

8. Set the origin of $L_n$ at the tip. Align $z^n$ with the approach vector, $y^n$ with the sliding vector, and $x^n$ with the normal vector of the phalanx. Set $k = 1$.

9. Locate point $b^k$ at the intersection of the $x^k$ and $z^{k-1}$ axes. If they do not intersect, use the intersection of $x^k$ with a common normal between $x^k$ and $z^{k-1}$.

10. Compute $\theta_k$ as the angle of rotation from $x^{k-1}$ to $x^k$ measured about $z^{k-1}$.

11. Compute $d_k$ as the distance from the origin of frame $L_{k-1}$ to point $b^k$ measured along $z^{k-1}$.

12. Compute $a_k$ as the distance from point $b^k$ to the origin of frame $L_k$ measured along $x^k$.

13. Compute $\alpha_k$ as the angle of rotation from $z^{k-1}$ to $z^k$ measured about $x^k$.

14. Set $k = k + 1$. If $k \leq n$, go to step 9; otherwise, stop.

## 3.2 Hand Anthropometry

The definition of anthropometry from a dictionary (Oxford English Dictionary)is a branch of anthropology dealing with measurement of the human body to determine the differences in individuals, groups, etc. Following this

definition, in this section we use the parametric lengths. Figure 3.2 shows the two parameters that we use to know the bones' length. These parameter's hand length ($HL$) and hand breadth ($HB$) are different for every person.

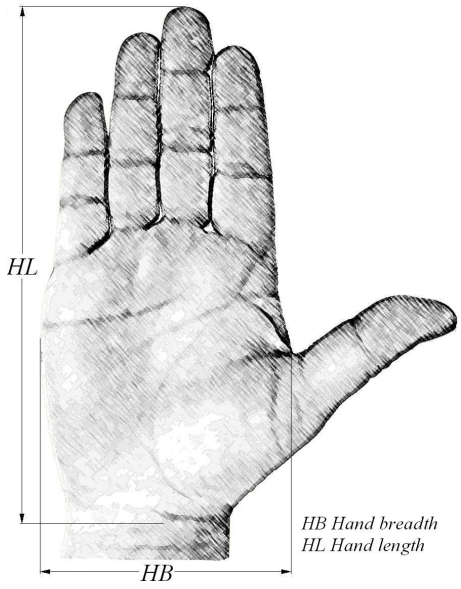


Figure 3.2: Parametric length for a hand

## 3.3 Parametric Model for Each Digit

Figure 3.3 shows the dimensions used for parametric dates for the four fingers $i = II, III, IV, V$, where the notation $II$ is for the index finger, $III$ is for the middle finger, $IV$ is for the ring finger, and $V$ for the small finger. Subindex $-0$ is the length from the global coordinates; these coordinates are located close to the pulse, in the wrist. Subindex $-1$ is the length of the metacarpal bones. Subindex $-2$ is the length of the proximal phalanx bones. Subindex $-3$ is the length of the medial phalanx bones. Subindex $-4$ is the length of the distal phalanx bones.

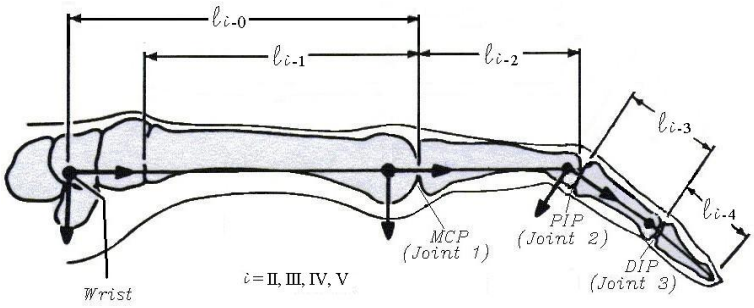


Figure 3.3: Parametric length for a finger

Figure 3.4 shows the same parametric length for the thumb, where $i = I$ and the lengths $1, 2, 3$ are for the thumb phalanx.

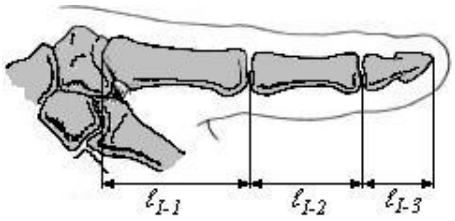


Figure 3.4: Parametric length for a thumb

Figures 3.3 and 3.4 are adapted from Buchcholz et al. [21] and Sancho-Bru [89].

Table 3.1 shows the lengths for the metacarpal bones; this length is a function of the parametric length of the hand *HL* and *HB*. From the work of Buchcholz et al. [21] and Sancho-Bru [89], we built this table. Other authors propose similar values for the length's, for more information, see the work by Brand and Hollister [20].

| | Metacarpal bones | |
|---|---|---|
| Thumb | $0.251 * HL$ | $\ell_{I-1}$ |
| Index | $\sqrt{(0.374 * HL)^2 + (0.126 * HB)^2}$ | $\ell_{II-1}$ |
| Middle | $0.373 * HL$ | $\ell_{III-1}$ |
| Ring | $\sqrt{(0.336 * HL)^2 + (0.077 * HB)^2}$ | $\ell_{IV-1}$ |
| Small | $\sqrt{(0.295 * HL)^2 + (0.179 * HB)^2}$ | $\ell_{V-1}$ |

Table 3.1: Lengths for the metacarpal bones

Table 3.2 shows the lengths for the phalangeal bones.

| | Proximal | | Middle | | Distal | |
|---|---|---|---|---|---|---|
| Thumb | $0.196 * HL$ | $\ell_{I-2}$ | – | – | $0.158 * HL$ | $\ell_{I-3}$ |
| Index | $0.265 * HL$ | $\ell_{II-2}$ | $0.143 * HL$ | $\ell_{II-3}$ | $0.097 * HL$ | $\ell_{II-4}$ |
| Middle | $0.277 * HL$ | $\ell_{III-2}$ | $0.170 * HL$ | $\ell_{III-3}$ | $0.108 * HL$ | $\ell_{III-4}$ |
| Ring | $0.259 * HL$ | $\ell_{IV-2}$ | $0.165 * HL$ | $\ell_{IV-3}$ | $0.107 * HL$ | $\ell_{IV-4}$ |
| Small | $0.206 * HL$ | $\ell_{V-2}$ | $0.117 * HL$ | $\ell_{V-3}$ | $0.093 * HL$ | $\ell_{V-4}$ |

Table 3.2: Lengths for the phalangeal bones

The parametric lengths are used in two ways: one is for the lengths applied in the D-H method (i.e., $a_i$,$\alpha_i$, $d_i$, and $\theta_i$), and the other is to transform local coordinates to global coordinates.

Table 3.3 shows the angles in degrees for the natural position of the hand (i.e., for this position all the muscles in the hand do not work). We need this table in the next chapter. When we evaluate the torque and the comfort or discomfort, we work from the neutral position to the actual position.

| | CMC | CMC | MCP | MCP | PIP | DIP |
|---|---|---|---|---|---|---|
| | Flex. | Abd. | Flex. | Abd. | Flex. | Flex. |
| Thumb | 0 | 0 | 30 | 0 | 30 | – |
| Index | – | – | 30 | 0 | 30 | 10 |
| Middle | – | – | 30 | 0 | 30 | 10 |
| Ring | 2 | – | 30 | 0 | 30 | 10 |
| Small | 5 | – | 30 | 0 | 30 | 10 |

Table 3.3: Angles for the natural position of the hand (degrees)

Once we know all the parametric lengths for each finger and follow the algorithm for the D-H representation, we can build the open-loop chain and the D-H table for each finger.

The transformation matrix ${}^{i-1}A_i$, where subscript $i = 1 \dots 25$, is shown below:

$$ {}^{i-1}A_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

The following figures and tables show are the D-H model for each finger and a table with the D-H parameters. To create this figure, we follow all points described in the Section 3.1.1.

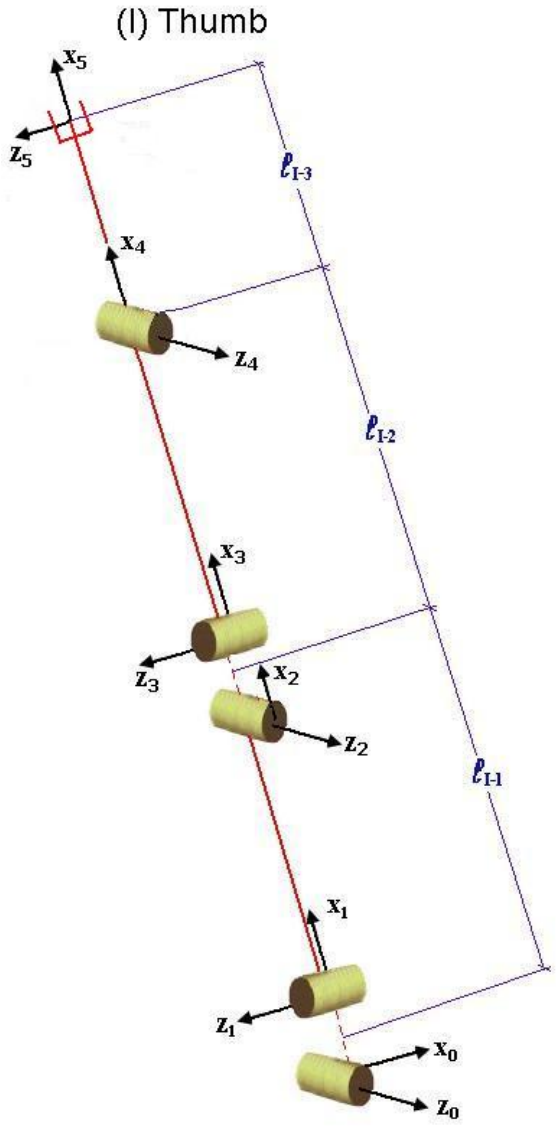


Figure 3.5: Thumb D-H model

| | $\theta_i$ | $d_i$ | $a_i$ | $\alpha_i$ |
|---|---|---|---|---|
| 1 | $q_1 + \frac{\pi}{2}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 2 | $q_2$ | 0 | $l_{I-1}$ | $\frac{\pi}{2}$ |
| 3 | $q_3$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 4 | $q_4$ | 0 | $l_{I-2}$ | $\frac{\pi}{2}$ |
| 5 | $q_5$ | 0 | $l_{I-3}$ | $\frac{-\pi}{2}$ |

Figure 3.6: Thumb D-H Table

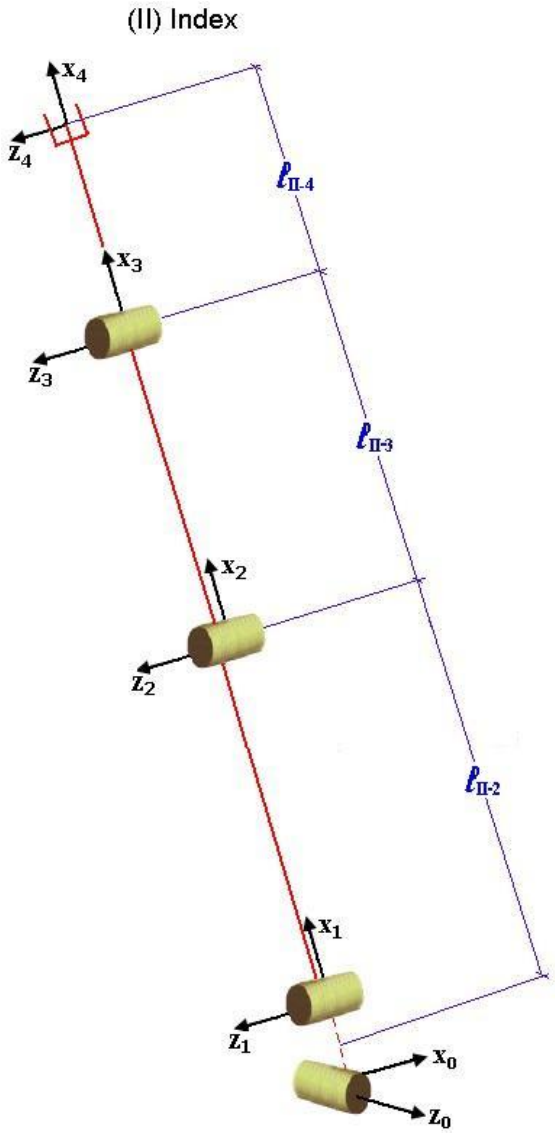


Figure 3.7: Index D-H model

| | $\theta_i$ | $d_i$ | $a_i$ | $\alpha_i$ |
|---|---|---|---|---|
| 6 | $q_6 + \frac{\pi}{2}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 7 | $q_7$ | 0 | $l_{II-2}$ | 0 |
| 8 | $q_8$ | 0 | $l_{II-3}$ | 0 |
| 9 | $q_9$ | 0 | $l_{II-4}$ | 0 |

Figure 3.8: Index D-H table

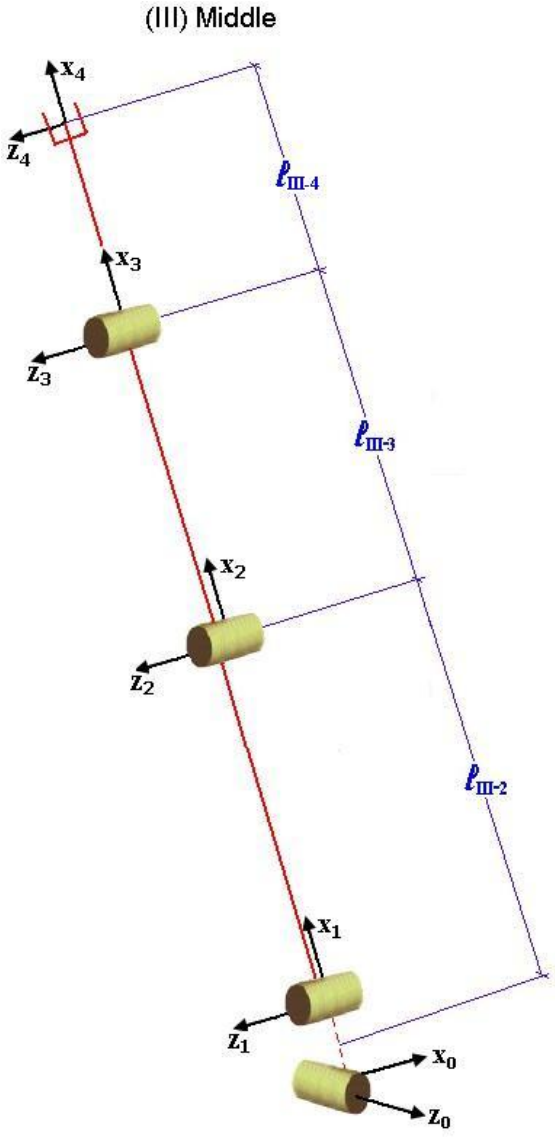


Figure 3.9: Middle D-H model

| | $\theta_i$ | $d_i$ | $a_i$ | $\alpha_i$ |
|---|---|---|---|---|
| 10 | $q_{10} + \frac{\pi}{2}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 11 | $q_{11}$ | 0 | $l_{III-2}$ | 0 |
| 12 | $q_{12}$ | 0 | $l_{III-3}$ | 0 |
| 13 | $q_{13}$ | 0 | $l_{III-4}$ | 0 |

Figure 3.10: Middle D-H Table

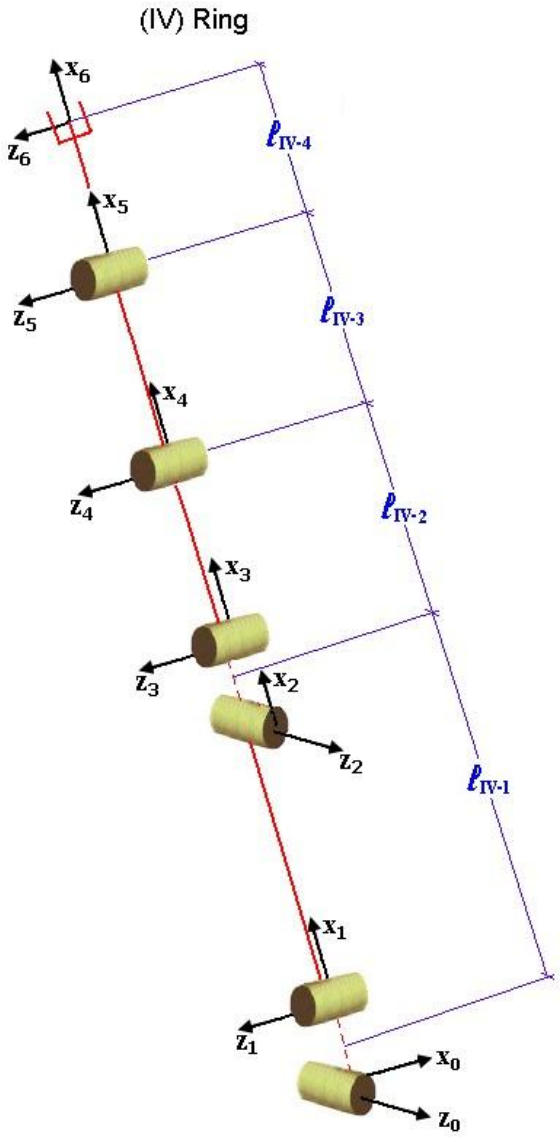


Figure 3.11: Ring D-H model

| | $\theta_i$ | $d_i$ | $a_i$ | $\alpha_i$ |
|---|---|---|---|---|
| 14 | $q_{14} + \frac{\pi}{2}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 15 | $q_{15}$ | 0 | $l_{IV-1}$ | $\frac{\pi}{2}$ |
| 16 | $q_{16}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 17 | $q_{17}$ | 0 | $l_{IV-2}$ | 0 |
| 18 | $q_{18}$ | 0 | $l_{IV-3}$ | 0 |
| 19 | $q_{19}$ | 0 | $l_{IV-4}$ | 0 |

Figure 3.12: Ring D-H Table

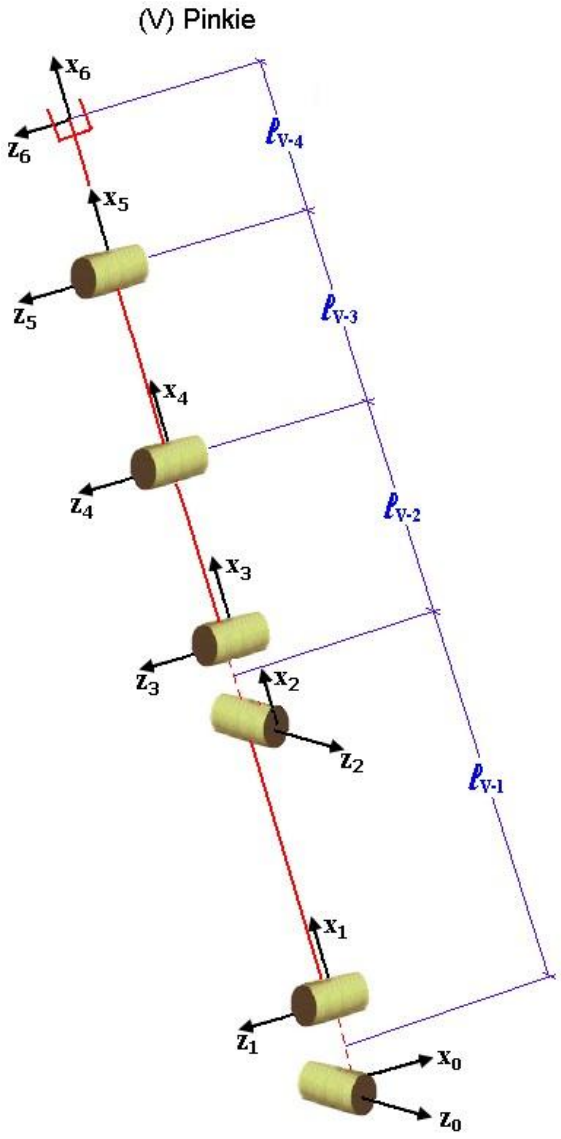


Figure 3.13: Small D-H model

| | $\theta_i$ | $d_i$ | $a_i$ | $\alpha_i$ |
|---|---|---|---|---|
| 20 | $q_{20} + \frac{\pi}{2}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 21 | $q_{21}$ | 0 | $l_{V-1}$ | $\frac{\pi}{2}$ |
| 22 | $q_{22}$ | 0 | 0 | $\frac{-\pi}{2}$ |
| 23 | $q_{23}$ | 0 | $l_{V-2}$ | 0 |
| 24 | $q_{24}$ | 0 | $l_{V-3}$ | 0 |
| 25 | $q_{25}$ | 0 | $l_{V-4}$ | 0 |

Figure 3.14: Small D-H Table

## 3.4 Transformation of Local Coordinates to Global Coordinates

In the last section, we presented the open-loop chain and D-H parameters for every finger. In this section, we show how we can transform the local coordinates to global coordinates. The base of the coordinates is located close to the pulse, as depicted in 3.15.

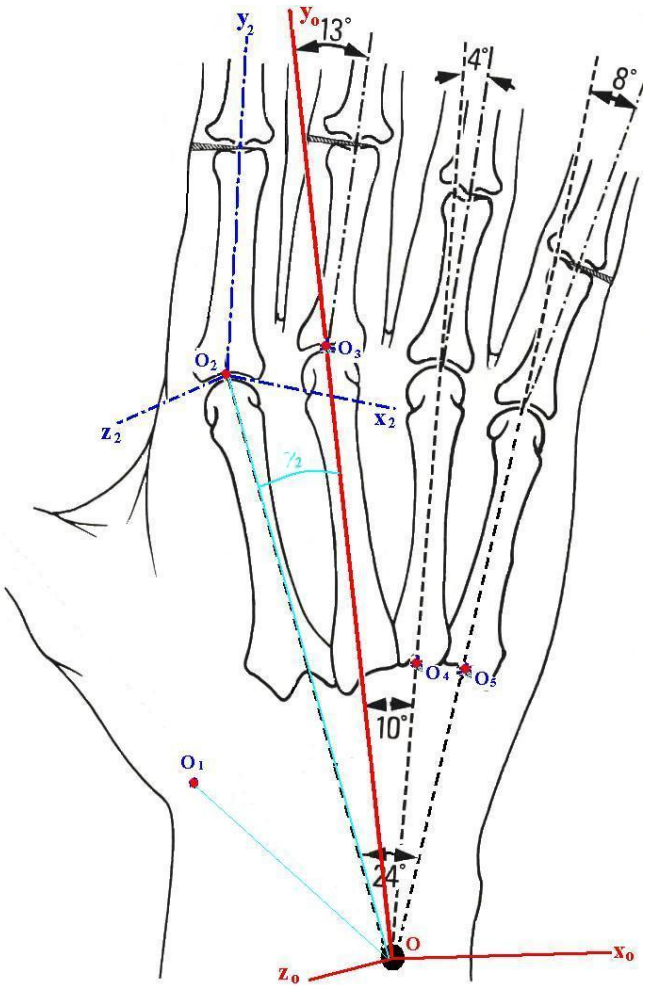


Figure 3.15: Global coordinates

In this figure we can observe the angle $\gamma_2$, which is the angle of line $OO_2$ with the axis $y_o$ (axis to global coordinates). The point $O_j$ $(j = 1 \ldots 5)$ is the point of reference at which the local coordinates for each finger begin. As for the angle $\theta_j$ $(j = 1 \ldots 5)$, this is the angle from line $OO_j$ to the global axis $y_o$.

To change the coordinate system from local to global, the following steps are necessary:

1. Put the palm in the position of pronation (in Fig. 3.15 we consider the right hand (dorsal view)).

2. Draw the axis $y_o$ located in the radius bone and positive to a metacarpal bone.

3. Draw the axis $x_o$ positive to the direction of the ulnar bone and perpendicular to $y_o$.

4. Draw the axis $z_o$ with the right-hand rule and observe that it is perpendicular to hand.

5. Locate the point $O_1$ in the carpometacarpal joint with a distance $l_{Io}$ and angle $\gamma_1 = 40^o$.

6. Locate in the top of the metacarpophalangeal joint for the index finger the point $O_2$ to distance $l_{oo_2}$ and one angle $\gamma_2 = 7^o$.

7. Locate in the top of the metacarpophalangeal joint for the middle finger the point $O_3$ to distance $l_{oo_3}$ and one angle $\gamma_3 = 13^o$.

8. Locate in the bottom of the metacarpophalangeal joint for the ring finger the point $O_4$ to distance $l_{oo_4}$ and one angle $\gamma_4 = 14^o$.

9. Locate in the bottom of the metacarpophalangeal joint for the small finger the point $O_5$ to distance $l_{oo_5}$ and one angle $\gamma_5 = 25^o$.

## Example of Changing Local to Global Coordinates

To show how to transform global coordinates to local coordinates, we present one example and build the transformation matrix for the transform coordinates. This example is for the thumb. The right and left hands have the same length and parameters; we show in this case the thumb of the left hand. Figure 3.16 shows the local coordinates $(x_{1L}, y_{1L}, z_{1L})$ where the

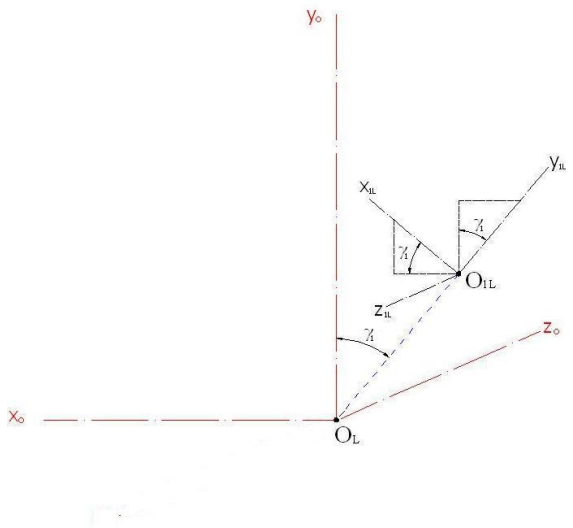


Figure 3.16: Global coordinates for thumb

subscript $(_{1L})$ is defined as follows: the first number is for the finger, or in this case, the thumb, and $(_L)$ is for the left hand. Point $O_{1L}$ using the same notation is for the local origin coordinates for the left thumb. Point $O_L$ is the global coordinates origin for the left hand. The global coordinates are $(x_o, y_o, z_o)$, and the distance $\ell_{O_L O_{1L}} = \ell_{Io} = 0.118HL$. This distance came

from the literature; see work by Buchcholz et al. [21] and Sancho-Bru [89]. From figure 3.16 we can write, with the convenient projections:

$$x_{1L} = \begin{bmatrix} \cos\gamma_1 \\ \sin\gamma_1 \\ 0 \end{bmatrix}; y_{1L} = \begin{bmatrix} -\sin\gamma_1 \\ \cos\gamma_1 \\ 0 \end{bmatrix}; z_{1L} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}; \ell_{Io} = \begin{bmatrix} -\ell_{Io}\sin\gamma_1 \\ \ell_{Io}\cos\gamma_1 \\ 0 \end{bmatrix}$$

And the transformation matrix for the thumb is:

$$^0H_1 = \begin{bmatrix} \cos\gamma_1 & -\sin\gamma_1 & 0 & -l_{Io}\sin\gamma_1 \\ \sin\gamma_1 & \cos\gamma_1 & 0 & l_{Io}\cos\gamma_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

In appendix A we can find the transformation matrix for the other fingers.

## 3.5 Hand Kinematic Formulation

For hand kinematics we study two possibilities: one is the forward kinematics and the other is inverse kinematics. Forward kinematics is when we know all the angles $(q_1 \dots q_{25})$ and the unknowns are the fingertip positions referred to as global coordinates. In inverse kinematics, we know the fingertip positions and we need to find the angles $(q_1 \dots q_{25})$. The solution is not direct and we need to solve several non-lineal equations.

The mathematical formulation can be found in the literature [9]. The formulation used is as follows: The position vector, is defined by $\mathbf{p}(\mathbf{q}^i)$ with respect to the local coordinate system by

$$\begin{bmatrix} \mathbf{p}(\mathbf{q}^i) \\ 1 \end{bmatrix} = {}^0A_1^1A_2 \ldots {}^{n-1}A_n \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

where $\mathbf{q}^i = [q_1 \ldots q_n]^T \; i = I, II, III, IV, V$ and $I$ is the thumb, $II$ is the index finger, $III$ is the middle finger, $IV$ is the ring finger, and $V$ is the small finger, and $n$ represents the DOF that depend on the finger right below. If we want to represent the position vector with respect to the global coordinate system (wrist), there is one transformation matrix for each finger we found this matrix (see the appendix and equation 3.1).

Therefore, the position vector with respect to the global frame is defined by

$$\begin{bmatrix} \mathbf{w}(\mathbf{q}^i) \\ 1 \end{bmatrix} = [{}^0H_i] \begin{bmatrix} \mathbf{p}(\mathbf{q}^i) \\ 1 \end{bmatrix} \tag{3.2}$$

With the application of the homogeneous transformation matrix, we can write:

$${}^0T_i = {}^0A_1^1A_2 \ldots {}^{n-1}A_n = \begin{bmatrix} n_x & o_x & p_x & d_x \\ n_y & o_y & p_y & d_y \\ n_z & o_z & p_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

where the vector $d = [d_x \; d_y \; d_z]^T$ is the position of the fingertip and each matrix $T$ has the angles for each joint and finger.

With the application of equation 3.3, we can find the vector $d$; in this case, it is forward kinematics. To find the angles when we know the vector

$d$ (position of fingertips) is inverse kinematics, the solution is not direct. In the next subsection, we present an example and discussion.

## Example: Index Finger Inverse Kinematics

The index finger has four degrees of freedom and to solve the inverse kinematics we use the theory of robotics [9], where:

$$T = {}^{0}A_1 . {}^{1}A_2 . {}^{2}A_3 . {}^{3}A_4 \tag{3.4}$$

and from the equation

$$T = \begin{bmatrix} n_x & o_x & p_x & d_x \\ n_y & o_y & p_y & d_y \\ n_z & o_z & p_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where the vector $d = [d_x \; d_y \; d_z]^T$ is a known vector position of the fingertip and each element of the right hand from the equation of the transformation matrix:

$${}^{0}A_1 = \begin{bmatrix} \cos(q_6 + \frac{\pi}{2}) & -\cos(-\frac{\pi}{2}) \sin(q_6 + \frac{\pi}{2}) & \sin(-\frac{\pi}{2}) \sin(q_6 + \frac{\pi}{2}) & 0 \\ \sin(q_6 + \frac{\pi}{2}) & \cos(-\frac{\pi}{2}) \cos(q_6 + \frac{\pi}{2}) & -\sin(-\frac{\pi}{2}) \cos(q_6 + \frac{\pi}{2}) & 0 \\ 0 & \sin(-\frac{\pi}{2}) & \cos(-\frac{\pi}{2}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{1}A_2 = \begin{bmatrix} \cos(q_7) & -\cos(0) \sin(q_7) & \sin(0) \sin(q_7) & \ell_{II-1} \cos(q_7) \\ \sin(q_7) & \cos(0) \cos(q_7) & -\sin 0 \cos(q_7) & \ell_{II-1} \sin(q_7) \\ 0 & \sin(0) & \cos(0) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$
{}^{2}A_{3} = \begin{bmatrix} \cos(q_8) & -\cos(0)\sin(q_8) & \sin(0)\sin(q_8) & \ell_{II-2}\cos(q_8) \\ \sin(q_8) & \cos(0)\cos(q_8) & -\sin 0\cos(q_8) & \ell_{II-2}\sin(q_8) \\ 0 & \sin(0) & \cos(0) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{3}A_{4} = \begin{bmatrix} \cos(q_9) & -\cos(0)\sin(q_9) & \sin(0)\sin(q_9) & \ell_{II-3}\cos(q_9) \\ \sin(q_9) & \cos(0)\cos(q_9) & -\sin 0\cos(q_9) & \ell_{II-3}\sin(q_9) \\ 0 & \sin(0) & \cos(0) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

The unknown are the angles $q_6,\ q_7,\ q_8$ and $q_9$. We can pre-multiply the equation on both sides for the inverse of $({}^{0}A_{1})^{-1}$, and the equation now:

$$({}^{0}A_{1})^{-1}.T = {}^{1}A_{2}.{}^{2}A_{3}.{}^{3}A_{4}$$

With this convenient operation we can see that from the left matrix the element $[3,4]_l$ is $-d_x\cos(q_6) - d_y\sin(q_6)$ and for the right hand the element $[3,4]_r$ is 0. We can assume that $[3,4]_l = [3,4]_r$, and we can calculate the first unknown with:

$$\tan(q_6) = -\frac{d_x}{d_y}$$

The tangent has two solutions with a variation of $\pi$, but the index has constrained the movement, and the range of motion is between $\frac{-13}{180}\pi$ and $\frac{7}{30}\pi$, and we only accept one solution within the range of motion. Note that this equation is only for this four-DOF system and this D-H table. We need three more independent equations to find the three unknowns. Following a similar method, the equation can be pre-multiplied for $({}^{1}A_{2})^{-1}$ on both sides and

$$({}^{1}A_{2})^{-1}.({}^{0}A_{1})^{-1}.T = {}^{2}A_{3}.{}^{3}A_{4}.$$

From this equation, $[1,4]_l = [1,4]_r$ and $[2,4]_l = [2,4]_r$; from this, we have two equations and three unknowns.

$$d_y \cos(q_6)\cos(q_7) - d_x \cos(q_7)\sin(q_6) - d_z \sin(q_7) - \ell_{II-1} = \ell_{II-2}\cos(q_8) +$$

$$+\ell_{II-3}\cos(q_8)\cos(q_9) - \ell_{II-3}\sin(q_8)\sin(q_9)$$

and

$$-d_y \cos(q_6)\sin(q_7) + d_x \sin(q_7)\sin(q_6) - d_z \cos(q_7) - \ell_{II-1} = \ell_{II-2}\sin(q_8) +$$

$$+\ell_{II-3}\sin(q_8)\cos(q_9) + \ell_{II-3}\cos(q_8)\sin(q_9)$$

We need one more equation to pre-multiply the equation for $({}^2A_3)^{-1}$. The next equation is

$$({}^2A_3)^{-1}.({}^1A_2)^{-1}.({}^0A_1)^{-1}.T = {}^3A_4$$

and with the new equation we can extract for $[1,4]_l = [1,4]_r$

$$-\ell_{II-1}\cos(q_8) - \ell_{II-2} - d_z(\cos(q_8)\sin(q_7) + \cos(q_7)\sin(q_8)) +$$

$$+d_y \cos(q_6)(\cos(q_7)\cos(q_8) - \sin(q_7)\sin(q_8)) -$$

$$-d_x \sin(q_6)(\cos(q_7)\cos(q_8) - \sin(q_7)\sin(q_8)) = \ell_{II-3}\cos(q_9)$$

We have one system of equations with three equations and three unknowns. Is a system of non-linear equations and to solve this system of equations, we need some iteration methods or applied optimization.

## 3.6 Results

### Forward Kinematics

For forward kinematics, we built a code in Visual Basic. Figure 3.17 shows the interactive, user-friendly screen. We can enter the hand breath ($HB$), the hand length ($HL$), and all the known angles, and the code produces the fingertip positions for all the fingers.

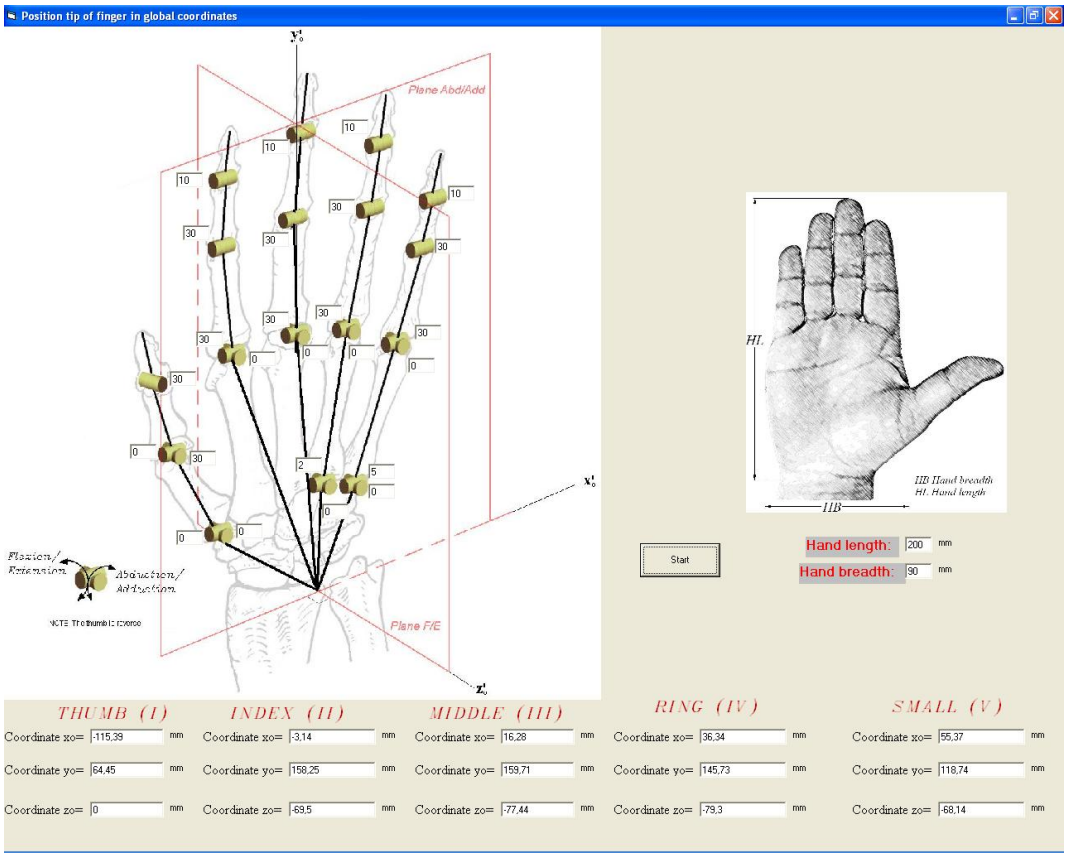


Figure 3.17: Screen for forward kinematic

At the bottom of the screen, we can see the fingertip position in global coordinates.

## Inverse Kinematics

When the position of the fingertip is given, the unknowns are the angles for the different joints from Equation 3.2. To reach the position by the fingertip, the joint angles can be different; therefore, the inverse kinematic solutions are generally not unique. However, the human body has some natural restrictions (i.e., the DIP joint cannot hyperextend 100 degrees) and the feasible space is really small for this problem. Therefore, the solution can be approximately unique.

To implement the inverse kinematics algorithm, we adapt a method from the robotics theory. The method requires that the number of unknowns is the same as that of equations, and finally, it solves a set of highly nonlinear equations. The Newton-Raphson (N-R) method is implemented for 4 DOF, 5 DOF and 6 DOF, but for some configurations the system of equations will not converge. That means the starting point is critical to the convergence of the N-R method. To enforce the N-R method convergence, we implement the steepest descent method to find a reasonable starting point before we solve the nonlinear equations using the N-R method. Figure 3.18 shows

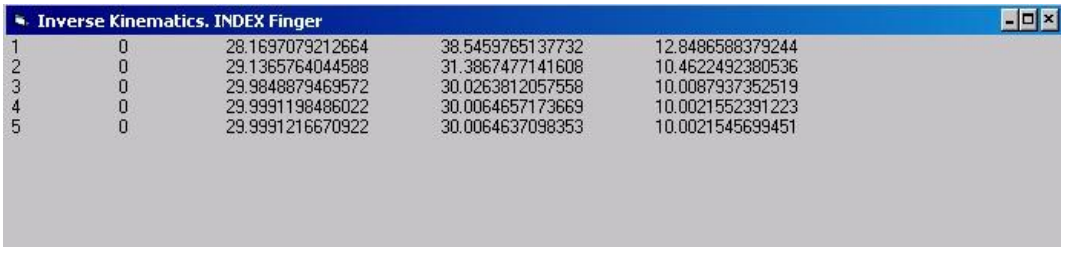


Figure 3.18: Screen for inverse kinematic

the fast convergence with the N-R method. The rows are the number of iterations; the first column is for the number of iterations, the second is for $q_6$, and the third is for $q_7, q_8$ and $q_9$.

## 3.7 Conclusions

In this chapter, the novel hand model is presented with 25 DOF. With this model, we work with the change of coordinates from local to global. The D-H method is applied for each finger. Forward and inverse kinematics algorithm have been implemented. It was shown that this 25-DOF hand model is a realistic model. It was also shown that this model is applicable to different percentiles because the D-H method is used.

The parametric model for each finger was shown and with this parametrization our model can be modified to include all hand sizes.

Chapter

# 4 Workspace Analysis

## 4.1 Introduction

The objective of this chapter is the representation of the workspace for each finger. In Chapter 3 we showed the parameters for $\theta_i$, $\alpha_i$, $d_i$ and $a_i$, and presented the forward and inverse kinematics for each finger. When the virtual human hand touches or grasps any object, it needs to know if the object is feasible for the virtual hand, i.e., if the object is in the workspace of the hand. This chapter shows the formulation for generating the workspace for each finger, beginning with Jacobian methods and the row-rank deficiency criteria. In the next section, we work to the formulation and after then show how to built the 25-DOF model visualization workspace. The two last sections are dedicated to some examples.

## 4.2 Jacobian Rank Deficiency Method

### Row-rank deficiency criteria

The position vector function generated by a point of interest (typically a fingertip) is written as a multiplication of rotation matrices and position vectors as

$$\mathbf{X}(q) = \begin{bmatrix} x(q) \\ y(q) \\ z(q) \end{bmatrix}$$

$$X(p) = \sum_{i=1}^{i=n} [\prod_{j=1}^{j=i-1} {}^{j-1}\mathbf{R}_j]^{i-1}\mathbf{p}_i = \mathbf{\Phi}(\mathbf{q}), \tag{4.1}$$

where both ${}^{i}\mathbf{p}_j$ and ${}^{i}\mathbf{R}_j$ are defined using the Denavit-Hartenberg (D-H) representation method, such that

$${}^{i-1}\mathbf{R}_i = \begin{bmatrix} \cos q_i & -\cos\alpha_i \sin q_i & \sin\alpha_i \sin q_i \\ \sin q_i & \cos\alpha_i \cos q_i & -\sin\alpha_i \cos q_i \\ 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix}$$

and

$${}^{(i-1)}\mathbf{p}_i = [a_i \cos q_i \quad a_i \sin q_i \quad d_i]^T,$$

where $q_i$ is the joint angle from the $x_{i-1}$ axis to the $x_i$ axis, $d_i$ is the shortest distance between the $x_{i-1}$ and $x_i$ axes, $a_i$ is the offset distance between the $z_i$ and $z_{i-1}$ axes, and $\alpha_i$ is the offset angle from the $z_{i-1}$ and $z_i$ axes.

The vector function $\mathbf{\Phi}(\mathbf{q}^*)$ characterizes the set of all points inside and on the boundary of the reach envelope generated by an anatomical landmark, typically a fingertip. The objective is to visualize this vector function

consisting of many parameters and to better understand the motion governed by $\mathbf{\Phi}(\mathbf{q}^*)$. At a specified position in space given by $P(x_p, y_p, z_p)$, Eq. 4.1 can be written as a constraint function as

$$\mathbf{\Omega}(\mathbf{q}^*) = \begin{bmatrix} x(\mathbf{q}^*) - x_p \\ y(\mathbf{q}^*) - y_p \\ z(\mathbf{q}^*) - z_p \end{bmatrix} = \mathbf{0}, \tag{4.2}$$

## Range of Motion

Joint limits (ranges of motion) are imposed in terms of inequality constraints in the form of $q_i^L \leq q_i \leq q_i^U$, where $q_i^L$ and $q_i^U$ are the lower and upper limits, respectively, and where $i = 1, \ldots n$, where $n$ is the number of DOFs. In order to include the range of motion in the formulation, we introduce a new set of generalized coordinates $\lambda = [\lambda_1 \ldots \lambda_n]^T$, such that

$$q_i = \big((q_i^L + q_i^U)/2\big) + \big((q_i^U - q_i^L)/2\big) \sin \lambda_i,$$

where the new variable $\lambda_i$ is inherently constrained by the sine function and does not change the dimensionality of the problem. In order to include the effect of the ranges of motion, augmentation of the constraint equation $\mathbf{\Phi}(\mathbf{q}^*)$ with the parameterized ranges of motion is proposed, such that

$$\mathbf{H}(\mathbf{q}) = \begin{bmatrix} x(\mathbf{q}^*) - x_p \\ y(\mathbf{q}^*) - y_p \\ z(\mathbf{q}^*) - z_p \\ q_i - a_i - b_i \sin \lambda_i \end{bmatrix} = \mathbf{0}, \tag{4.3}$$

where $i = 1, \ldots, n$, and $\mathbf{q} = [q^{*T} \quad \lambda^T]^T$ is the vector of all generalized coordinates. Note that although $n$-new variables ($\lambda_i$) have been added, $n$-equations have also been added to the constraint vector function without

losing the dimensionality of the problem. The *Jacobian* of the constraint function $\mathbf{H}(\mathbf{q})$ at a point $\mathbf{q}^0$ is the $(3+n) \times 2n$ matrix

$$\mathbf{H}_{\mathbf{q}} = \partial \mathbf{H} / \partial \mathbf{q}, \tag{4.4}$$

where the subscript denotes a derivative. With the modified formulation that includes ranges of motion, the Jacobian is expanded to

$$\mathbf{H}_{\mathbf{q}} = \begin{bmatrix} \mathbf{\Phi}_{\mathbf{q}^*} & \mathbf{0} \\ \mathbf{I} & \mathbf{q}^*_\lambda \end{bmatrix}, \tag{4.5}$$

where $\mathbf{q}^*_\lambda = \partial \mathbf{q}^* / \partial \lambda$, $\mathbf{\Phi}_{\mathbf{q}^*} = \partial \mathbf{\Phi} / \partial \mathbf{q}^*$, $\mathbf{0}$ is a $(3 \times n)$ zero matrix, $\mathbf{I}$ is the identity matrix, and

$$\mathbf{\Phi}_{\mathbf{q}^*} = \begin{bmatrix} x_{q_1} & x_{q_2} & \dots & x_{q_n} \\ y_{q_1} & y_{q_2} & \dots & y_{q_n} \\ z_{q_1} & z_{q_2} & \dots & z_{q_n} \end{bmatrix}$$

$$\mathbf{q}^*_\lambda = \begin{bmatrix} -((q_1^U - q_1^L)/2)\cos\lambda_1 & 0 & \dots & 0 \\ 0 & -((q_2^U - q_2^L)/2)\cos\lambda_2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & -((q_n^U - q_n^L)/2)\cos\lambda_n \end{bmatrix}$$

Abdel-Malek et al. in their earlier work aimed at determining difficulties in the control of robot manipulators [1–4], and they later showed that impediments to motion (halting of a trajectory) arise inside the workspace when the Jacobian becomes singular [5]. Because the Jacobian is not square, rank deficiency criteria were developed. Before addressing these criteria, however, it is important to show why the singularity of the Jacobian has a direct effect on the control. The differentiation of Eq. 4.1 with respect to time yields the velocity of the fingertip $\dot{\mathbf{\Phi}}$ as

$$\dot{\mathbf{\Phi}} = \mathbf{\Phi}_{\mathbf{q}^*}\dot{\mathbf{q}}^* \tag{4.6}$$

where $\dot{\mathbf{q}}^*$ is the vector of joint velocities. Given a specified path trajectory (i.e., $\dot{\mathbf{\Phi}}$ ), the calculation of $\dot{\mathbf{q}}^*$ (i.e., joint velocities) requires computing an inverse of the Jacobian $\mathbf{\Phi}_{\mathbf{q}^*}$. For a singular Jacobian, it is not possible to compute the required velocities. These cases are typically associated with a kinematic configuration of the upper extremity that does not admit motion in a particular direction and requires a change in the finger's posture in order to execute the path. If the Jacobian was square, then the determinant of $\mathbf{\Phi}_{\mathbf{q}^*}$ will yield the postures in space where singular behavior occurs. We will use this concept to explore the surrounding workspace.

We can proceed in similar way for each finger; the only difference based in Chapter 3 is the number of DOF. For the index finger and the middle finger we represent the movement of these two fingers with four DOF; for the other two fingers; ring and small, we use six DOF.

## 4.3 Formulation

We will use the idea of a singular *Jacobian* to identify all barriers inside and on the boundary of the workspace. Because the Jacobian is non-square, we define such barriers as a subset of the workspace at which the Jacobian of the constraint function of Eq. 4.4 is **row rank deficient** [2]; i.e., barriers are defined by $\partial W$ and characterized by

$$\partial W \subset \{\text{Rank } \mathbf{H}_{\mathbf{q}^*}.(\mathbf{q}^*) < k,\ \textit{for some } \mathbf{q}^* \textit{ with } \mathbf{H}(\mathbf{q}^*) = \mathbf{0}\}\,, \tag{4.7}$$

where $k$ is at least $(3 + n - 1)$. Because of the form of the Jacobian characterized by Eq. 4.7, three distinct conditions arise:

**1. Type I singularity sets:** If no joints have reached their limits, the diagonal sub-matrix $\mathbf{q}_\lambda$ is full row rank. Therefore, the only possibility for $\mathbf{H}_{q^*}$ to be row-rank deficient is when the block matrix $\mathbf{x}_q$ is row rank deficient. The type I singularity set is defined as

$$S^{(1)} \equiv \{\mathbf{p} \in \mathbf{q} : Rank[\mathbf{x_q}] < 3, for\ some\ constant\ subset\ of\ \mathbf{q}^*\}, \quad (4.8)$$

where $\mathbf{p}$ is within the specified joint limit constraints and may contain joints that are functions of others or constant values.

**2. Type II singularity sets:** When certain joints reach their limits, e.g., $\partial\mathbf{q}^{\lim} = [q_i^{\text{limit}}, q_j^{\text{limit}}, q_k^{\text{limit}}]^T$, the corresponding diagonal elements in the matrix will be equal to zero. Therefore, the corresponding matrix is subjected to the rank-deficiency criterion, where will take on the following form

$$\mathbf{H}_{q^*} \approx \begin{bmatrix} \mathbf{x}_{q_1} & \dots & \mathbf{x}_{q_i} & \mathbf{x}_{q_j} & \mathbf{x}_{q_k} & \dots & \mathbf{x}_{q_n} \\ 0 & \dots & 1 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & 1 & \dots & 0 \end{bmatrix} \quad (4.9)$$

and where the three columns pertaining to $\mathbf{x}_{q_i}$ , $\mathbf{x}_{q_j}$ , and $\mathbf{x}_{q_k}$ are removed such that the rank deficiency criteria are applied again. From the foregoing observation, the second type of singular sets are formulated. Define a new vector $\partial\mathbf{q}^{\lim} = [q_i^{\text{limit}}, q_j^{\text{limit}}, q_k^{\text{limit}}]^T$ which is a sub-vector of $\mathbf{q}$ where

$$1 \leqslant \dim\left(\partial\mathbf{q}^{\text{limit}}\right) \leqslant (n-3).$$

The type II singularity set is defined as

$$\left\{\mathbf{p} = [\hat{\mathbf{p}} \cup \partial\mathbf{q}^{\text{limit}}] : \ \text{Rank}[\mathbf{x_q}(\mathbf{w}, \partial\mathbf{q}^{\text{limit}})] < 3, \text{for some } \hat{\mathbf{p}} \in \mathbf{q}^*, \ \mathbf{dim}(\partial\mathbf{q}^{\text{limit}}) \leqslant (n-3)\right\} \tag{4.10}$$

where $\widehat{\mathbf{p}}$ is the singular set as a result of applying the rank deficiency criteria to Eq. 4.9.

**3. Type III singularity sets:** are all sets that are composed of the combination of joints at their limits and is defined by:

$$S^{(3)} \equiv \left\{\mathbf{p} \in \mathbf{R}^{(n-2)} : \ \mathbf{p} \equiv \partial\mathbf{q}^{\text{limit}} = [q_i^{\text{limit}}, q_j^{\text{limit}}, \dots]\right\} \tag{4.11}$$

where $i \neq j$.

Barriers are identified by substituting the sets $\mathbf{p}_i$ characterized by Eqs. 4.8, 4.10, and 4.11 into the accessible set $\mathbf{x}(\mathbf{p})$, which yields the equation of a surface that can be readily shown. This surface is indeed a barrier associated with a generalized variable that has reached its limit. Determining joint angles of a fingers given a specific position and orientation was defined as the inverse kinematics in Chapter 3. Motion from one configuration to another along a trajectory sometimes requires halting the motion and changing the inverse kinematics in order to proceed with the motion. An example of this occurs when attempting to reach a point located behind one's shoulder. Starting with one trajectory may become very uncomfortable because of joint limits, while trying another trajectory becomes simpler. Similarly, reaching a doorknob and turning sometimes is difficult to complete and requires orienting the initial hand configuration in a different posture. These barriers due to singular sets identified by Eqs. 4.8, 4.10, and 4.11 may admit motion only in one normal direction, and hence are called impediments to motion by Abdel-Malek, et al. [2]. In this case, the arm, for example, will not be able to cross such a barrier.

## 4.4 Parametric Model for Each Finger

Imposition of the rank deficiency condition can be implemented using a variety of methods; perhaps the most computationally efficient one is the repeated elimination of square sub-Jacobians, until several non-linear equations are determined. For example, consider a 5-DOF model of the thumb finger, where the Jacobian is $(8 \times 10)$, where $\mathbf{\Phi}_{\mathbf{q}^*}$ is in the following form,

$$\begin{bmatrix} - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \end{bmatrix}_{3\times 5}$$

and where $\mathbf{q}^*_\lambda$ is in this form:

$$\begin{bmatrix} - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \end{bmatrix}_{5\times 5}$$

Therefore, three sets of singularities can be identified:

**I)** The largest square submatrix of $\mathbf{\Phi}_{\mathbf{q}^*}$ is $3 \times 3$

$$\begin{bmatrix} \underbrace{\begin{bmatrix} - & - & - \\ - & - & - \\ - & - & - \end{bmatrix}}_{J_1} & - & - \\ & - & - \\ & - & - \end{bmatrix}_{3\times 5} = \cdots = \begin{bmatrix} - & - & \underbrace{\begin{bmatrix} - & - & - \\ - & - & - \\ - & - & - \end{bmatrix}}_{J_{10}} \\ - & - & \\ - & - & \end{bmatrix}_{3\times 5}$$

There are a total of $\frac{n!}{3!(n-3)!} = \frac{5!}{3!2!} = 10$ submatrices (also called sub-Jacobians). The determinants of these matrices are

$$|J_1| = \begin{vmatrix} - & - & - \\ - & - & - \\ - & - & - \end{vmatrix}, \ldots, |J_{10}| = \begin{vmatrix} - & - & - \\ - & - & - \\ - & - & - \end{vmatrix}$$

If all the determinants are kept at zero, then the group of equations can be numerically solved to identify the singular sets.

**II)** When it reaches its limit, eliminate the *ith* column in $\mathbf{\Phi}_{\mathbf{q}^*}$, and it will be a $3 \times 4$ matrix. Therefore, 4 sub-Jacobians and 4 equations can be solved together to obtain the third set. When $q_i$ and $q_j$ reach their limits, repeat the same procedure to find the singular set. When $q_i$ , $q_j$, and $q_k$ reach their limits, repeat the same procedure to find the singular set until the remaining Jacobian of $\mathbf{\Phi}_{\mathbf{q}^*}$ is a square matrix. These sets are denoted by a, b, and c, where each set comprises a number of constant joint values (from the total number of DOFs) and two variables. Upon substituting these sets into Eq. 4.1, we obtain an equation of a surface parameterized in terms of two variables (i.e., a 2-DOF surface in 3D space), such that:

$$\mathbf{f}^{(i)}(\mathbf{u}^{(i)}) = \mathbf{\Phi}(\mathbf{u}^{(i)}, \mathbf{q}^{+}) \tag{4.12}$$

where $\mathbf{q}^{+} \rightarrow \mathbf{q}^{*} \cap \mathbf{u}^{(i)}$ and $\mathbf{f}^{(i)}$ is the vector function describing the new parametric surface characterizing a barrier to motion.

**III)** Since $\mathbf{q}^{*}_{\lambda}$ is a square matrix, the singular sets can be obtained by solving the equation

$$|\mathbf{q}_{\lambda}^{*}| = \begin{vmatrix} - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \end{vmatrix} = 0.$$

The singular sets are exact when the joints reach their limits.

## 4.5 Visualization Workspace

In order to simplify the model, without losing generality, we present the algorithm for the thumb and after then we can extend it to other fingers. Do not show the workspace for each finger in the whole hand, because the visualization becomes very confused.

### Thumb

In Chapter 3 we showed the model and D-H table for the fingers. In this case, for the thumb, the range of motions are defined as follows in Table 4.5

| | Min. | Max. |
|---|---|---|
| $q_1$ | 0 | 60 |
| $q_2$ | -25 | 35 |
| $q_3$ | 0 | 60 |
| $q_4$ | -10 | 55 |
| $q_5$ | -15 | 80 |

Table 4.1: Range of motion thumb. Angles in degrees

**Singularity set**

Since $\mathbf{\Phi_{q^*}}$ is a $3 \times 5$ matrix, there are $\frac{n!}{3!(n-3)!} = \frac{5!}{3!2!} = 10$ equations to be simultaneously solved, which represent the determinants of all square $3 \times 3$ sub-Jacobians. There are three sets of solutions:

I) As a result of solving the equations generated by the sub-Jacobians, the rank deficiency of $\mathbf{\Phi_{q^*}}$ is a $3 \times 5$ yields $s_1 = \{q_3 = 0, q_4 = 0, q_5 = 0\}$. This set, when substituted into Eq. 4.1, yields a surface parameterized into three generalized coordinates.

II) The second singularity set can be obtained by substituting two joint limit into the Jacobian and solving the determinant of the sub-Jacobian excluding the column corresponding to the joint reaches limit.

III) The third set occurs when three of the variables reach their constraint limits. There are a total of 80 singular surfaces. In order to perform this symbolic manipulation, we have developed a computer code using Mathematica.

When we plot all of the singular surfaces together and the boundary will be obtained, as shown in Figures 4.1 and 4.2.

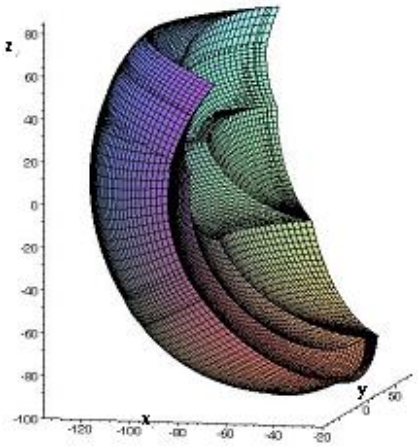


Figure 4.1: One view of the workspace for thumb.

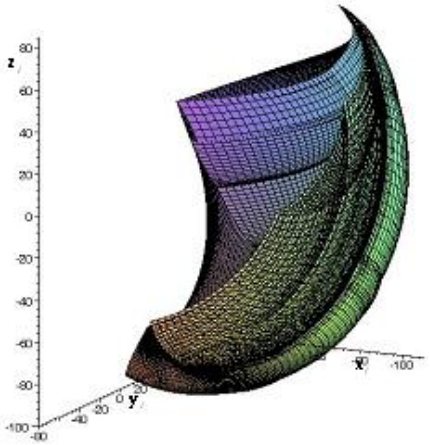


Figure 4.2: Another view of the workspace for thumb.

By implementing the same procedure for the thumb, one can obtain the workspace of the index finger. Note that the index and middle fingers have similar kinematic model, each finger has 4 DOFs, however, the dimensions are different. Therefore, here we only show the workspace of index fingers as Figures 4.3 and 4.4.

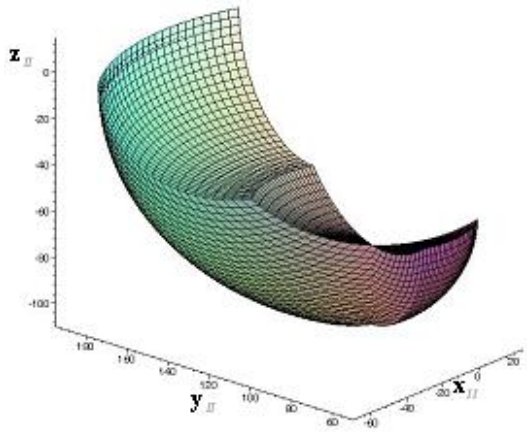


Figure 4.3: One view of the workspace for the index finger.

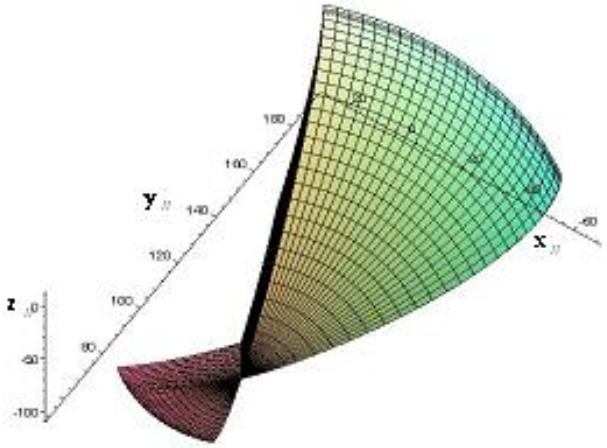


Figure 4.4: Another view of the workspace for the index finger.

Similarly, the ring and small fingers have the same kinematic model, and we only show the workspace of the ring fingers in Figures 4.5 and 4.6.

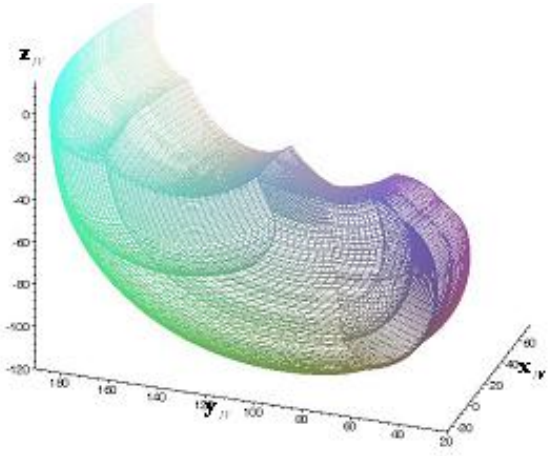

Figure 4.5: One view of the workspace for the ring finger.

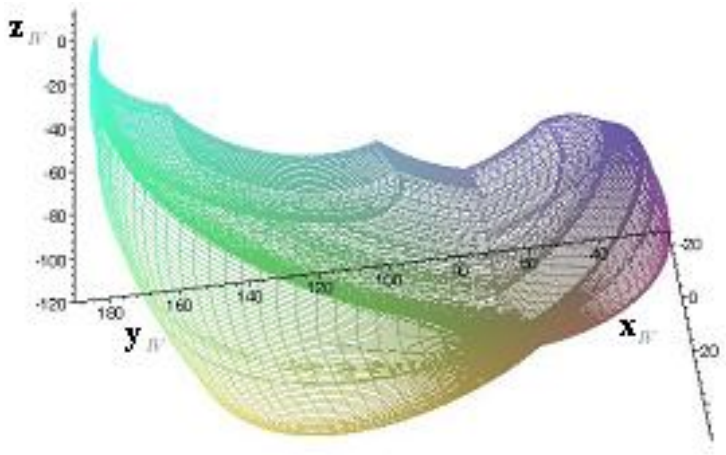

Figure 4.6: Another view of the workspace for the ring finger.

Hands are connected to the upper body, and a similar process can applied to this part of the virtual human. Figure 4.7 depicts the workspace for the upper body. Although this is beyond the scope of this dissertation, the hand is connected with the upper body by the arm and for some actions that approximate the virtual human to the virtual object, it is interesting to know about the whole body.

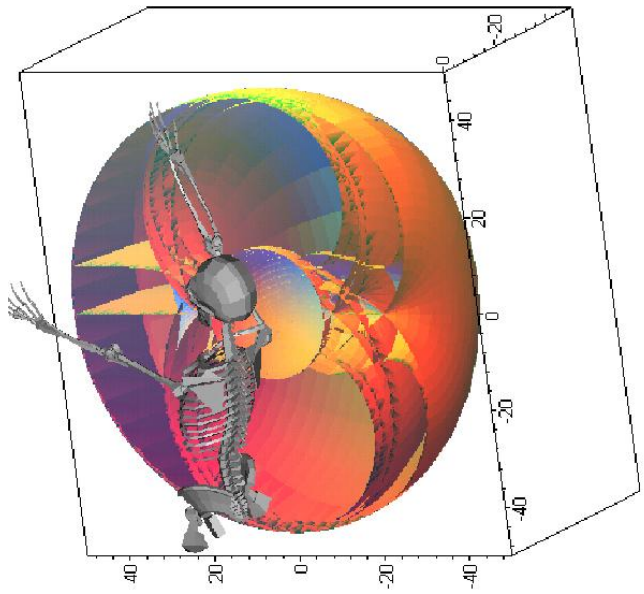


Figure 4.7: Depicting the workspace and barriers with respect to the torso.

## 4.6 Conclusions

In this chapter, we showed the formulation for generating the workspace. Before the virtual human grasps the object, it is interesting to know the position of the object with respect to the hand, and this permits the virtual human to acquire some propioception between the virtual hand and the object in the virtual environment. In the next chapters, when we discuss virtual grasping, we show that with knowledge of the feasibility of the object, the virtual human may first need do some action, i.e., walk, move the upper body, etc., for to put the object in the workspace of the hand, if it is not already there.

Chapter

# 5 A Task-oriented Object Grasping Approach: Theory

## 5.1 Introduction

Several authors have spent many years and a great deal of effort studying grasping. We divide this research into two large groups: one is dedicated to robotics, and the other is dedicated to virtual human grasp. The groups are related because the techniques and methods used in robotics are applied to the virtual human hand and vice versa. In the virtual human hand group, see the work from Miyata et al. [69, 70, 72]. This group is working in motion capture, and after implementing in virtual environment, studied several types of grasp and application like a cell phone. Savescu et al. [90] present a model of a virtual hand and implement it in MAN3D. For more references and authors, in chapter 1 we describe the state of the art for grasping. In this chapter, the objective is to present a novel theory for grasping in the virtual environment (VE) any object with the virtual human (VH). The novel concepts for this application are the autonomous grasp, implementation of several types of grasp, and a new algorithm for grasp. The chapter begins with a basic concept of grasp, definitions of grasping

parameters, and the relationship between grasping parameters. We present a mathematical model for the type of grasp, and the next section shows the algorithm implemented for how to grasp. In the last section, we discuss this chapter and present conclusions.

## 5.2 Basic Concepts

For basic concepts, we introduce definitions from the literature [61], [94], [95].

**Free Motion** In free motion, the hand moves freely in space. The basic free motions possible for the hand are:

- *Opening*: Fully extending the fingers and the thumb until the hand is fully open, as in the anatomical position.
- *Closing*: Fully flexing until the hand is closed in a fist with the thumb overlapping the index and middle fingers.
- *Clawing*: The motion that reaches the terminal position of metacarpophalangeal (MP) extension, interphalangeal (IP) flexion.
- *Reciprocal*: The motion that reaches a terminal position of MP flexion, IP extension.

Twelve variations of these motions are observed if the terminal position of each motion is the starting point of each other motion.

**Resisted Motion** Resisted motion is that performed by the hand against an external resistance, for the purpose of exerting force on an external object and sometimes changing its position.

- *Power grip*: Gripping an object against the palm (primarily isometric motion).

- *Precision handling*: Manipulation of an object by the thumb and fingers, not in contact with the palm (primarily isometric motion).
- *Pinch*: Isometric compression between the thumb and fingers.

## 5.3 Grasping Parameters

Before grasping the object, the VH needs do some actions first. We classify these actions as pre-grasp, grasp, and after-grasp. To concentrate only on grasp, we consider the object and the VH in position, meaning that the object is in the workspace of the hand. Pre-grasp actions are:

- *Hand Position:* The position of the wrist respect to the object.
- *Hand Orientation:* The hand has adequate orientation for the grasp.

These two pre-grasp actions need to work together. If we consider each separately, a good position with a bad orientation, results in an inability to grasp the object, and good orientation with a bad position results in the object not being reachable. In our algorithm both actions are doing in same time.

In this dissertation, after-grasp actions are considered in terms of the final position of the object.

Grasp involves operations related to the hand and the object, and we consider grasp touch, pull, push, etc. If we take the definition from a dictionary, one of the definitions is to take hold of or seize firmly with the hand. We extend this definition to include touch, pull, etc., ultimately amplifying the concept of grasp.

Parameters for the object and the hand are considered in the new step called grasp.

- *Object Attributes:* In the virtual environment, the object was built with techniques of computer-aided geometric design, and the basic attributes there are known. Other attributes, such as temperature, are described below.

- *Hand Orientation:* Hand orientation is related to hand shape. An adequate lecture of object attributes can give all possible options for grasp, with axes, planes, etc. Hand surface and object surface can give the hand orientation relative to the object.

- *Hand Position:* Hand position is similar to hand orientation and follows the same procedures for positioning the wrist as in the function of the object attributes.

- *Task:* This parameter can help the virtual human decide how to grasp the object.

- *Object Initial Position :* In some operations we need to know the initial position.

- *Object Final Position :* In some operations we need to know the final position.

- *One or Two Hands :* This parameter is related to several of the attributes discussed above.

- *Finger Number :* The number of fingers to use, depending on the type of grasp, object shape, etc.

- *Object Weight :* Object weight can be derived from the object shape if we know the density.

- *Object Stability:* For any small movement close to the position of equilibrium, the object stays in the equilibrium position.

- *Hand Anthropometry:* Virtual humans like real humans, have different-sized hands.

## 5.4 Relationship between Grasping Parameters

In this section, we establish the relationship between parameters and demonstrate that they need work together sometimes. To show this relationship, we present Figure 5.1. This figure shows the relationship independent of the dominant hand; for our VH, the dominant hand is usually the right hand.

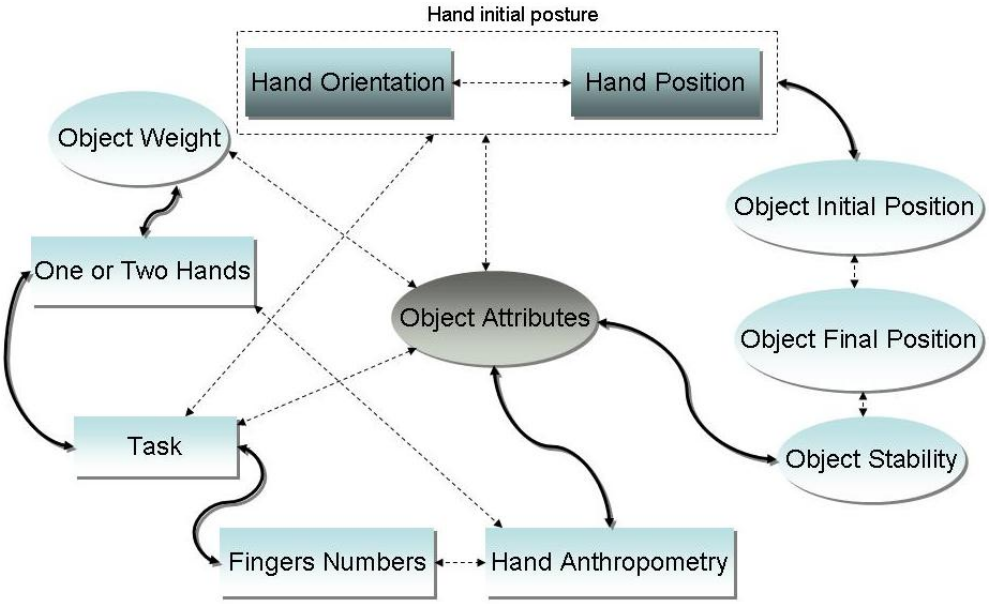


Figure 5.1: Relation between grasping parameters

In this figure, fill arrows indicate functions between parameters that connect, arrows with dashed lines show relationships, the elliptic shapes are for the object, and rectangular shapes are for the hand. Hand orientation and hand position are related and closed with a dashed rectangle; both parameters work together, and we classify both as hand initial posture.

Hand initial position is a direct function with object initial position; they each depend one the other, e.g. in order to perform some action with the object, knowledge of the hand becomes necessary and with this first approximation we can know if the object is reachable or not. These parameters are related to object attributes, i.e., the object attributes permit different actions and depend on the hand initial position. In a similar way, the task can be done in relation with the hand initial position, i.e., if the object is not in the workspace, the task cannot be done.

The object weight, also relates to the object attributes, in the virtual environment, if we know the geometry and the density of the object we can know the weight with the geometric relation $W = V_0.\gamma$ in absolute terms, where $W$ is the weight, $V_0$ is the volume, and $\gamma$ is the specific weight .

The number of fingers and hand anthropometry are related, and the hands are used during the action too.

Object stability is a function of the object shape; a tall glass is less stable than a short glass when it is sitting on a table.

When the action is to do some particular task, the action is related directly to whether one or two hands are used or how many fingers are used. In the section below, we describe some tasks in which we can see this relation.

## 5.5 Parameters and Relationship

### Object Attributes

Figure 5.2, shows the object attributes from engineering design. Attributes inherent to the object are volume, mass, inertia center, and inertia matrix. Some commercial programs call these characteristics, but we call them attributes. Inertia center is the center of mass (COM). Other attributes are:

- Temperature
- Fragility
- Surface shape

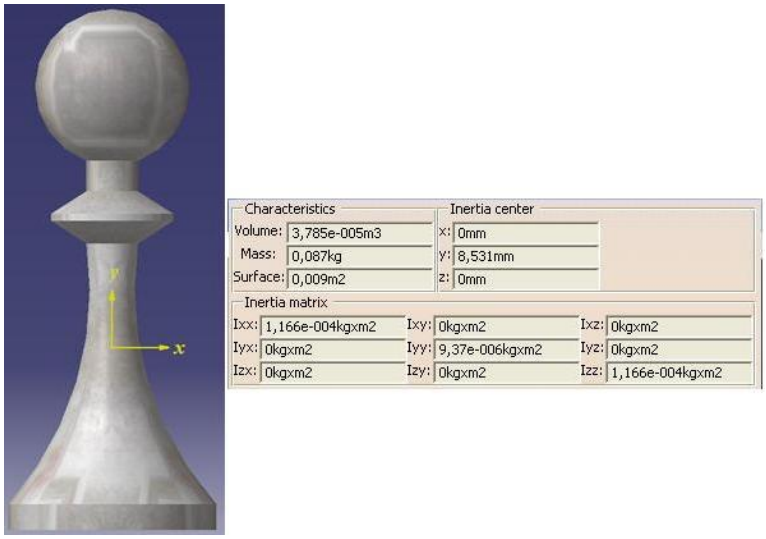


Figure 5.2: Object with some attributes information

**Temperature:** When the objects have a temperature, this attribute can help decide what type of grasp is required.. For example, when grasping

a mug, if the object are is filled with hot coffee and the action is to move we do not can grasp the side.

**Fragile:** If the object is fragile, this attribute can help decide what type of grasp is required.

**Surface Shape:** In a virtual environment, when we use the B-rep form found in the references [54], [40], [102], and [99] , we show the definition of the object shape. In domain $M$ in the plane with parameters $(u, v)$, there is continuously differentiable and locally injective mapping $M \rightarrow S$, which takes points $(u, v)$ in $M$ into $\Re^3$. Then every point in the image set $S$ can be described by a vector function $\mathbf{X}^i(u, v)$, where $i = 1, \ldots, n$ and $n$ is the number of object, and represents the object $i$ selected by the user. $\mathbf{X}^i(u, v)$ is called the *parametrization* of the surface $S$ for the object $i$, and $u, v$ are called the parameters of this representation; see Figure 5.3.

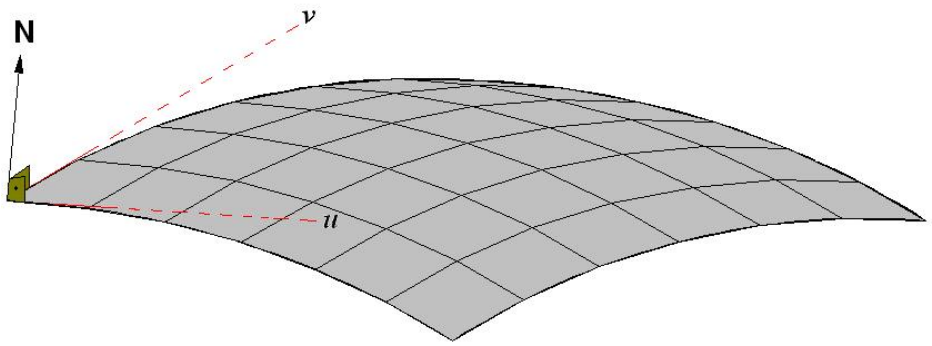


Figure 5.3: Parametric surface

A parametrization is *regular* if for every point of $S$, the normal vector is defined:

$$\left| \frac{\partial \mathbf{X}^i}{\partial u} \times \frac{\partial \mathbf{X}^i}{\partial v} \right| \neq 0 \tag{5.1}$$

If $\left|\frac{\partial \mathbf{X}^i}{\partial u} \times \frac{\partial \mathbf{X}^i}{\partial v}\right|$ has a zero at the point $\mathbf{P}(u_0, v_0)$, then the surface has a *singularity* at $\mathbf{P}$.

The *tangent vector* to a surface curve $\mathbf{X}^i(u(t), v(t))$ can be computed as

$$\dot{\mathbf{X}}^i = \frac{\partial \mathbf{X}^i}{\partial u}\dot{u} + \frac{\partial \mathbf{X}^i}{\partial v}\dot{v} \tag{5.2}$$

In particular, the tangents to the parametric curves are given by

$$\mathbf{X}_u^i = \frac{\partial \mathbf{X}^i}{\partial u}\ (lines\ v = const.), \quad \mathbf{X}_v^i = \frac{\partial \mathbf{X}^i}{\partial v}\ (lines\ u = const.) \tag{5.3}$$

The two vectors $\frac{\partial \mathbf{X}^i}{\partial u}$ and $\frac{\partial \mathbf{X}^i}{\partial v}$ uniquely determine the plane, which is tangent to the surface at the point $\mathbf{P}(u, v)$.

The *unit normal vector* to the surface can compute for the object, using the vector product, as

$$\mathbf{N_o} = \frac{\mathbf{X}_u^i \times \mathbf{X}_v^i}{|\mathbf{X}_u^i \times \mathbf{X}_v^i|} \tag{5.4}$$

Usually we assume that the hand size is defined by two parameters shown in Chapter 3, $HL = 190\ mm$ and $HB = 90\ mm$. With these two parameters and the object size, we can know if the virtual human can grasp the object as a whole or if it needs to search for parts of the object for grasp, i.e., one cannot grasp a whole chair, but we can grasp the back of the chair.

## Hand Orientation

For hand orientation, we use the theory for two oriented surfaces. In this case the hand surface is oriented with the object surface, and the hand can be the right, left, or both hands. For simplify, we refer in this case to the right hand; the same equations can apply for the left hand or both

hands. One surface $M_H$, where the subscript $H$ indicates the hand surface, is orientable if the mapping of $M_H \rightarrow S^2$ is *regular*, and the vector normal is defined in Equation 5.4. If we change to other coordinates $(u_1, v_1)$, shown for the hand in Figure 5.4, the new parametric representation is

$$\mathbf{Y}^j_{u_1} \times \mathbf{Y}^j_{v_1} = \frac{\partial(u,v)}{\partial(u_1,v_1)} \mathbf{X}^i_u \times \mathbf{X}^i_v \tag{5.5}$$

where $j = 1, 2$, 1 for the right hand, 2 for the left hand, and for the corresponding unit vector:

$$\mathbf{N}_H = \epsilon \mathbf{N}_o \tag{5.6}$$

where

$$\epsilon = sign \left| \frac{\partial(u,v)}{\partial(u_1,v_1)} \right| \tag{5.7}$$

The orientation in the new parametrization is the same if the **Jacobian** of the transformation is positive, and is opposed if it is negative.

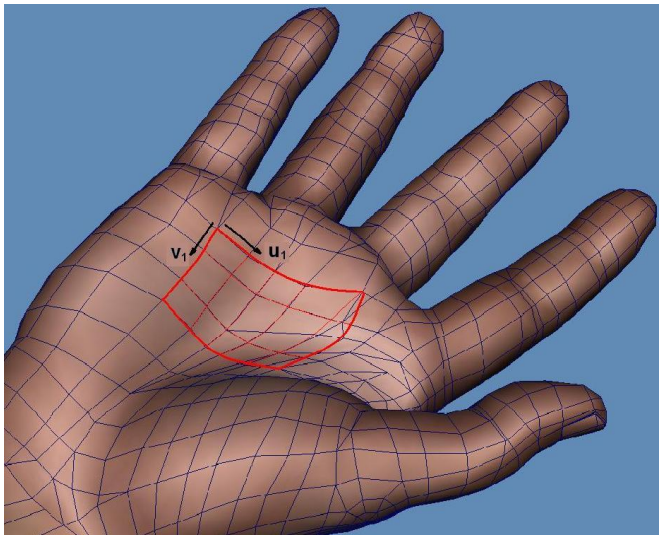


Figure 5.4: Parametric surface for the hand

## Hand Position

To consider the hand position with respect to the object, we refer this position with respect to the wrist. For us the global position for the hand is given by the position of the wrist. The point of reference for the object is the COM.

When the object enters the virtual environment, we know the position of COM. That is information inherent with the object; the position of the wrist is also known in each moment.

The coordinates of COM are $\mathbf{x}_c^i = [x_c^i \; y_c^i \; z_c^i]^T$, and the coordinates of the wrist are $\mathbf{x}_w^j = [x_w^j \; y_w^j \; z_w^j]^T$. We can locate the hand (wrist) with respect to the object with the linear transformation:

$$\mathbf{x}_w^j = A\mathbf{x}_c^i \tag{5.8}$$

where $A$ is a transformation matrix. Figure **??** shows the wrist position with respect to the center of mass in pre-grasp action.

## Task

The task is the most important attribute, and many times it decides for itself how to grasp the object; i.e., when grasping a mug containing drink, the usual response is to grasp the handle; when moving the mug, we can grasp the handle or the top. The *task* in the dictionary (Oxfort English Dictionary) is *a piece of work assigned or done as part of one's duties.* Follow this definition and the definition of task analysis.

**Task Analysis:** Task analysis is the analysis or a breakdown of exactly how a task is accomplished, such as what sub-tasks are required.

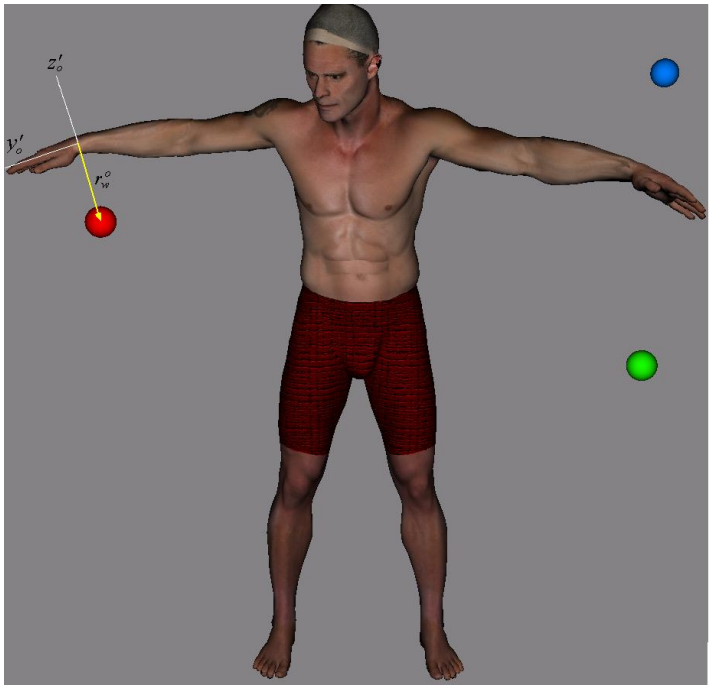


Figure 5.5: Hand position respect the object

We divide tasks and sub-tasks into elemental actions, i.e., open the door is a task, and the elemental action is pulling or push. This elemental action can also be used for other tasks, like moving a joystick in a machine.

### Elementary Actions

**Pull:** To pull is to apply force so as to cause or tend to cause motion toward the source of the force. This action can be done with one, two, three, or four fingers. We define this as $PL_i$, where $i = 1 \ldots 4$ and $PL_1$ means pulling with one finger and so on.

**Push:** To push is to apply pressure against for the purpose of moving. The number of fingers used is the similar as for pulling, but we add the use of the palm. We define this as $PS_i$, where $i = 1 \ldots 5$ and $PS_3$ is pushing with three fingers.

**Pinch:** A pinch is isometric compression between the thumb and the fingers. The number of fingers to used is defined as $PI_i$, where $i = 1 \dots 4$ and $PI_2$ is pinching one object with the thumb and two fingers.

**Power Grasp:** A power grasp is the gripping of an object against the palm. We define this action as $PG$.

**Precision Handling:** Precision handling is the manipulation of an object by thumb and fingers, not in contact with the palm. We define this action as $PH_i$, where $i = 1 \dots 4$ and $PH_4$ is precision handling with the thumb and four fingers.

**Touch:** To touch is to cause or permit a part of the body, especially the hand or fingers, to come into contact with so as to feel. We consider this action is transformed with the index. We define $TO$ for touch.

## Object Initial Position

The point of reference for object initial position is the center of mass (COM) of the object; this point is independent of the type of grasp, but the wrist position is referred to this point. In Figure 5.5 we can see the object position with respect to the wrist. This position in the global coordinates of the virtual environment (VE) tells the system where it is with respect to the global coordinates of the virtual human. Object initial position is defined by:

$$\mathbf{x}_c^i = [x_c^i \; y_c^i \; z_c^i]^T \tag{5.9}$$

## Object Final Position

Knowledge of final position permits us to know a priori if the virtual human can do the task; if it cannot do the task, we need add some actions like walking, advancing the body, etc. To tell the system to add actions,

a check of the workspace helps us determine the reachability of the action. In some tasks, i.e., moving a joystick in a machine to elevate the load, the object is fixed in the base and only moves with the constraints of spherical joints; the final position is always the same as the initial position.

We can use equations similar to those used for determining hand position to determine object final position. We know the object initial position relative to COM $\mathbf{x}_c^i = [x_c^i \; y_c^i \; z_c^i]^T$, and we know the object final position relative to COM $\mathbf{x}_f^i = [x_f^i \; y_f^i \; z_f^i]^T$, both with respect to the global coordinates. The relationship between the initial and final position is:

$$\mathbf{x}_f^i = B\mathbf{x}_c^i \tag{5.10}$$

where $B$ is a transformation matrix.

## One or Two Hands

This parameter is a function of the shape of object. Shape provides information about the size and the weight of object, which determine if one hand or both hands are used or if the action is not performed.

## Number of Fingers

For touching, a virtual human only needs one finger, but for precision handling, the virtual human may need one, two, three, four, or five fingers for grasp. This parameters is a function of the shape and weight of the object.

### Stability

Analysis of stability for some objects helps us to determine whether or not the object in a particular position can be touched. An example is a tall bottle. If the virtual human touch the top of the bottle and it is not stable the botlle can lose stability and fall. Falling not is the final reaction when we touch the object.

From the statics equilibrium, we can calculate if the object's position is stable, unstable, or neutral equilibrium; this is interesting to know for the actions of pulling, pinching, or touching. The general case with one degree of freedom for the object, defined by independent coordinate $s$, is

$$\frac{dV}{ds} = 0,\ \frac{d^2V}{ds^2} > 0\ \textit{stable equilibrium}$$
$$\frac{dV}{ds} = 0,\ \frac{d^2V}{ds^2} < 0\ \textit{unstable equilibrium}$$
$$\frac{dV}{ds} = \frac{d^2V}{ds^2} = \frac{d^3V}{ds^3} = \cdots = 0\ \textit{neutral equilibrium}$$

where $V$ is the potential energy of the object.

### Hand Anthropometry

We give the hand anthropometry for a 95th-percentile human hand in the Chapter 3 related to the hand length and breath. This parametric length for each bone permit the simulation of almost all the population.

## 5.6 Mathematical Model

We have defined task in elemental actions, and these actions are related to some parameters. In this section, we write each elemental action and

show the equations in the function of parameters.

$$Pull(OS, HO, HP, T, OIP, OTH, FN, ST, HA)$$

$$Push(HO, HP, T, OIP, OFP, OTH, FN, ST, HA)$$

$$Pinch(OS, HO, HP, T, OIP, OFP, FN, HA)$$

$$Power\ grasp(OS, HO, HP, T, OTH, FN, HA)$$

$$Precision\ handling(OS, HO, HP, T, FN, ST, HA)$$

where:

$OS$= Object Shape, and the mathematical function is $\mathbf{X}^i(u, v)$

$HO$= Hand Orientation, $\mathbf{Y}^j(u_1, v_1)$

$HP$= Hand Position, $\mathbf{x}_w^j = A\mathbf{x}_c^i$

$T$= Task, user chooses the task

$OIP$= Object Initial Position, $\mathbf{x}_c^i = [x_c^i\ y_c^i\ z_c^i]^T$

$OFP$= Object Final Position, $\mathbf{x}_f^i = [x_f^i\ y_f^i\ z_f^i]^T$

$OTH$= One or Two Hands, a function of OS and represented by, $OTH = f(OS)$

$FN$= Finger Number, $FN = f(OS)$

$ST$= Object Stability, $dV$

$HA$= Hand Anthropometry, $HL$, $HB$

## Pull

Pulling is a function of the object shape because in order to do this action we need to know a priori if is permitted to pull or not. For example, we can pull a door even though it is a big object, but we cannot pull a wall even though it is a similar object shape. The shape of the object can give us additional information, such as which parts of the object to pull.

It is also a function of hand orientation and hand position that if the object is outside the workspace of the hand, the virtual human cannot do this action. However, if the virtual human moves, he can reach the object.

As discussed earlier, task is the most important, because it can determine whether or not the action is pulling.

Object initial position is related to the workspace and hand position, i.e., object and hand position are two parameters that work together to define the object reachability.

When pulling, humans use one or two hands; for realistic actions, virtual humans use similar behavior and sometimes pull a door using two hands and pull a joystick using only one hand. In similar way, we can use a determinate number of fingers; for pulling a door with two hands, the virtual human uses all the fingers, and for pulling a joystick and two fingers are sufficient.

Object stability helps determine if a small variation in the equilibrium position causes the object to lose the equilibrium; or not, e.g., pulling a 1.5-liter bottle of water by the top can cause the bottle to fall, and this is not the final position that the virtual environment is looking for.

Hand anthropometry helps to pull an object and can tell the system which part to pull, e.g., for a door, the handle.

## Push

Knowledge of hand orientation, hand position, task, and object initial position are similar for the action of pull.

If we know object final position in pushing, we can see if this action can completed without additional actions, like walking or moving the virtual human.

The other parameters considerations for pulling apply here.

## Pinch

Let me explain why pinching is function of the parameters shown above. Picture 5.6 shows a lateral pinch for a key; the virtual human is using the thumb and the index finger laterally.

Figure 5.6: Lateral pinch

To pinch a key, is the object shape was analyzed first, then the hand orientation and hand position with respect to the object and then the object initial position and final position were considered. If the action were putting the key in a lock and rotating it ninety degrees to open or close the door, the final position for the object is the lock. Finger number in this case, following

our definition above, is $PL_1$ because only one finger and the thumb is used and pinching always uses the thumb. Finally, the hand anthropometry was considered because in some cases the object with respect to the hand is too big for pinching, in this case, the object and the hand are in good proportion.

### Power Grasp

In the object shape function, if the object is so big compared to the hand anthropometry that the virtual human cannot grasp the whole object,it can look for a part to grasp, e.g., a frying pan was designed for power grasping the handle. Shape can tell us the possibles parts for grasping.

All the other parameters have a similar role, which we describe above for the other basic functions.

### Precision Handling

Precision handling is similar to power grasping with respect to the function parameters. The only remarkable difference is that in this case normally we only use one hand, and do not use the palm. In the function of the action relative to the object, the virtual human can use one, two, or more fingers. At this time the number of fingers to use is related to the object shape and the task to perform.

## 5.7 Grasping Strategy

We have defined all the parameters that work in grasping, and in this section we present the process of grasping. Figure 5.7, presents a flowchart of grasping. The objects are in the virtual environment, and the user chooses

one object from among several. Each object has attached information about several attributes, like task, shape, position of center of mass, etc.; these attributes help the system make a decision. Some objects can do different tasks, e.g., if the object is a mug, the task can be moving or drinking. Therefore, in this first approximation the user chooses the task inherent to the object.

Once the user chooses a task, the system helps make a decision about how to grasp because the system knows the attributes of the object and the task. But maybe the object is not in the workspace of the virtual human; if this is true, we need to tell the virtual human approximatelyhow to put the object in the workspace of the hand.

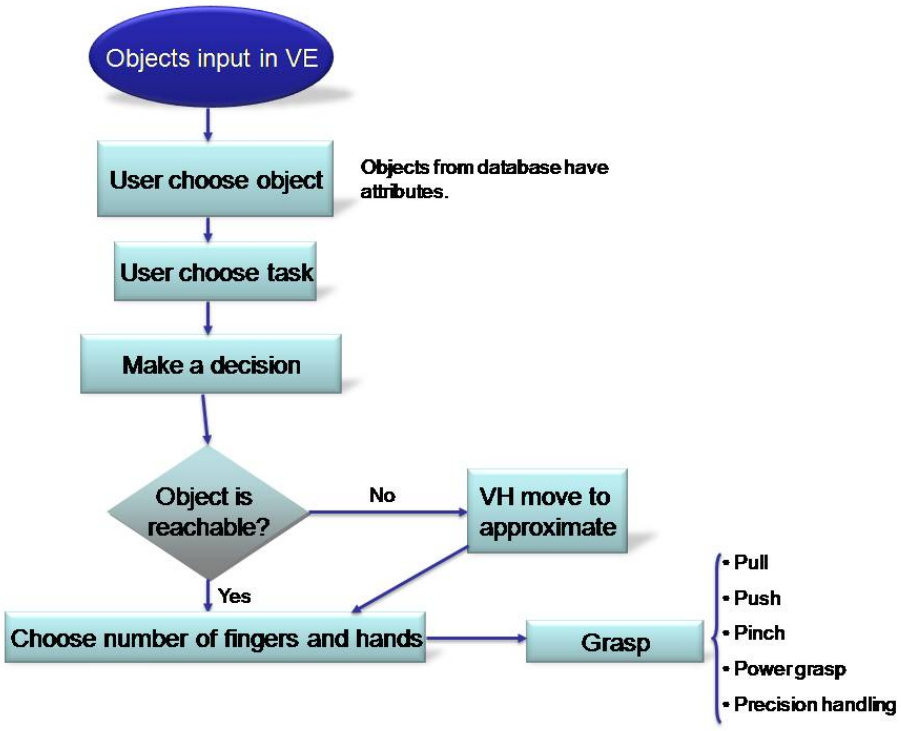


Figure 5.7: Grasping flowchart

With the types of grasp and the shape of the object, the next step is to calculate the number of fingers and hands to use for grasping the object, and then grasp.

This is a heuristic definition for grasping. In the next section, we explain the semi algorithm for grasping the object.

### Objects Input in the Virtual Environment

When the objects are in the virtual environment and the user choose one object to grasp, the system reads all of the attributes and saves them in a file with the extension "*.txt", which allows the file to be used in any code of language. Figure 5.2 shows an object with attributes like center of mass (COM) or, in this case, Inertia center, volume, mass, characteristics, and Inertia matrix.

### User Chooses Object

This action occurs when there are many objects in the virtual environment; if virtual environment has only one object, the virtual human will know the object. One example is that the virtual human is taking a breakfast, and the objects on the table (virtual environment) are a coffee mug, cereal bowl, and spoon. The user chooses a coffee mug, and this inherently has all the properties with the center of mass, of course can change position of center of mass is function of quantity of coffee. For this the object, there are two reasonable tasks, drinking or moving. If the task chosen by the user is drinking, once the virtual human has drunk, it will put the coffee mug back on the table in the same position. During this process, the position of the wrist of the virtual human is known, the position of the center of

mass is known, and we can calculate the hand orientation for performing this action.

## User Chooses Task

This action is mentioned above, and the preceding section has an example of the task and how the user chooses.

## Making a Decision

Inputs in a simple associator are task, weight, shape, temperature, etc., and output pattern is type of grasp, i.e. pinch, pull, etc. When VH take a decision there is based in a system linearly separable.

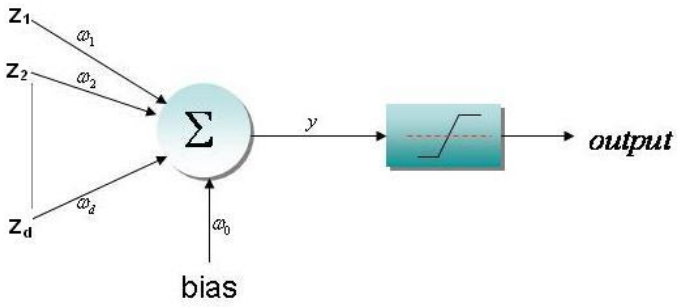


Figure 5.8: Single perceptron

Figure 5.8 show a single perceptron from the works [48], [53], [8] where the input vector is $\mathbf{z} = [z_1, z_2, \ldots, z_d]^T$, bias is $b$, the weight vector considering each input is $\mathbf{w} = [\omega_1, \omega_2, \ldots, \omega_d]^T$, and the weight for the bias is $\omega_0$.

In our case we choose apply these networks for a classification problems, in which the inputs are binary images of attributes like task, shape, temperature, and density. The output of the perceptron is given by

$$\mathbf{y} = g(\sum_{j=1}^{d} \omega_j \phi_j(z) + \omega_0) = g(\mathbf{z}^T \phi) \tag{5.11}$$

where $\phi$ denotes the vector formed from the activations $\phi_0, \ldots, \phi_d$ , and the function of activation in this case is a symmetric saturating linear (satlins), because this are a linearly separable. The input/output relation is:

$$\begin{aligned} g(a) &= -1 \qquad a < -1 \\ g(a) &= a \quad -1 \leq a \leq 1 \\ g(a) &= 1 \qquad n > 1 \end{aligned}$$

## Determining Whether Object is Reachable and Approximating

The answer to this question can be found in Chapter four, where the analysis of workspace for the hand was studied. When checking the workspace for each finger with respect to the wrist position, the virtual human can know if the object with respect to the wrist position is reachable or not. If the object not is reachable, the virtual human can perform some actions to approximate, like walk and reduce the space to a convenient distance between the wrist and the center of mass of the object.

## Choosing Number of Fingers and Hands

Based on object shape and hand anthropometry, we can determine if the use of one hand is adequate or if the use of two hands is required. Based on the geometric primitives for a regular basketball and one person with a 185 mm hand length, the ball cannot be grasped with one hand; two hands are required. Some objects are designed to grasp with both hands.

Normally, if the object to be grasped is based on the primitive geometries for a sphere, cylinder, or box, if the dimensions for grasp are greater than the hand length, it automatically requires two hands.

$$If\ D > HL,\ then\ use\ two\ hands$$

where $D$ is the side dimensions to grasp, and $HL$ is the hand length.

To determine the number of fingers to use for a grasp, parameters like object shape, hand anthropometry, type of grasp, and task are used . We define elemental actions, and each elemental action shows the number of fingers to use. Power grasp and touch do not need the number of fingers defined; normally a power grasp uses five fingers and touch uses the index finger. Other grasp types are a function of parameters mentioned earlier. For precision handling, a primitive sphere is a function of the shape, or better if the radius of equator of sphere is $\rho$ we can define the number of fingers how:

$$If\ 1 <= \rho <= 20\ mm\ then\ PL_1$$

$$If\ 20 <= \rho <= 40\ mm\ then\ PL_2$$

$$If\ 40 <= \rho <= 60\ mm\ then\ PL_3$$

$$If\ 60 <= \rho <= 90\ mm\ then\ PL_4$$

where the subscript 2 means the thumb and two fingers, and subscript 3 means thumb and three fingers, and so on.

We can use the same relationship for pinching. Pulling and pushing are also a function of size of the object; normally, to push a door we use all the fingers, but sometimes if we know the door, we use two or three fingers. For this analysis, we always use five fingers, similar to pulling.

## Grasp

The last action is grasping. Grasping is function of the parameters founded above and forward and inverse kinematics. For several reasons, we prefer to explain the action of grasping in the next chapter and work in several examples demonstrating how this theory works in a virtual human. The output of grasp in this flowchart is the type of grasp and how the grasp uses all the parameters shown in this chapter. Figure 5.9 shows a type of grasp based on the task and other parameters. The positions of fingers are determined as a function of all the parameters defined for grasping, i.e., when grasping a mug to drink from the side, the object in this position has a cylindric shape and the virtual human knows the COM. Based on this, we can calculate the hand position and orientation and the position for all the fingers. In the case shown in Figure 5.9, the task is drinking, the beverage contained in the mug is hot, and the weight is light. In this case, the decision is to grasp for a handle, and the virtual environment knows the position of the handle. The handle has a known shape, and we can calculate the number and position of fingers.


Figure 5.9: Grasping a mug

## 5.8 Conclusions

We have presented a novel theory for grasping based on the objects and their functionality. When the object is selected for the user, it is associated with more parameters, which we describe below. After the user chooses the task, the virtual human, if the object is feasible, grasps with the type of grasp calculated as a function of the mathematical model. The new concept in this chapter is that the virtual human can grasp autonomosly without the user once the task is chosen. Support Vector Machine (SVM) theory, for a perceptron, was applied for this autonomous grasp. The position and orientation of the hand or hands was calculated in reference to the center of mass of the object. Angles for each finger were calculated with the equations of forward and inverse kinematics shown in Chapter 3 for the novel 25-DOF hand model.

Chapter

# 6 A Task-oriented Object Grasping: Implementation

## 6.1 Introduction

In the preceding chapter, we presented a general picture for grasping any object based on its functionality, i.e., every object was designed for use in a determinate task. Based on this theory, this chapter presents implementation, examples, and describes how grasping objects is based on the parameters described in Chapter 5.

## 6.2 Grasp Examples

Our approximation for to grasping is based on the movement of fingers. There are two types of movements. For grasping with power, the movement described, each finger except the thumb, is circular. For the second, when grasping with precision, the movement of fingertips, including the thumb, approximates a circle. Based on these approximations, we can simulate all the human grasping proposed by Buchcholz et al. [21]. Pinching is a particular case of grasping with precision. Pulling, pushing, and touching

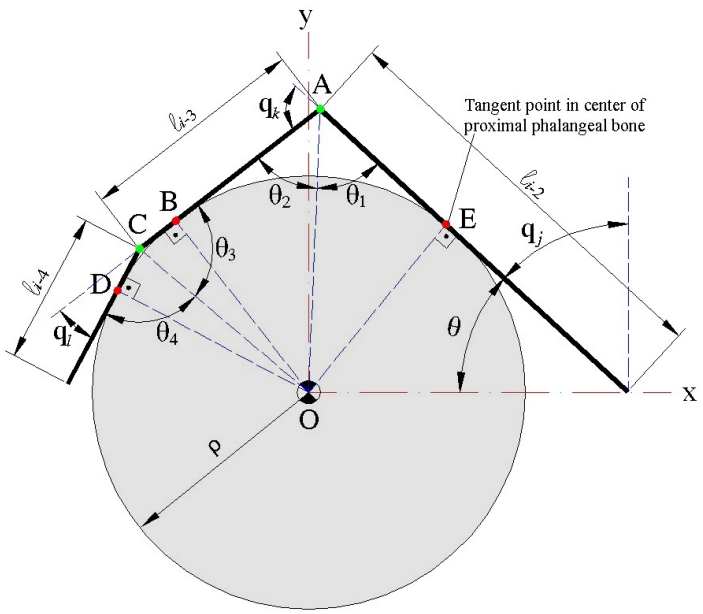


Figure 6.1: The geometry relationship of finger segments

we are considered positioning finger problems. For the first approximation of power grasping we can apply forward kinematics and calculate all the angles for every finger. For a cylinder with radius $\rho$ , Figure 6.1 depicts cross-section of the cylinder and the schematic phalanx bones. The angles $\theta$ and $q_i$ are obtained from the geometry relationship. This example is considered a power.

Where angle $q_j$ for each finger, and subscript is $j = II \ldots V$ where subindex $II$ is for the index, $III$ middle, $IV$ ring, and $V$ small. These angles are for the proximal phalanx respect to the metacarpal bones for each finger. Similar for $q_k$, where $k = II \ldots V$, the subindex mean the same fingers related before and the angles are between proximal phalanx to medial phalanx. For $q_l$, where subscript is $l = II \ldots V$ are referred to angles between middle phalanx and distal phalanx. All of these angles are calculated for geometry and changed from local to global with the transformation

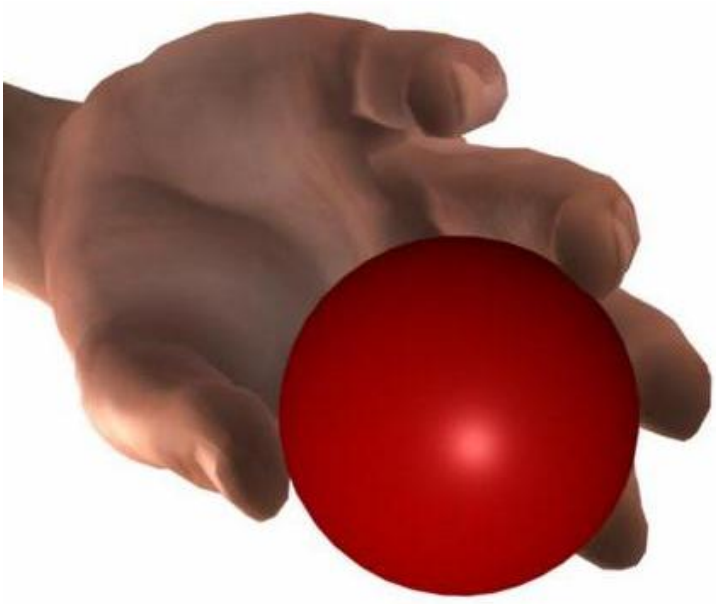

Figure 6.2: Grasping a sphere

matrix developed in Appendix A.

For the second movements, we can calculate the fingertip position for each finger when grasp any object with precision and after applying inverse kinematics. Figures 6.2 and 6.3 depict the fingertip positions.

The angles $\beta$ and $\alpha$ depend on the diameter of the ball. From the observation when people grasp a ball with radius $\rho = 27.5\ mm$ , $\alpha = 0$ and $\beta = 60^o$. Also we impose that the middle finger stays in its neutral position. Therefore, we can determine the fingertip positions for the thumb, index, and ring fingers with respect to the wrist (global) coordinate system, and the small finger stays in the neutral position.

The inverse kinematic solutions depend on the initial values of the design variables for both iterative and optimization-based methods. Table 6.1 presents the solutions ($q_i$ in degrees) for the index finger with the Newton-Raphson method, where the global coordinate is $[-11.22 \quad 152.341 \quad 77.4]$ in

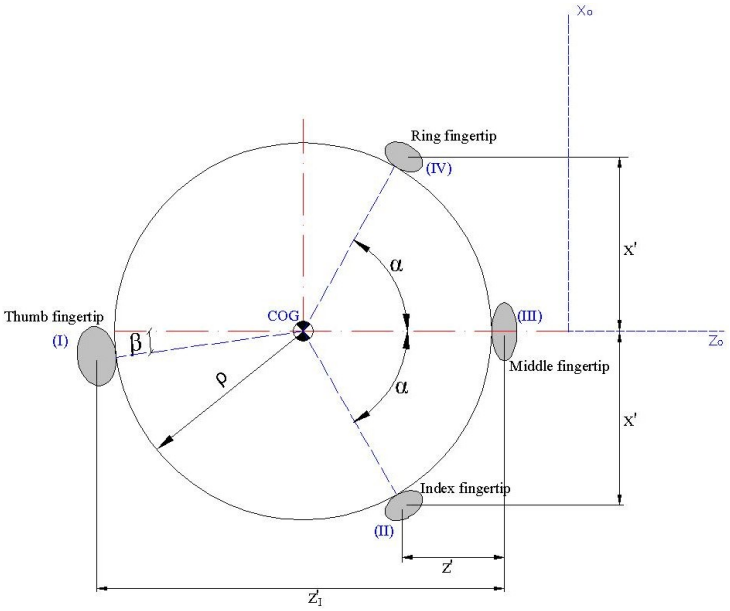


Figure 6.3: Equator section with position of fingertips used

mm, the hand breadth is 200 mm, and the local coordinate is $[-7.3 \;\; 59.9887 \;\; 77.4]$ in mm.

From Table 6.1 it is shown that the convergence for Newton-Raphson method is very fast when the initial angles are close to the solution. For the first set of initial values, the solution for joint DIP ( $q_4$ ) is negative and is in the range of motion. The negative angle for this joint represents hyper-extension. However, usually, we can observe that humans never grasp this sphere by DIP hyper-extension.

In practice, some joints in the fingers are coupled, i.e., the movement of one joint depends on the motion of another joint. For example, each finger except the thumb has two coupled joints. The distal interphalangeal joint (DIP) depends on the proximal interphalangeal joint (PIP) and the relationship is defined in the literature [86], where the superscript $i$ identifies the finger, beginning with the index finger and the last ending with the small

| | | | | |
|---|---|---|---|---|
| Initial | $q_1 = 0$ | $q_2 = 30$ | $q_3 = 30$ | $q_4 = 10$ |
| Iteration=7 | $q_1 = 6.95$ | $q_2 = 39.3$ | $q_3 = 30$ | $q_4 = -7.65$ |
| Initial | $q_1 = 0$ | $q_2 = 0$ | $q_3 = 0$ | $q_4 = 0$ |
| Iteration=10 | $q_1 = 6.95$ | $q_2 = 42.3$ | $q_3 = 10$ | $q_4 = 26.6$ |
| Initial | $q_1 = 0$ | $q_2 = 10$ | $q_3 = 10$ | $q_4 = 0$ |
| Iteration=7 | $q_1 = 6.95$ | $q_2 = 42.3$ | $q_3 = 10$ | $q_4 = 26.6$ |

Table 6.1: Index joint angles with Newton-Raphson method

finger.

$$q^i_{DIP} = \frac{2}{3} q^i_{PIP} \tag{6.1}$$

For the thumb, we can find a similar relationship:

$$q_3 = 2(q_2 - \frac{1}{6}\pi) \tag{6.2}$$

$$q_5 = \frac{7}{5} q_4 \tag{6.3}$$

The subindex shown above can found in Chapter 3.

When we impose the joint coupling function in Equation 6.1 to the index finger, it will have 3 DOF because $q_4$ is a function of $q_3$ . The result is shown in Table 6.2. The solution is the same for all different initial $q_i$ values.

Another approach for finger inverse kinematics is the optimization-based method. The hypothesis is that the feasible space is small for fingers during grasping. The problem will be: Given a point $\mathbf{x}^p$ , find the joint angles $\mathbf{q}^i$ to minimize $\|\mathbf{p}^i - \mathbf{x}^p\|$, and the joint angles should lie in the limits as constraints. Therefore, the formulation is as follows

$$Find : \mathbf{q}_i$$

| | | | | |
|---|---|---|---|---|
| Initial | $q_1 = 0$ | $q_2 = 0$ | $q_3 = 0$ | $q_4 = - - -$ |
| Iteration=9 | $q_1 = 6.95$ | $q_2 = 39.3$ | $q_3 = 20.13$ | $q_4 = 13.42$ |
| Initial | $q_1 = 0$ | $q_2 = 30$ | $q_3 = 30$ | $q_4 = - - -$ |
| Iteration=7 | $q_1 = 6.95$ | $q_2 = 39.3$ | $q_3 = 20.13$ | $q_4 = 13.42$ |
| Initial | $q_1 = 0$ | $q_2 = 10$ | $q_3 = 10$ | $q_4 = - - -$ |
| Iteration=9 | $q_1 = 6.95$ | $q_2 = 39.3$ | $q_3 = 20.13$ | $q_4 = 13.42$ |
| Initial | $q_1 = 0$ | $q_2 = 40$ | $q_3 = 40$ | $q_4 = - - -$ |
| Iteration=7 | $q_1 = 6.95$ | $q_2 = 39.3$ | $q_3 = 20.13$ | $q_4 = 13.42$ |

Table 6.2: Index joint angles with the Newton-Raphson method including joint coupling

$$\begin{aligned} Minimize: \ & \left\| \mathbf{p}^i - \mathbf{x}^p \right\| \\ Subject\ to: \ & q_j^{iL} \leq q_j^i \leq q_j^{iU} \end{aligned} \tag{6.4}$$

A gradient-based optimizer [44] is implemented to solve this problem.

Table 6.3 shows the predicted joint angles for the optimization problem with different initial values. The results show that the optimization problem always has a feasible solution regardless of the initial values. It also shows that and they are the same for all different cases and the final solution is more reasonable than those obtained from the Newton-Raphson method.

| | | | | |
|---|---|---|---|---|
| Initial | $q_1 = 0$ | $q_2 = 30$ | $q_3 = 30$ | $q_4 = 10$ |
| Solution | $q_1 = 6.95$ | $q_2 = 39.22$ | $q_3 = 24.54$ | $q_4 = 6.24$ |
| Initial | $q_1 = 0$ | $q_2 = 0$ | $q_3 = 0$ | $q_4 = 0$ |
| Solution | $q_1 = 6.95$ | $q_2 = 39.36$ | $q_3 = 23.37$ | $q_4 = 8.47$ |
| Initial | $q_1 = 0$ | $q_2 = 10$ | $q_3 = 10$ | $q_4 = 0$ |
| Solution | $q_1 = 6.95$ | $q_2 = 39.22$ | $q_3 = 24.56$ | $q_4 = 6.22$ |

Table 6.3: Index joint angles with the optimization-based method

## 6.3 Making a Decision

In this section we provide an example applying the Support Vector Machine (SVM) presented in the last chapter. The example shown here can be extended to any case, without losing generality. This example consists of putting an object in our virtual environment, in this case a mug. If there are more objects in the virtual environment, the user chooses the mug. Based on the functionality of mug, it has only two associated tasks, one is drinking and the other is moving. The user in this case chooses between these two tasks. In this case, to simplify the example, the known input parameters for this object are as follows, the others are 0:

$$Task \begin{cases} Drink = 1 \\ Move = -1 \end{cases}$$

$$Temperature \begin{cases} Hot = 1 \\ Normal = -1 \end{cases}$$

$$Weight \begin{cases} Heavy = 1 \\ Normal = -1 \end{cases}$$

We can add all the parameters, inherent to the object and described in Section 5.6, such a shape, hand orientation, number of fingers, etc. The problem become a classification problem between two classes.

$$y = hardlim\mathbf{g}(W^T\mathbf{p} + b)$$

If we choose $x = 0$, axis for classification and $b = 0$ the bias, the weight input vector is

$$W^T = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0].$$

The output, or decision, is to grasp the side or grasp the handle. In this case, we found a problem: if the mug is hot and the task is move, the virtual human cannot grasp the side of the mug. For this reason, we need to connect these outputs to another preceptron and build another output. Figure 6.4 shows the proposed network connected in parallel for the problem of output that is not consistent with the inputs. In this case, the perceptrons connected in parallel are $p = t - 1$, where $t$ is the number of types of grasp inherent to the object in the function of the task.

## 6.4 Choosing Number of Fingers and Hands

In this example, the parameters for grasping are any of the three ways to grasp a mug: by the side, top, or handle. The dimensions of the side to grasp are smaller than the hand length, and the virtual human uses only one hand. For the number of fingers to use, the parameter used for the first two types is $\rho = 80\ mm$. From Chapter 5, the number of fingers is $PL_4$,

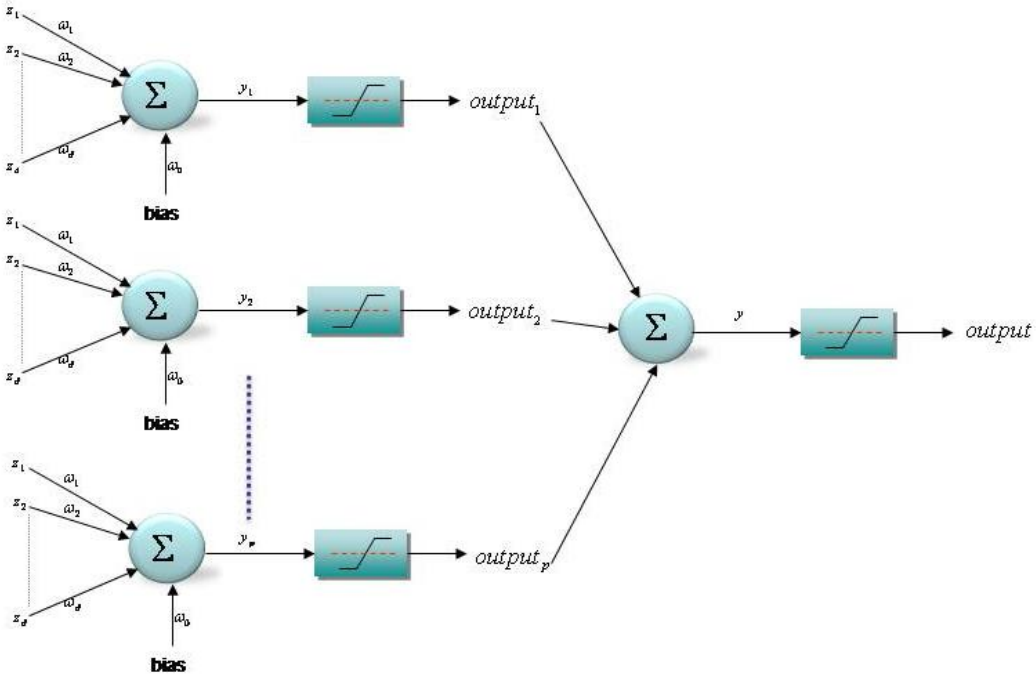


Figure 6.4: Perceptrons connected in parallel

which means the thumb and four fingers. When the decision is to grasp the handle, the parameter is $\rho = 50\ mm$ and the finger number is $PL_3$, that is, index, middle, ring, and thumb. This grasp is shown in Figure 5.9.

## 6.5 Grasping

The last action is grasping, once the user chooses the object among others in the virtual environment and the task, some objects only has designed for one task, or only there is one object in the virtual environment, if there is more than one choice. After, the algorithm take automatically makes a decision, choosing the number of fingers and hands, calculating the wrist position and orientation, and calculating the angles of every joint in our 25-DOF model. Wrists are connected with the upper body, and with this

information $Santos^{TM}$ can grasp with his whole body in a realistic posture.

Figure 6.5 shows the results of applying the task-oriented object semi-intelligent system for grasping a mug. For this example, the three parameters drink, normal, and light, are choosen randomly, without lost generality. We can leave it so the user chooses the task. Once the decision is to grasp the handle, this is the result.

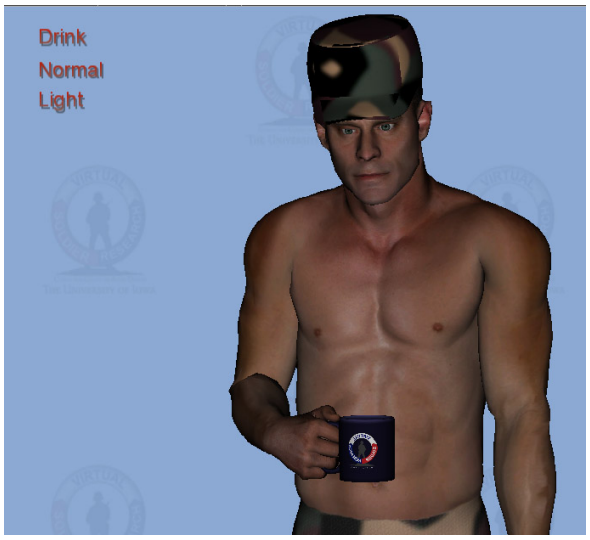


Figure 6.5: Grasping a mug

Other types of grasp for different types of tasks and parameters for a mug are shown in Figures 6.6 and 6.7. Figure 6.6 makes a decision shows a grasp with power, and Figure 6.7 grasp shows grasping the top with precision.

Similar process and results are shown in Figures 6.8 and 6.9 for a joystick. In this case, there are only two tasks and the parameters are inherent to the joystick.


Figure 6.6: Grasping a mug; power grasp


Figure 6.7: Grasping a mug; precision grasp

## 6.6 Conclusions

Implementation of the theory presented in Chapter 5 was made here. Based on the fact every object has some inherent parameters, attributes, and tasks which they are designed, the objects can be grasped in different

Figure 6.8: Grasping a joystick; power grasp

Figure 6.9: Grasping a joystick; power grasp

situations and positions connected with the upper body. Semi-intelligent task-oriented grasping object was implemented in Vitools$^{TM}$ and demonstrates that the proposed algorithm works with several objects.

Chapter

# 7 Conclusions and Future Work

## 7.1 Conclusions

We finalized each chapter with a section of conclusions. In this chapter we present general conclusions for this dissertation.

- Based on the exhaustive study of human hand anatomy, we developed a 25-DOF virtual hand model. This model was connected with the upper body of $Santos^{TM}$, a virtual human developed in The University of Iowa. This high number of degrees of freedom permits arching the palm of our virtual human.

- The Denavit-Hartenberg (D-H) method was implemented in a 25-DOF hand model and applied forward and inverse kinematics to calculate the joint angles for each finger $[q_1 \dots q_{25}]$. An algorithm for transforming the local coordinates to global coordinates was implemented, to connect with the wrist. Forward kinematics was based on geometric models and inverse kinematics in iterative the Newton-Raphson (N-R), and optimization methods.

- When applied forward and inverse kinematics, the output was the joint angles. Knowledge of joint angles, give new tools for creating a new prosthetics hands.

- Row-rank deficiency criteria was applied to check the reachability of an object in the analysis of workspace. A parametric model for each finger was developed, and implemented. Visualization workspace was shown for the thumb with 5 DOF, the index finger with 4 DOF, and the ring finger with 6 DOF. The middle finger has the same number of DOF as the index finger and small finger as the same as the of ring finger.

- Task-oriented object grasping was developed and implemented based on attributes inherent to objects. Task was the most important attribute, and based on these we developed a semi-intelligent algorithm for grasping any object in the virtual environment. The user chooses the object and task. Based on this election, the Vector Support Machine (SVM), with a hardlim function of activation, the virtual human makes a decision about how to grasp the object.

- A mathematical model for the previous actions was defined. We define the actions (pulling, pushing, pinching, power grasping, and precision handling) as a function of parameters. These parameters are object shape, hand orientation, hand position, object initial position, object final position, finger number, number of hands, object stability, and hand anthropometry.

- A grasping flowchart was developed and implemented. The steps in the flowchart are as follows. Objects are input in virtual environment. The user chooses one of the objects to grasp, and each object

has some inherent task. The user chooses the task. Routine calculations are performed and the virtual human makes a decision about grasp. Based on workspace analysis, the virtual human verifies the reachability of the object and calculates the number of fingers and hands to uses, based on the attributes of the objects. Grasping is the last action based on all the previous steps in the grasping flowchart.

## 7.2 Future work

This dissertation developed a novel method for grasping, but while we were finishing this development, new questions appeared. For future work to continue this research, we propose:

- Fingers were considered rigid bodies. The application of a soft finger, and the analysis of deformations of fingers can add more realistic grasping.

- New models of vision applied to the virtual environment can eliminated part of the grasping flowchart.

- Integrate muscles in our virtual hand model. Total joint torques due to several external loads are obtained as a sum of all joint torques:

$$\tau = \sum_{k} \mathbf{J}_k^{\mathbf{T}} \mathbf{F}_k$$

If muscles are added, the study of carpal syndrome and other diseases related with to the hand can be performed in our virtual environment.

- Develop new finger models with sensibility, and imitate the tact of human. That can help to autonomously know attributes such as a temperature.

- After development of the grasping strategy and simulation, the next step is manipulation and the considering of object stability during manipulation.

# Bibliography


[1] K. Abdel-Malek, H.-J. Adkins, F. Yeh, and E. Haug. On the determination of boundaries to manipulator workspaces. *Robotics and Computer-Integrated Manufacturing*, 13(1):63–72, 1997.

[2] K. Abdel-Malek and S. Othman. Multiple sweeping using the denavit-hartenberg representation method. *Computer-Aided Design*, 31(9):567–583, 1999.

[3] K. Abdel-Malek and H.-J. Yeh. Analytical boundary of the workspace for general three degree-of-freedom mechanisms. *International Journal of Robotics Research*, 16(2):198–213, 1997.

[4] K. Abdel-Malek and H.-J. Yeh. Path trajectory verification for robot manipulators in a manufacturing environment. *IMechE Journal of Engineering Manufacture*, (211):547–556, 1997.

[5] K. Abdel-Malek and H.-J. Yeh. Crossable surfaces of robotic manipulators with joint limits. *ASME Journal of Mechanical Design*, 122(1):52–61, 2000.

[6] A. Al-Fahed Nuseirat and R. Abu-Zitar. Neural network approach to firm grip in the presence of small slips. *Journal of Robotic Systems*, 18(6):305–215, 2001.

[7] I. Albrecht, J. Haber, and H.-P. Seidel. Construction and animation of anatomically based human hand models. In *Eurographics/SIGGRAPH Symposium on Computer Animation*, pages 1–12, 2003.

[8] J. A. Anderson. *An Introduction to Neural Networks*. Bradford Book. Third edition, 1997.

[9] J. Angeles. *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*. Mechanical Engineering Series. Second edition, 2002.

[10] Y. Aydin and M. Nakajima. Database guided computer animation of human grasping using forward and inverse kinematics. *Computers and Graphics*, 23(1):145–154, 1999.

[11] F. Barbagli, A. Frisoli, K. Salisbury, and M. Bergamasco. Simulating human fingers: a soft finger proxy model and algorithm. In *HAPTICS '04. Proceedings. 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 9–17, 2004.

[12] K. Bernardin, K. Ogawara, K. Ikeuchi, and R. Dillmann. A sensor fusion approach for recognizing continuous human grasping sequences using hidden markov models. *IEEE Transactions on Robotics*, 21(1):47–57, 2005.

[13] A. Bicchi. On the closure properties of robotics grasping. *The International Journal of Robotics Research*, 14(4):319–334, 1995.

[14] A. Bicchi and A. Marigo. Rolling contacts and dexterous manipulation. In *IEEE International Conference on Robotics and Automation*, pages 282–287, 2000.

[15] C. Borst, M. Fischer, and G. Hirzinger. A fast and robust grasp planner for arbitrary 3d objects. In *Proceedings. IEEE Iternational Conference on Robotics and Automation*, volume 5, pages 1890–1896 vol.5, 1999.

[16] C. Borst, M. Fischer, and G. Hirzinger. *Calculating hand configurations for precision and pinch grasps*. PhD thesis, 2002.

[17] C. Borst, M. Fischer, and G. Hirzinger. Grasp planning: How to choose a suitable task wrench space. In *Proceedings. IEEE Iternational Conference on Robotics and Automation*, volume 4, pages 319–325, 2004.

[18] D. Bowers and R. Lumia. Manipulation of unmodeled objects using intelligent grasping schemes. *IEEE Transactions on Fuzzy Systems*, 11(3):320–330, 2003.

[19] P. Braido and X. Zhang. Quantitative analysis of finger motion coordination in hand manipulative and gestic acts. *Human Movement Science*, 22:661–678, 2004.

[20] P. W. Brand and A. M. Hollister. *Clinical Mechanics of the Hand*. Mosby, Inc. Third edition, 1999.

[21] B. Buchcholz, T. Armstrong, and S. Goldstein. Anthropometric data for describing the kinematics of the human hand. *Ergonomics*, 35(3):261–273, 1992.

[22] M. Buss, H. Hashimoto, and J. B. Moore. Grasping force optimization for multi-fingered robot hands. *IEEE International Conference on Robotics and Automation*, 0-7803-1965-6/95:1034–1039, 1995.

[23] M. Buss, H. Hashimoto, and J. B. Moore. Dextrous hand grasping force optimization. *IEEE Transactions on Robotics and Automation*, 12(3):406–418, 1996.

[24] J. ButterfaB, M. Grebenstein, H. Liu, and G. Hirzinger. Dlr-hand ii: Next generation of a dextrous robot hand. In *Proceedings. IEEE International Conference on Robotics and Automation*, volume 1, pages 109–114, 2001.

[25] A. Caffaz and G. Cannata. The design and development of the dist-hand dextrous gripper. In *Proceedings. IEEE International Conference on Robotics and Automation*, volume 5, pages 2075–2080 vol.5, 1998.

[26] G. Casalino, G. Cannata, G. Panin, and A. Caffaz. On a two-level hierarchical structure for the dynamic control of multifingered manipulation. In *Proceeding. IEEE International Conference on Robotics and Automation*, volume 0-7803-6475-9/01, pages 77–84, 2001.

[27] J. Chalfoun, R. Younes, M. Renault, and F. Ouezdou. Forces, activation and displacement prediction during free movement in the hand and forearm. *Journal of Robotics Systems*, 22(11):653–660, 2005.

[28] F.-T. Cheng and D. E. Orin. Efficient algorithm for optimal force distribution-the compact-dual lp method. *IEEE Transactions on Robotics and Automation*, 6(2):178–187, 1990.

[29] J. Cornella and R. Suarez. Fast and flexible determination of force-closure independent regions to grasp polygonal objects. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA 2005*, ISBN 0-7803-8915-8:778–783, 2005.

[30] J. Cornella and R. Suarez. A new framework for planning three-finger grasps of 2d irregular objects. *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'2006*, ISBN 1-4244-0259-X:5688–5694, 2006.

[31] J. Cornella, R. Suarez, R. Carloni, and C. Melchiorri. Grasping force optimization using dual methods. *8th IFAC Symposium on Robot Control, SYROCO 2006*, 2006.

[32] M. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Transactions on Robotics and Automation*, 5(3):269–279, 1989.

[33] W. G. Darling and K. J. Cole. Muscle activation patterns and kinetics of human index finger movements. *Journal of Neurophysiology*, 65(5):1098–1108, 1990.

[34] W. G. Darling, K. J. Cole, and G. F. Miller. Coordination of index finger movements. *Journal of Biomechanics*, 27(4):479–491, 1994.

[35] N. A. Davidoff and A. Freivalds. A graphic model of the human hand using catia. *International Journal of Industrial Ergonomics*, 12:255–264, 1993.

[36] J. Denavit and R. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *Journal of Applied Mechanics*, 22:215–221, 1955.

[37] D. Ding, Y.-H. Lee, and S. Wang. Computation of 3-d form-closure grasps. *IEEE Transactions on Robotics and Automation*, 17(4):515–522, 2001.

[38] M. J. Dodson, A. W. Goodwin, A. S. Browning, and H. M. Gehring. Peripheral neural mechanisms determining the orientation of cylinders grasped by the digits. *J. Neurosci.*, 18(1):521–530, 1998.

[39] R. G. Dong, J. Z. Wu, T. W. McDowell, D. E. Welcome, and A. W. Schopper. Distribution of mechanical impedance at the fingers and the palm of the human hand. *Journal of Biomechanics*, 38(5):1165–1175, 2005.

[40] G. Farin, J. Hoschek, and M.-S. Kim. *Handbook of COMPUTER AIDED GEOMETRIC DESIGN*. Elsiever. 1 edition, 2002.

[41] C. Ferrari and J. Canny. Planning optimal grasps. In *Proceedings. IEEE International Conference on Robotics and Automation*, pages 2290–2295 vol.3, 1992.

[42] J. Freund, R. Toivonen, and E.-P. Takala. Grip forces of the fingertips. *Clinical Biomechanics*, 17(7):515–520, 2002.

[43] N. Fukaya, S. Toyama, T. Asfour, and R. Dillmann. Design of the tuat/karlsruhe humanoid hand. pages 1–6, 2000.

[44] P. Gill, W. Murray, and A. Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM Journal of Optimization*, 12(4):979–1006, 2002.

[45] F. Giorgi, A. Caffaz, G. Casalino, and A. Turetta. Fpga-based technology for modular control of anthropomorphic robotics hands. In *IEEE*, pages 387–391, 2004.

[46] Y. Guan and H. Zhang. Kinematic feasibility analysis of 3-d multifingered grasps. *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, 19(3).

[47] Y. Guan and H. Zhang. Feasibility analysis of 2d grasps. In *(IROS 2003). Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 4, pages 3435–3440 vol.3, 2003.

[48] M. T. Hagan, H. B. Demuth, and M. H. Beale. *Neural network design.* Boston : PWS Pub. 1996.

[49] C. Hager-Ross and M. H. Schieber. Quantifying the independence of human finger movements: Comparisons of digits, hands, and movement frequencies. *J. Neurosci.*, 20(22):8542–8550, 2000.

[50] L. Han, J. Trinkle, and Z. Li. Grasp analysis as linear matrix inequality problems. In *Proceedings. IEEE International Conference on Robotics and Automation*, volume 2, pages 1261–1268 vol.2, 1999.

[51] L. Han, J. Trinkle, and Z. Li. Grasp analysis as linear matrix inequality problems. *IEEE Transactions on Robotics and Automation*, 16(6):663–674, 2000.

[52] M. Hanes, S. Ahalt, K. Mirza, and D. Orin. Neural network control of force distribution for power grasp. In *Proceedings.IEEE International Conference on Robotics and Automation*, pages 746–751 vol.1, 1991.

[53] S. Haykin. *Neuronal Network. A Comprensive Foundation.* Prentice Hall international, Inc. Second edition, 1999.

[54] J. Hoschek and D. Lasser. *Fundamentals of COMPUTER AIDED GEOMETRIC DESIGN.* AK Peters. 1 edition, 1993.

[55] W. Howard and V. Kumar. On the stability of grasped objects. *IEEE Transactions on Robotics and Automation*, 12(6):904–917, 1996.

[56] C. Huang and S. Xiao. Geometric modeling of human hand for muscle volume conductor computation. In *Proceeding. 19th International Conference - IEEE/EMBS*, pages 2078–2081 vol.3, 1997.

[57] Y.-B. Jia. Computation on parametric curves with application in grasping. *The International Journal of Robotics Research*, 23(7-8):825–855, 2004.

[58] S. Kamojima, N. Miyata, and J. Ota. Identification of position and orientation of hand bones from mr images by bones model registration. In *Proceedings. IEEE International Intelligent Robots and Systems*, pages 2021–2027, 2004.

[59] M. Kaneko and K. Tanie. Contact point detection for grasping an unknown object using self-posture changeability. *IEEE Transactions on Robotics and Automation*, 10(3):355–367, 1994.

[60] S. Kang and K. Ikeuchi. Toward automatic robot instruction from preception-mapping human grasps to manipulator grasps. *IEEE Transactions on Robotics and Automation*, 13(1):81–95, 1997.

[61] I. Kapandji. *Fisiologia articular. Miembro superior.* Medica Panamericana, Madrid. 5 edition, 1996.

[62] S. L. Kilbreath, R. B. Gorman, J. Raymond, and S. C. Gandevia. Distribution of the forces produced by motor unit activity in the human flexor digitorum profundus. *J Physiol (Lond)*, 543(1):289–296, 2002.

[63] D. Kragic, A. Miller, and P. Allen. Real-time tracking meets online grasp planning. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*, volume 3, pages 2460–2465 vol.3, 2001.

[64] D. M. Lane, J. B. C. Davies, G. Robinson, D. J. OBrien, J. Sneddon, E. Seaton, and A. Elfstrom. The amadeus dextrous subsea hand: Design, modeling, and sensor processing. *Journal of Oceanic Engineering*, 24(1):96–111, 1999.

[65] H. Liu, J. Butterfass, S. Knoch, P. Meusel, and G. Hirzinger. A new control strategy for dlr's multisensory articulated hand. In *Proceedings. 1999 IEEE Control Systems*, pages 47–54, 1999.

[66] H. Liu, T. Iberall, and G. Bekey. Neural network architecture for robot hand control. *IEEE Control Systems Magazine*, 9(3):38–43, 1989.

[67] Y. Liu, G. Starr, J. Wood, and R. Lumia. Spatial grasp synthesis for complex objects using model-based simulation. *Industrial Robot: An International Journal*, 32(1):24–31, 2005.

[68] A. Miller, P. Allen, V. Santos, and F. Valero-Cuevas. From robotic hands: a visualization and simulation engine for grasping research. *Industrial Robot: An International Journal*, 32(1):55–63, 2005.

[69] N. Miyata, M. Kouchi, T. Kurihara, and M. Mochimaru. Modeling of human hand link structure from optical motion capture data. In *(IROS 2004). Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2129–2135 vol.3, 2004.

[70] N. Miyata, M. Kouchi, and M. Mochimaru. Posture estimation for design alternative screening by dhaibahand - cell phone operation. In *SAE 2006 Digital Human Modeling for Design and Engineering Conference*, pages pp.2006–01–2327, 2006.

[71] N. Miyata, M. Kouchi, and M. Mochimaru. Generation and validation of 3d links for representative hand models. In *SAE 2007 Digital Human Modeling for Design and Engineering Conference*, pages pp.2007–01–2512, 2007.

[72] N. Miyata, M. Kouchi, M. Mochimaru, K. Kawachi, and T. Kurihara. Hand link modeling and motion generation from motion capture data based on 3d joint kinematics. In *SAE 2005 Digital Human Modeling for Design and Engineering Conference*, pages pp.2005–01–2687, 2005.

[73] J. Molina-Vilaplana and J. Lopez-Coronado. A neural network model for coordination of hand gesture during reach to grasp. *Neural Network*, (19):12–30, 2006.

[74] M. A. Moussa. Combining expert neural networks using reinforcement feedback for learning primitive grasping behavior. *IEEE Transactions on Neural Networks*, 15(3):629–638, 2004.

[75] K. Nagai and T. Yoshikawa. Dynamic manipulation/grasping control of multifingered robot hands. In *Proceedings. IEEE International Conference on Robotics and Automation*, pages 1027–1033 vol.3, 1993.

[76] K. Nagai and T. Yoshikawa. Grasping and manipulation by arm/multifingered-hand mechanisms. In *Proceedings. IEEE International Conference on Robotics and Automation*, volume 1, pages 1040–1047 vol.1, 1995.

[77] T. Nagashima, H. Seki, and M. Tanako. Analysis and simulation of grasping/manipulation by multi-fingersurface. *Mechanism and Machine Theory*, 32(2):175–191, 1997.

[78] C. P. Neu, J. J. Crisco, and S. W. Wolfe. In vivo kinematic behavior of the radio-capitate joint during wrist flexion-extension and radio-ulnar deviation. *Journal of Biomechanics*, 34(11):1429–1438, 2001.

[79] C. Nolker and H. Ritter. Visual recognition of continuous hand postures. *IEEE Transactions on Neural Networks*, 13(4):983–994, 2002.

[80] T. Omata and K. Nagata. Rigid body analysis of the indeterminate grasp force in power grasps. In *Proceedings.IEEE International Conference on Robotics and Automation*, volume 2, pages 1787–1794 vol.2, 1996.

[81] T. Omata and K. Nagata. Rigid body analysis of the indeterminate grasp force in power grasps. *IEEE Transactions on Robotics and Automation*, 16(1):46–54, 2000.

[82] A. E. Park, J. J. Fernandez, K. Schmedders, and M. S. Cohen. The fibonacci sequence: Relationship to the human hand. *The Journal of Hand Surgery*, 28(1):157–160, 2003.

[83] V. Pavlovic, R. Sharma, and T. Huang. Visual interpretation of hand gestures for human-computer interaction: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.

[84] J. L. Pearlman, S. S. Roach, and F. J. Valero-Cuevas. The fundamental thumb-tip force vectors produced by the muscles of the thumb. *Journal of Orthopaedic Research*, 22(2):306–312, 2004.

[85] J. Ponce and B. Faverjon. On computing three-finger force-closure grasps of polygonal objects. *IEEE Transactions on Robotics and Automation*, 11(6):868–881, 1995.

[86] H. Rijpkema and M. Girard. Computer animation of knowledge-based human grasping. *Computer Graphics*, 25(4):339–348, 1991.

[87] M. Roa and R. Suarez. Determination of independent contact regions on discretized 3d objects. In *IEEE International Symposium Conference on Assembly and Manufacturing, ISAM'2007*, volume ISBN 1-4244-0563-7.

[88] F. Saito and K. Nagata. Interpretation of grasp and manipulation based on grasping surfaces. In *Proceedings. IEEE International Conference on Robotics and Automation*, volume 2, pages 1247–1254 vol.2, 1999.

[89] X. Sancho-Bru. *Model biomecanic de la ma orientat al disseny d'eines manuals*. PhD thesis.

[90] A. Savescu, L. Cheze, X. Wang, G. Beurier, and J. P. Verriest. A 25 degrees of freedom hand geometrical model for better hand attitude simulation. In *SAE International. 2004-01-2196*, 2004.

[91] S. E. Sonenblum, J. J. Crisco, L. Kang, and E. Akelman. In vivo motion of the scaphotrapezio-trapezoidal (stt) joint. *Journal of Biomechanics*, 37(5):645–652, 2004.

[92] R. Suarez and P. Grosch. Mechanical hand ma-i as experimental system for grasping and manipulation. *Video Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, ICRA:ISBN 0–7803–8915–8, 2005.

[93] M. Svinin, K. Ueda, and M. Kaneko. On the stability of an object in multi-finger grasping. In *Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 412–417, 1999.

[94] R. Tubiana. *The hand. Volume I.* W.B. Saunders Company. 2 edition, 1981.

[95] R. Tubiana, J. Thomine, and E. Mackin. *Examination of the hand and wrist.* Martin Dunitz. 2 edition, 1996.

[96] F. J. Valero-Cuevas. Predictive modulation of muscle coordination pattern magnitude scales fingertip force magnitude over the voluntary range. *J Neurophysiol*, 83(3):1469–1479, 2000.

[97] F. J. Valero-Cuevas. An integrative approach to the biomechanical function and neuromuscular control of the fingers. *Journal of Biomechanics*, 38(4):673–684, 2005.

[98] F. J. Valero-Cuevas, M. Johanson, and J. D. Towles. Towards a realistic biomechanical model of the thumb: the choice of kinematic description may be more critical than the solution method or the variability/uncertainty of musculoskeletal parameters. *Journal of Biomechanics*, 36:1019–1030, 2003.

[99] W. Wang and K. Wang. Geometric modeling for swept volume of moving solids. *IEEE CG and A*, 86:8–17, 1986.

[100] R. Yokogawa and K. Hara. Measurement of distribution of maximum index-fingertip force in all directions at fingertip in flexion/extension plane. *Journal of Biomechanical Engineering*, 124:302–307, 2002.

[101] Y. Yu, Y. Li, and S. Tsujio. Analysis of finger position regions on grasped object with multifingered hand. In *Proceedings of the IEEE International Symposium on Assembly and Task Planning*, pages 178–183, 2001.

[102] I. Zeid. *Mastering CAD/CAM*. McGraw-Hill, Higher Education. 1 edition, 2005.

[103] X.-Y. Zhang, Y. Nakamura, K. Goda, and K. Yoshimoto. Robustness of power grasp. *IEEE*, 1050-4729/9:2828–2835, 1994.

[104] Y. Zhang, F. Gao, and W. Gruver. Determination of contact forces in grasping. In *IROS 96, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 1038–1044 vol.3, 1996.

[105] Y. Zhang and W. Gruver. Definition and force distribution of power grasps. In *Proceedings. IEEE International Conference on Robotics and Automation*, volume 2, pages 1373–1378 vol.2, 1995.

[106] Y. Zhang, W. A. Gruver, J. Li, and Q. Zhang. Classification of grasps by robot hands. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICSPART B: CYBERNETICS*, 31(3):436–444, 2001.

[107] X. Zhu and H. Ding. Planning force-closure grasps on 3-d objects. In *Proceedings. ICRA '04. IEEE International Conference on Robotics and Automation*, volume 2, pages 1258–1263 Vol.2, 2004.

[108] X. Zhu, H. Ding, and J. Wang. Grasp analysis and synthesis based on a new quantitative measure. *IEEE Transactions on Robotics and Automation*, 19(6):942–953, 2003.

[109] X. Zhu, H. Ding, and M. Wang. A numerical test for the closure properties of 3-d grasps. *IEEE Transactions on Robotics and Automation*, 20(3):543–549, 2004.

Appendix

# A Transformation Matrices

## Index Finger

$$
{}^{0}H_{2} = \begin{bmatrix} \cos\gamma_2 & \sin\gamma_2 & 0 & -l_{oo_2}\sin\gamma_2 \\ -\sin\gamma_2 & \cos\gamma_2 & 0 & l_{oo_2}\cos\gamma_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

## Middle finger

$$
{}^{0}H_{3} = \begin{bmatrix} \cos\gamma_3 & \sin\gamma_3 & 0 & 0 \\ -\sin\gamma_3 & \cos\gamma_3 & 0 & l_{oo_3} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

## Ring finger

$$ {}^{0}H_4 = \begin{bmatrix} \cos\gamma_4 & \sin\gamma_4 & 0 & l_{oo_4}\sin\gamma_4 \\ -\sin\gamma_4 & \cos\gamma_4 & 0 & l_{oo_4}\cos\gamma_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

## Small finger

$$ {}^{0}H_5 = \begin{bmatrix} \cos\gamma_5 & \sin\gamma_5 & 0 & l_{oo_5}\sin\gamma_5 \\ -\sin\gamma_5 & \cos\gamma_5 & 0 & l_{oo_5}\cos\gamma_5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Appendix

# B Research Publications

## Journal Papers

1. Yang, J., Peña Pitarch, E., Kim, J., and Abdel-Malek, K. Posture prediction and Force/Torque Analysis for Human Hand SAE technical paper series , 1-10. ISSN: 0148-7191, 2006.

2. Yang, J., Peña Pitarch, E., Potratz, J., Beck, S., and Abdel-Malek, K. Synthesis and Analysis of a Flexible Elephant Trunk Robot. ADVANCED ROBOTICS , 20 (6) : 631-659. ISSN: 0169-1864, 2006.

3. Potratz, J., Yang, J., Abdel-Malek, K., Peña Pitarch, E., and Grosland, N. A Light Weight Compliant Hand Mechanism with High Degrees of Freedom. ASME Journal of Biomechanical Engineering, 127 (934) : 1-12. ISSN: 0148-0731, 2005.

4. Jingzhou Yang, Esteban Peña Pitarch, Karim Abdel-Malek, Amos Patrick, and Lars Lindkvist. A Multi-Fingered Hand Prosthesis. Mechanism and Machine Theory,39(6), 2004.

## Conference Papers

1. Peña-Pitarch, E., Abenoza Guardiola, M., Vilanova Masana, R., Yang, J., and Abdel-Malek, K. Hand Grasp: Analysis and Experiments 8ème Congrès de Mécanique. Mécanique des Solides. VOLUME I. SMSM. SOCIET MAROCAINE DES SCIENCES MCANIQUES. EL JADIDA. MAROC., p. 418-420, 2007.

2. Jingzhou Yang, Esteban Pea Pitarch, Joo Kim, and Karim Abbel-Malek. Posture Prediction and Force/Torque Analysis for Human Hands. Digital Human Modeling for Design and Engineering. Conference Proceedings. SAE2006, 2006, p. 1-8. Lyon (FRANCE), 2006.

3. Peña Pitarch, E., Yang, J., Abdel-Malek, K., Kim, J., and Marler, T. Joystick Ergonomic Study in Material Handling. ASME International Engineering Congress, November 5-11, Orlando, Florida, USA, 2005.

4. Peña Pitarch, E., Yang, J., Abdel-Malek, K., and Marler, T. Hand Grasping Strategy of Virtual Humans. The 3rd IASTED International Conference on Biomechanics, September 7-9, Benidorm, Spain, 2005.

5. Peña-Pitarch, E., Yang, J., and Abdel-Malek, K. $SANTOS^{TM}$ Hand: A 25 Degree-Of-Freedom Model. SAE International, Iowa City, IA, June 14-16, 2005-01-2727 DHM, 2005.

6. Peña-Pitarch,E., Yang,J., and Abdel-Malek,K. $SANTOS^{TM}$ Hand: Workspace Analysis. 16th IASTED International Conference on Modelling and Simulation, Cancun, Mexico, May 18-20, 2005.

7. Yang, J., Peña Pitarch, E., Abdel-Malek, K., and Lindkvist, Lars. Analysis of A Multi-Fingered Hand Prosthesis. Proceedings of ASME

28th Biennial Mechanism and Robotics Conference, Salt Lake City, Utah, Sept. 28-Oct. 2, 2004.

8. Peña-Pitarch,E., Al Omar,A., Yang, J., and Abdel-Malek,K. Diseño de un mecanismo para simulación del movimiento de un dedo. 6ème Congrès de Mécanique. Mécanique des Solides. TOME I. SMSM. SOCIET MAROCAINE DES SCIENCES MCANIQUES.TANGER. MAROC., p. 276-276, 2003.

9. Peña Pitarch, E., Yang, J., Abdel-Malek, K., and Saavedra Vilchez, D. Diseño y análisis de una prótesis de mano. Actas del XV Congreso Nacional de Ingeniera Mecánica. Asociación Española de Ingeniería Mecánica. Centro Tecnológico de Cádiz, p. 1-6, 2002.

10. J. Yang, K. Abdel-Malek, D. Degraff, and E. Peña Pitarch. DETC2002/MECH-34297: Design and Analysis of an Inherently-Compliant Light-Weight Active Hand Prosthesis. The Technical Program for the 2002 DETC/CIE Conference. ASME International (DETC/CIE), 2002.