Use Authorization

In presenting this thesis in partial fulfillment of the requirements for an advanced degree at Idaho State University, I agree that the Library shall make it freely available for inspection. I further state that permission to download and/or print my thesis for scholarly purposes may be granted by the Dean of the Graduate School, Dean of my academic division, or by the University Librarian. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Signature______________________________

Date__________________________________

DESIGN OF A FIVE-FINGERED UNDERACTUATED HAND FOR TWO-POSITION TASKS

By

Taher Deemyad

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science in Mechanical Engineering

College of Engineering, Idaho State University

May, 2016

COMMITTEE APPROVAL

To the Graduate Faculty:

The members of the committee appointed to examine the thesis of TAHER DEEMYAD find it satisfactory and recommend that it be accepted.

____________________________________**Major Advisor**

Dr. Alba Perez Garcia

____________________________________ **Member**

Dr. Brian Williams

____________________________________ **Graduate Faculty Representative**

Dr. Hossein Mousavinezhad

I dedicate this thesis to:

My father…

Creative man of my life who inspired love of science in me

and

My Mother…

Bright lady of my life who taught me to never give up

Acknowledgments

I am indebted to all those people who made my experience as a graduate student one that I will cherish forever.

I would like to sincerely thank …

My wonderful supervisor, Dr. Alba Perez, who with continuous support and guidance showed me the beauty of research and I am proud of being her student,

Dr. Brian Williams, Dr. Ken Bosworth, Dr. Mary Hofle, Dr. Bennett Palmer for teaching me advanced topics of mechanical engineering

My supervisory committee: Dr. Alba Perez, Dr. Brian Williams, and Dr. Hossein Mousavinezhad for all their time, encouragements and insightful comments.

Dr. Soheil Porkhial, Dr. Mohammad Layeghi, and Dr. Shahrokh Sorkhkhah who introduced me to engineering and Eng. Shahram Fotoohi for providing the internship opportunity and supporting me to continue my higher education.

Finally I have my love and gratitude for my family and friends who are always there for me and for their support in making this long journey happen;

With special thanks to
My amazing parents Dr. Mohammadreza Deemyad and Effat Anvarinejad
My lovely sisters Dr. Tara Deemyad and Dr. Shanti Deemyad
My awesome brother and my best friend Dr. Taha Deemyad,
My fabulous brothers in law Dr. Soroush Sadeghi and Dr. Saveez Saffarian
My dear uncle Col. Anavarinejad
My Friends and family here in US and back in my home country with whom I always feel home

And . . .

My home of Mechanical Engineering . . .

Idaho State University

• Contents

List of Figures VIII

List of Tables X

Abstract XI

1 Introduction 1

1.1 Thesis Goals 1

1.2 Literature Review 1

1.3 Organization of the Thesis 2

2 Basic Kinematic Definitions 3

2.1 Type of joints 3

2.2 Kinematic sketch methods 5

2.3 Basic Definitions 6

2.4 Hand Topology 7

2.5 Task 7

2.6 Solvability 8

2.7 Forward Kinematics 9

2.8 Kinematic Synthesis 9

3 Kinematics Background 10

3.1 Finite Displacement 10

3.1.1 Rigid Motion 10

3.1.2 Pure Translation 10

3.1.3 Pure Rotation 11

3.1.4 General Motion 12

3.2 Composition of Displacements 14

3.3 Inverse Displacement 15

3.4 Relative Displacement 16

4 Mobility and Reachable Positions 18

4.1 Number of Reachable Positions 18
4.1.1 Serial Robot 18
4.1.2 Tree Topology 19
4.2 Mobility Equations 20
4.3 Various Mobility Cases for Planar Robots 21
4.3.1 Reduced Mobility or Constrained Robot 21
4.3.2 Full Mobility Robot 21
4.3.3 Redundant Robot 21
4.3.4 Over Constrained Robot 21
4.4 Closed Chains 22
4.4.1 Planar Case 23
4.4.2 Bennet Linkage 24
5 Robotic Hands 25
5.1 Human Hand 25
5.2 Problems with designing a robotic hand similar to human hand 26
5.3 Various Robotic Hands 27
5.3.1 Commercial Robotic Hands 27
5.3.2 Robotic Hands with External Actuators 28
5.3.3 Simple Robotic Hand Design 28
6 Design of a Simple Wristed, Five-Fingered Robotic Hand 29
6.1 Introduction 29
6.2 Hand Topology and Solvability 29
6.3 Task 30
6.4 Methodology of Finding the Positions and Orientations of Joints 32
6.5 Constrain the Motion 33
6.5.1 Pulley and Belt System in Plane 33
6.5.2 Result of in Plane Pulley Belt System 34
6.5.3 Pulley and Belt System in Space 37
6.5.4 Linkage System in Space 39
7 Conclusions 49
8 References 50

List of Figures

Figure 1 : *Analyze the degree of freedom for six most popular joints* [17] *3*

Figure 2 *5*

Figure 3 *5*

Figure 4 *6*

Figure 5 : The left Topology shows a hand with 8 End-Effectors and the right Topology shows a hand with 7 End-Effectors *7*

Figure 6 : These figures show two positions of the End-Effectors for a five fingered-hand in space *8*

Figure 7 : *Pure Translation* *11*

Figure 8 : *Pure Rotation Around X Axis* *12*

Figure 9 : *General Motion* *13*

Figure 10 : *Composition of Displacement* *14*

Figure 11 : *Inverse Displacement* *16*

Figure 12 : *Relative Displacement* *17*

Figure 13 : Higher Pairs *20*

Figure 14 : *A) RR Serial Robot B) RRRR Closed Chain Robot* *22*

Figure 15 : *RRRR close chain parallel Robot* *23*

Figure 16: *Bennet Linkage* *24*

Figure 17 : *Joints and Bones in human hand* *25*

Figure 18 : *Barret Hand* *27*

Figure 19 : *Scott-Hand* *28*

Figure 20 : *Motion Capture System* *31*

Figure 21 : *Two Finger signs as a task for robot* *32*

Figure 22 : *Pulley System* *33*

Figure 23 *34*

Figure 24 *35*

Figure 25 *37*

Figure 26 *38*

Figure 27 *38*

Figure 28 *39*

Figure 29 : *Five finger robotic hand with Bennet Linkage* *46*

Figure 30 : *SolidWorks Model of a five finger hand with one revolute joint for each finger and one common revolute joint in the wrist in the First position* *47*

List of Tables

Table 1 : *Family of wristed-robotic hand with one joint for each finger* ____________________ *30*

Table 2 : *Initial and final position of fingertips (End-Effectors) in dual quaternion form*_______ *31*

Table 3 : *Dual quaternion form of all four joints of Bennet Linkage for all fingers for the First solution* ____ *44*

Table 4 : *Dual quaternion form of all four joints of Bennet Linkage for all fingers for the Second solution* ____ *45*

Abstract

Here instead of talking about the signs (that is just the application), I would talk more in general of two arbitrary positions.

The design of a five-fingered wristed robotic hand is discussed. The robot is capable of showing two signs with fingers using a kinematic synthesis technique and one revolute joint per finger.

The resulting candidate designs have two specified positions, first and second sign. This work deals with the implementation of these designs with minimal actuation. A coupled transmission system is designed in which all fingers are actuated by a single motor while each finger has its own separate reduction.

The reduction ratios are modified in order to adapt to different finger signs. The difference in behavior between this implementation and an underactuated, compliant design is studied.

Two different transmission systems; pulley-belt system and linkages system are compared in both planar and spatial case.

To define the tasks, we used Motion Capture System to record data for two sets of positions of fingertips in space. After determining the Hand-Topology, and Task, we used Kinematic Synthesis method, to find all the design parameters of the robot. These include length of links, positions of joints, and angle between links. Finally, we sketched a 3D model of our robot with SolidWorks and made a real model using prototyping. If you want to explain the methodology in this paragraph, then you should talk about designing serial chains to reach the positiosn in a first step, and then designing the constrain system in a second step to obtain a 1-dof system for each finger.

1 Introduction

1.1 Thesis Goals

The goal of this thesis is design of a robotic hand which performs simple tasks such as showing two different sings or pick and place an object with minimum degrees of freedom (DOF).

To reach this aim, we choose a five-fingered robotic hand with one revolute joint for each finger and one revolute joint in the wrist. Methods of coupling between wrist joint and fingers joints, Pulley-Belt System and Linkage System in planar and spatial cases were analyzed.

Finally, using SolidWorks software, the example was modeled a graphical model of a wristed-hand with five finger which uses the Bennet Linkage for connection between wrist and fingers.

1.2 Literature Review

Recent developments on the kinematic synthesis of tree topologies [4], [5] for a simultaneous task of all end-effectors, can be used for designing the robotic hands. In particular, if the task is given by two positions of each fingertip, it is possible to design a hand to show two signs with fingers or a simple pick-and-place task.

As one of the first steps, a hand topology needs to be selected and checked for solvability.

A complete chart of all solvable multi-fingered hands with identical fingers is presented by A. Makhal and A. Perez-Gracia [6].

Table [1] shows the special case of the family of one jointed hands. It can be seen that the family of one-jointed hands are solvable for several topologies. The kinematic synthesis of this family is studied by A. Perez Gracia [7]. In their study a simplified synthesis algorithm is presented and examples are shown for hands with one revolute joint at the wrist and multiple fingers. Each finger is attached to the palm with one revolute joint. In the present study we select a hand from five finger family, which is solvable for two positions for each finger. The hand that is synthesized in the present manuscript is designed for showing two different sings with fingers as a task.

In this project there are chose two finger signs which were completely different, so it may not require full actuation of all fingers, but rather a system to switch from first sign to second sign. In order to accomplish this, a single actuator may be used to couple the motion of all fingers through an appropriate transmission system. [10], [11], [12]

In this work, synthesis algorithms [19] is used for designing this five-fingered robotic hand. Also two different reduction systems, pulley- belt [21] and Linkages system [8], [10], [20] will be analyzed in plane and space. Finally, Bennet linkages [15], [16] will be used for designing this robotic hand.

1.3 Organization of the Thesis

The thesis is organized as follows. Chapter 1 discusses the theme of the thesis and the literature review. Chapter 2 provides some basic definitions of the kinematics. In Chapter 3 discusses the background of motion study. Chapter 4 focuses on mobility and number of reachable positions for a robotic hand. In Chapter 5, various types of robotic hands are explained. Chapter 6 focuses on process of designing a robotic hand, which followed with an example. Chapter 7 provides conclusions and future works.

2 Basic Kinematic Definitions

2.1 Type of joints

There are several different types of joints in kinematics [13]. Revolute Joint (R), Prismatic Joint(P) Cylindrical Joint, Universal Joint, Spherical Joint, Helical Joint, Plane Pair, Gear Pair, Cam Pair Figure [1] shows some of the popular types of them.

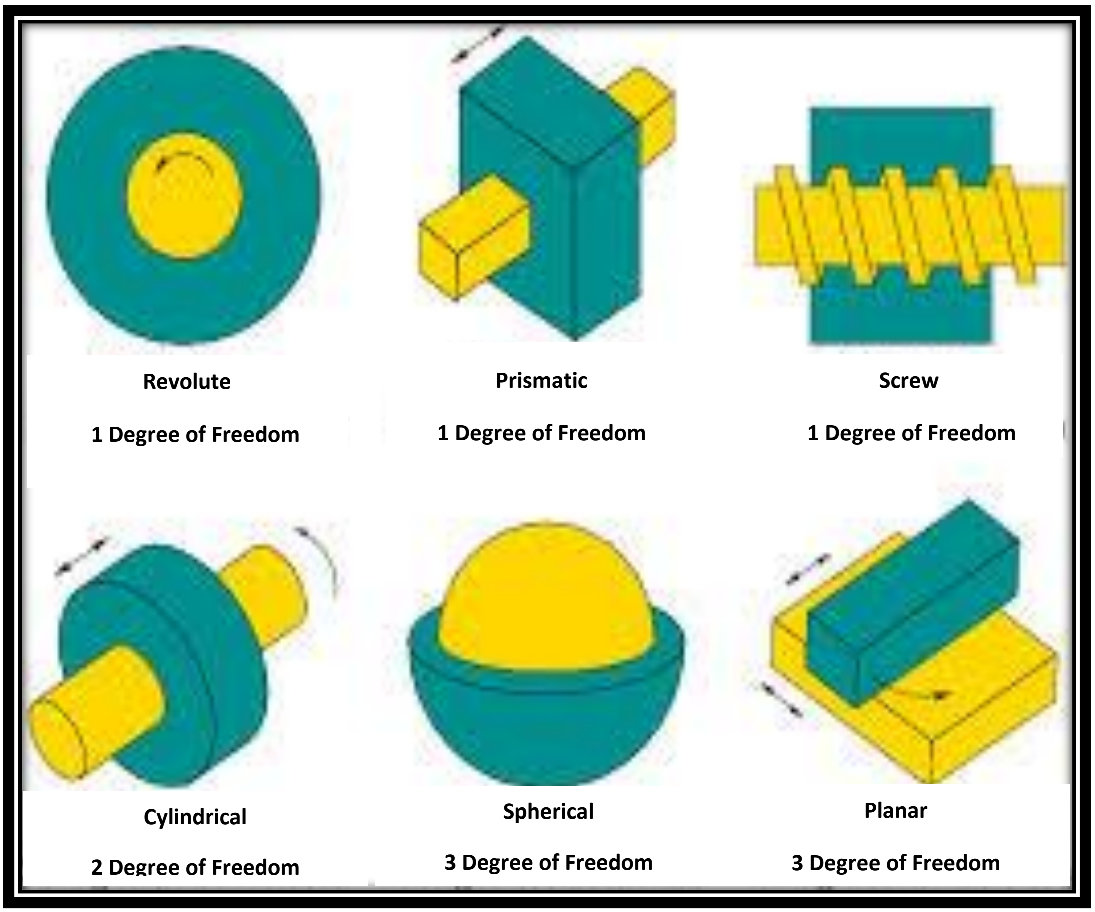


Figure 1 : *Analyze the degree of freedom for six most popular joints [17]*

Revolute Joint: As shown in Figure [1] this joint allows rotation (of the links) about an axis, therefore it has only one degree of freedom.

Prismatic Joint: This joint allows one translation along a single direction (Figure [1]), therefore similar to the revolute joint it has only one degree of freedom.

Screw Joint: In this joint, when the joint allows link rotates around an axis, that link has a translation along one direction (Figure [1]). So, there is a relationship between rotation and translation therefore similar to the two previous joints it has only one degree of freedom.

Cylindrical Joint: Like screw joint, this joint allows a rotation around an axis and a translation along a direction. However, unlike screw joint, in a cylindrical joint the rotation and translation are independent (Figure [1]). As a result there are two degrees of freedom.

Spherical Joint: In a spherical joint, links are free to rotate around all three axes X,Y,Z. In this type of joint there are 3D rotations and it has three degrees of freedom (Figure [1]).

Planar Joint: A planar joint allows a rotation around an axis perpendicular to a plane and translation along two directions in the plane (Figure [1]). Therefore, it has three degrees of freedom.

2.2 Kinematic sketch methods

Kinematic mechanisms can be shown in two different ways:

Kinematic diagram or kinematic scheme: In this method only the connectivity between links and joints of a mechanism or machine is shown and the sketch is very simple (Figure [2]).

Graph: In this method the links are shown with points and joints with lines(which appears to be in the opposite order of real shape). Also, in the graph method a point surrounded with a circle is used as a symbol for fixed link (Figure [2], [3]). To show the links are connected to each other with which type of joints, usually type of joint are wrote on top of its corresponding line.

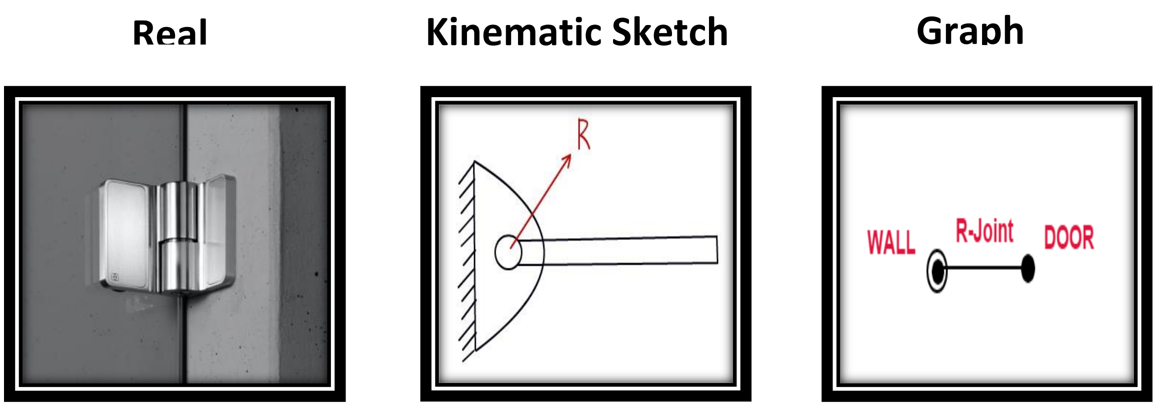


Figure 2

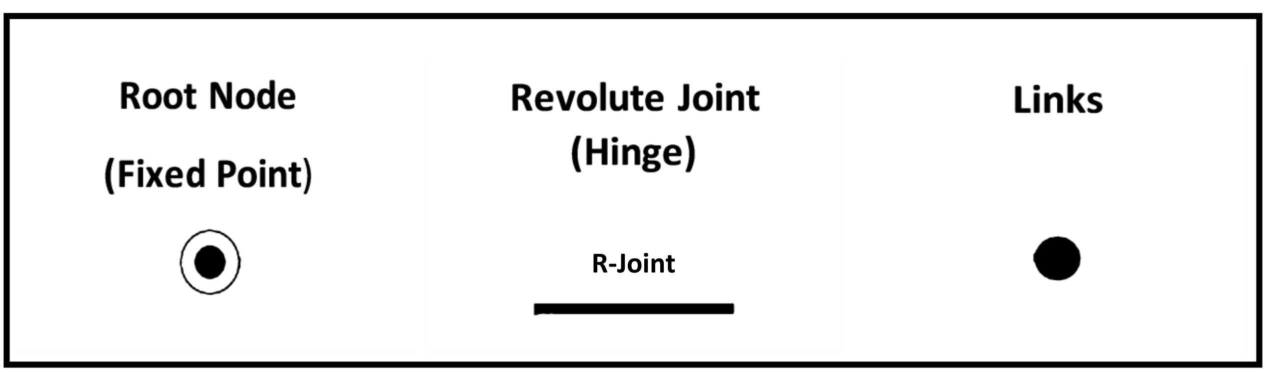


Figure 3

2.3 Basic Definitions

A graph is shown in the Figure [4]. In this graph, it can be defined:

- Root Node (Base) : Fixed link (Ground) of the robot (⦿)
- End-Effector : Last points in the fingers of the robot (fingertips) which contact to object (Pink Points)
- Chain : Series of joints and links
- Branch: Serial chain starting at root node and ending in one end effector (there are three branches in this example)

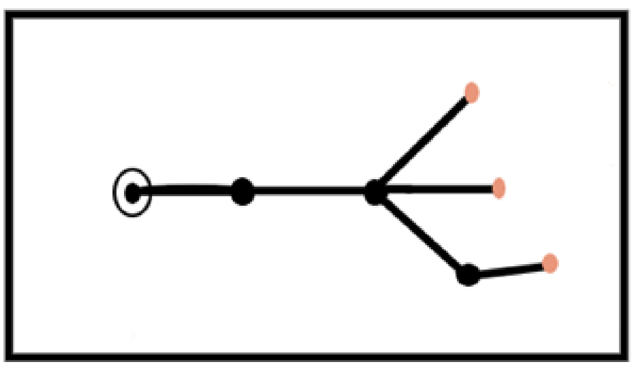


Figure 4

2.4 Hand Topology

The number of joints, type of joints, number of fingers and fingertips, number of branches and connectivity between points define the Hand Topology. Defining the Hand Topology is one of the basic steps for designing any robotic hand. Two different hand topologies are shown in Figure [5] (different number of joints, number of branches, etc.).

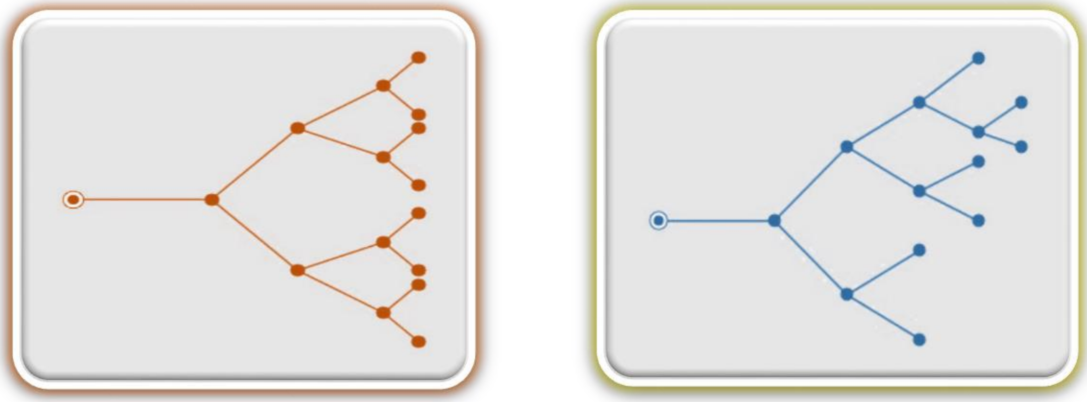


Figure 5 : The left Topology shows a hand with 8 End-Effectors and the right Topology shows a hand with 7 End-Effectors

2.5 Task

When we define a task for our robot, we need to know the positions and orientations of the fingertips for each End-Effector in space. For example as shown in Figure [6], if the robot is designed to get to two different positions, there must be two sets of information for each end-effector of the robot in the space (position and orientation).

First Position **Second Position**

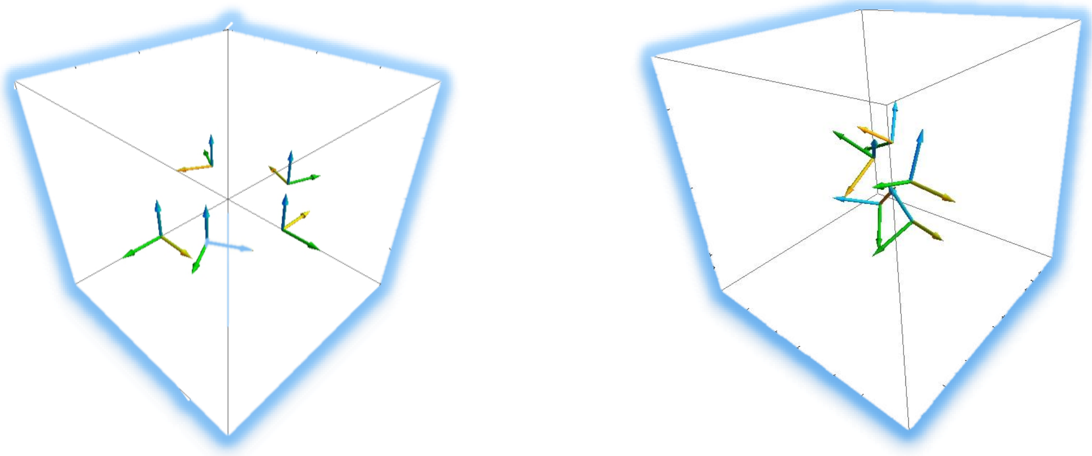


Figure 6 : These figures show two positions of the End-Effectors for a five fingered-hand in space

2.6 Solvability

Define solvable as having a finite number of positions ≥ 2 for the hand without over-constraining any finger.

Let 'm' be the number of positions for the overall tree. Then $m_i \geq m$, when m_i being the number of positions for each sub-graph of the tree graph, starting at the root rode or an end-effector and ending at one or more end-effectors.

2.7 Forward Kinematics

Forward Kinematics refers to the use of the kinematic equations of a robot to compute the position of the end-effector from specified values for the joint parameters [18]. In absolute forward kinematic, the position of the joints (or length of the links and angle between them) are given and with using the Homogeneous Transform Matrix and matrix multiplication (It will be explained in the next chapter) the position of the end-effector will be found. However, usually the relative forward kinematic are utilized rather than absolute forward kinematic. For relative forward kinematic the multiplication of relative rotation from reference configuration of the joints will be equal to relative displacement of end-effector. With solving the equations, the positions of the end-effector will be found.

Relative Forward Kinematic:

$$[S_1(\Delta\theta_1^i)] \dots [S_n(\Delta\theta_n^i)] = [D_{0i}] \qquad \textbf{\textit{Equation (1)}}$$

$$\Delta\theta_n^i = \theta_n^i - \theta_n^0$$

2.8 Kinematic Synthesis

The synthesis is a method which use for designing robotic hands. In the synthesis also we can use absolute or relative displacement. Opposite to Forward Kinematic, in this method the hand topology of our robot and the task (positions of the end-effectors) are given and this time the joint parameters are unknown. So first the solvability must be checked. Then for the Relative Synthesis same as Relative Forward Kinematic, we need to find the relative displacements of end-effectors and take them equal to relative displacement of joints. However, this time the positions of the end-effector are known. With solving these equations the length of each link, position of each joint and all other design parameters that are needed for design of the robot will be found.

In this chapter, some of the basic kinematic definitions such as Solvability, Forward Kinematics, and Kinematic Synthesis were reviewed which will be used in the fallowing chapters. The next chapter is about various types of displacement and motion.

3 Kinematics Background

3.1 Finite Displacement

3.1.1 Rigid Motion

Any way of moving all the points in the plane such that the distance between points and angle between lines remains unchanged is named Rigid Motion. Totally there are four types of Rigid motions, translation, rotation, reflection, and glide reflection. In the next sub-sections the translation and rotation will be explained.

3.1.2 Pure Translation

All points move by the same amount (Figure [7]). A translation is expressed as a 3D vector:

$$t = \begin{Bmatrix} t_x \\ t_y \\ t_z \end{Bmatrix}$$ ***Equation (2)***

Composition of translation is vector addition.

$$P = p + t$$ ***Equation (3)***

($\boldsymbol{p} \longrightarrow$ Initial position, $\boldsymbol{P} \longrightarrow$ Final position)

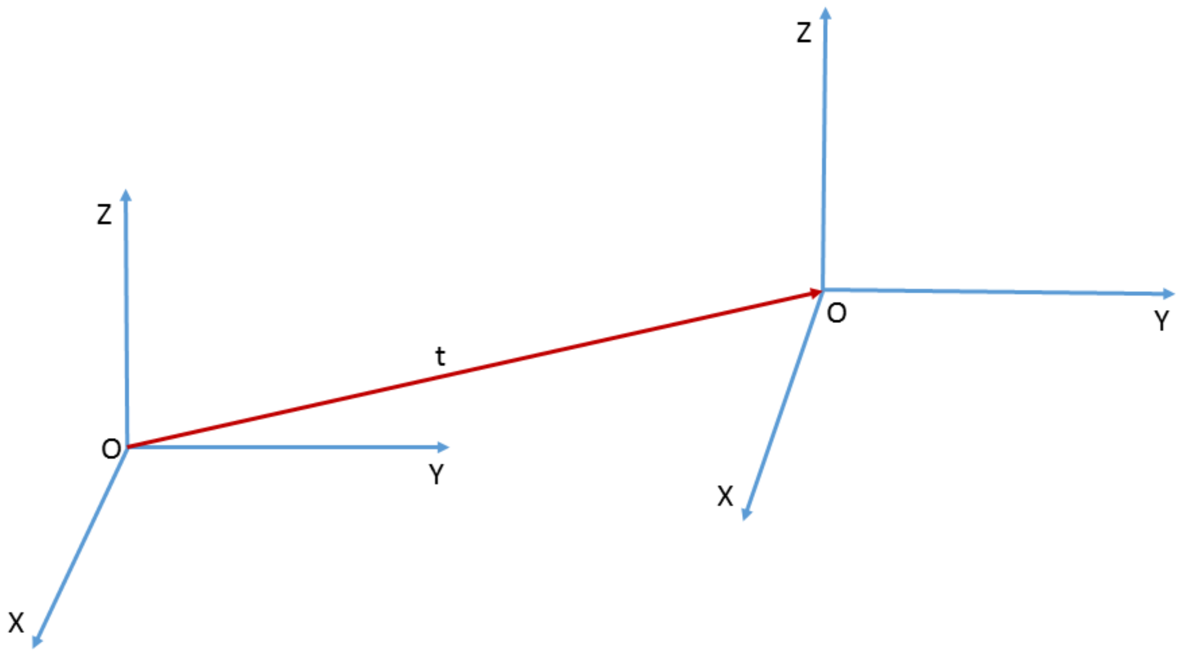


Figure 7 : Pure Translation

3.1.3 Pure Rotation

Origin remains fixed, all other points move proportional to the distance to the fixed axis (Figure [8]). A rotation is represented as an orthogonal 3D matrix.

$$[R_x] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & Cos\theta & -Sin\theta \\ 0 & Sin\theta & Cos\theta \end{bmatrix}$$

$$[R_y] = \begin{bmatrix} Cos\theta & 0 & Sin\theta \\ 0 & 1 & 0 \\ -Sin\theta & 0 & Cos\theta \end{bmatrix}$$

Equation (4)

$$[R_z] = \begin{bmatrix} Cos\theta & -Sin\theta & 0 \\ Sin\theta & Cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

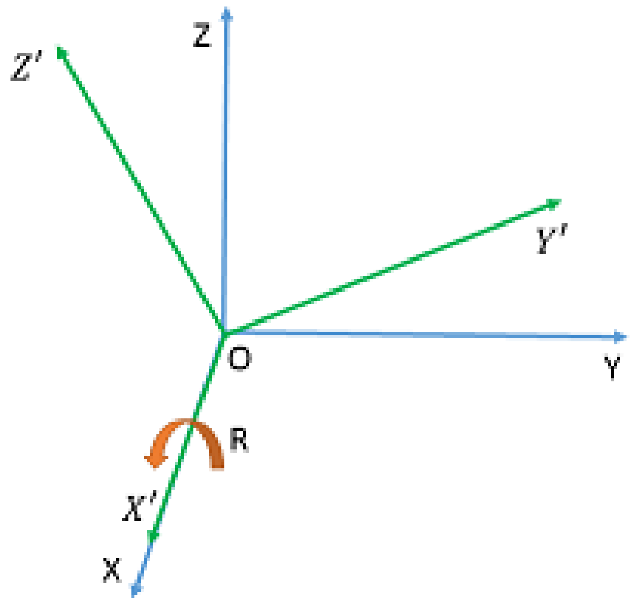


Figure 8 : Pure Rotation Around X Axis

When a point transformed by a pure rotation, new position will be found by matrix multiplication (Equation [5]).

$$P = [R] \times p$$ ***Equation (5)***

($\boldsymbol{p} \longrightarrow$ Initial position, $\boldsymbol{P} \longrightarrow$ Final position)

3.1.4 **General Motion**

The General Motion is a combination of Rotation and Translation (Figure [9]). It is equal to composition of Rotation plus Translation

$$P = [R] \times p + t$$ ***Equation (6)***

We can write rotation and translation as a special 4D matrix [T]. This matrix is called a Homogeneous Transform Matrix.

$$[T] = \left[\begin{array}{ccc|c} & [R] & & t \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Equation (7)

The composition of Homogeneous Transform Matrix is the matrix multiplication (Action is matrix-vector product)

Because we embedded our 3D space in a 4D space, we need to define points and directions as a 4 parameters vector:

$$\text{Points} : P = \begin{Bmatrix} p \\ 1 \end{Bmatrix} \ , \quad \text{Directions} : V = \begin{Bmatrix} v \\ 0 \end{Bmatrix}$$

Equation (8)

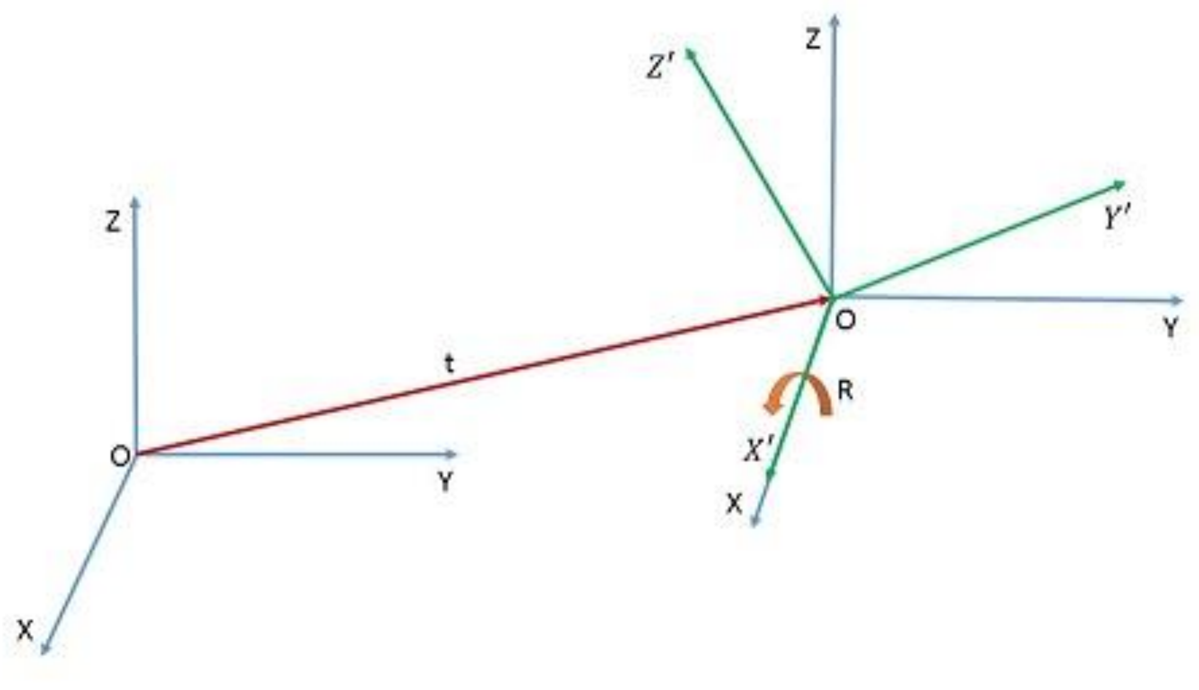


Figure 9 : *General Motion*

3.2 Composition of Displacements

In this case we just need to find a new Homogeneous Transform Matrix which is a combination of all displacements produced by Homogeneous Transform Matrices. That will be a simple matrix multiplication (Figure [10])

$$[\boldsymbol{T3}] = [\boldsymbol{T1}][\boldsymbol{T2}] = \left[\begin{array}{ccc|c} & [\boldsymbol{R1}] & & \boldsymbol{t1} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right]\left[\begin{array}{ccc|c} & [\boldsymbol{R2}] & & \boldsymbol{t2} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right] = \left[\begin{array}{ccc|c} & [\boldsymbol{R1}][\boldsymbol{R2}] & & [\boldsymbol{R1}]\boldsymbol{t2} + \boldsymbol{t1} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right] \qquad \textit{Equation (9)}$$

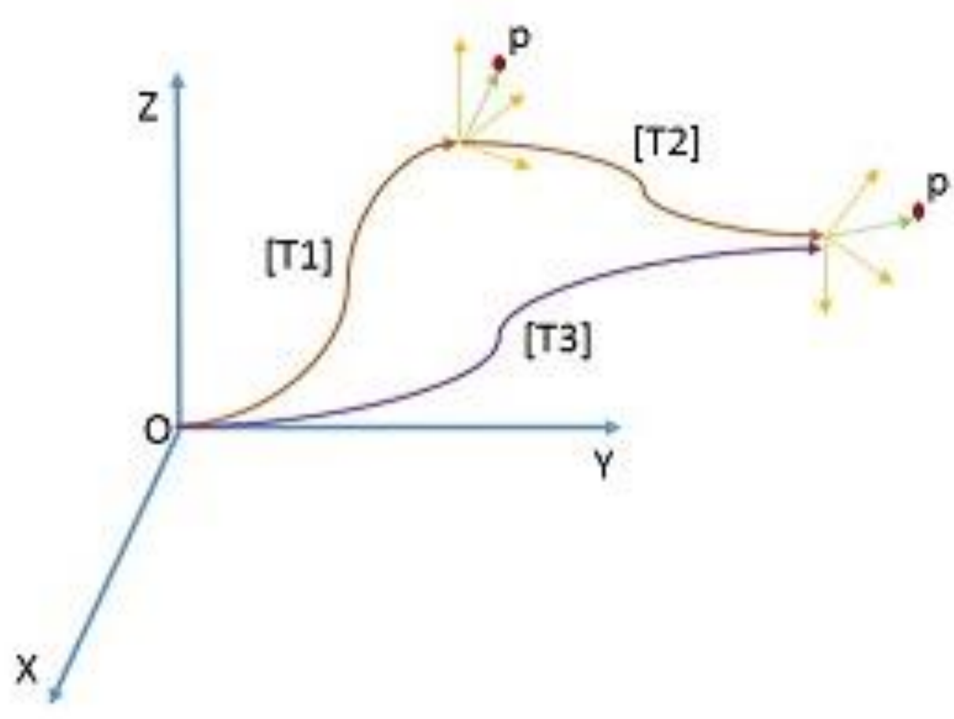


Figure 10 : *Composition of Displacement*

3.3 Inverse Displacement

In this part we will find a relationship between original displacement and inverse displacement. As you can see in figure [11], transform matrix [T2] is inverse of transform matrix [T1]. Therefore, [T1] [T2] = [Id]. Also, because [R1] is an orthogonal matrix, so $[R1]^{-1} = [R1]^T$

$$\left[\begin{array}{ccc|c} & [\boldsymbol{R1}] & & \boldsymbol{t1} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right]\left[\begin{array}{ccc|c} & [\boldsymbol{R2}] & & \boldsymbol{t2} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right] = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}$$

$$\left[\begin{array}{ccc|c} & [\boldsymbol{R1}][\boldsymbol{R2}] & & [\boldsymbol{R1}]\boldsymbol{t2} + \boldsymbol{t1} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right] = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}$$ ***Equation (10)***

$[R1][R2] = [Id] \Rightarrow [R2] = [R1]^{-1} = [R1]^T$

$[R1]t2 + t1 = 0 \Rightarrow t2 = -[R1]^T t1$

So : $$[\boldsymbol{T1}]^{-1} = \left[\begin{array}{ccc|c} & [\boldsymbol{R1}]^{\boldsymbol{T}} & & -[\boldsymbol{R1}]^{\boldsymbol{T}}\boldsymbol{t1} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{array}\right]$$ ***Equation (11)***

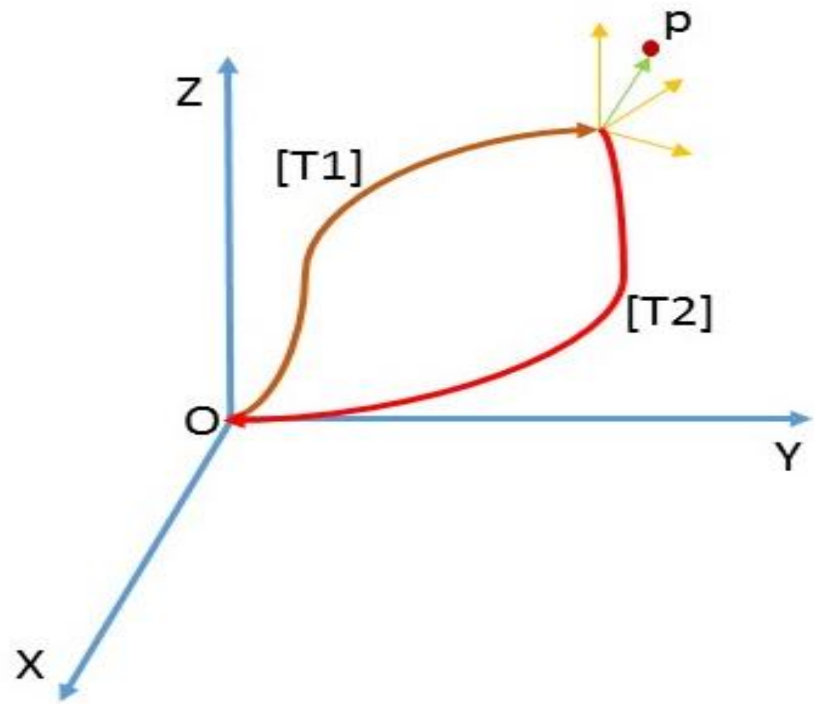


Figure 11 : *Inverse Displacement*

3.4 Relative Displacement

In kinematic synthesis relative motion is used rather than general motion. As shown in Figure [12], the relative displacement is a displacement between two positions coordinates in the fixed frame. The relative displacement is found by equation [12].

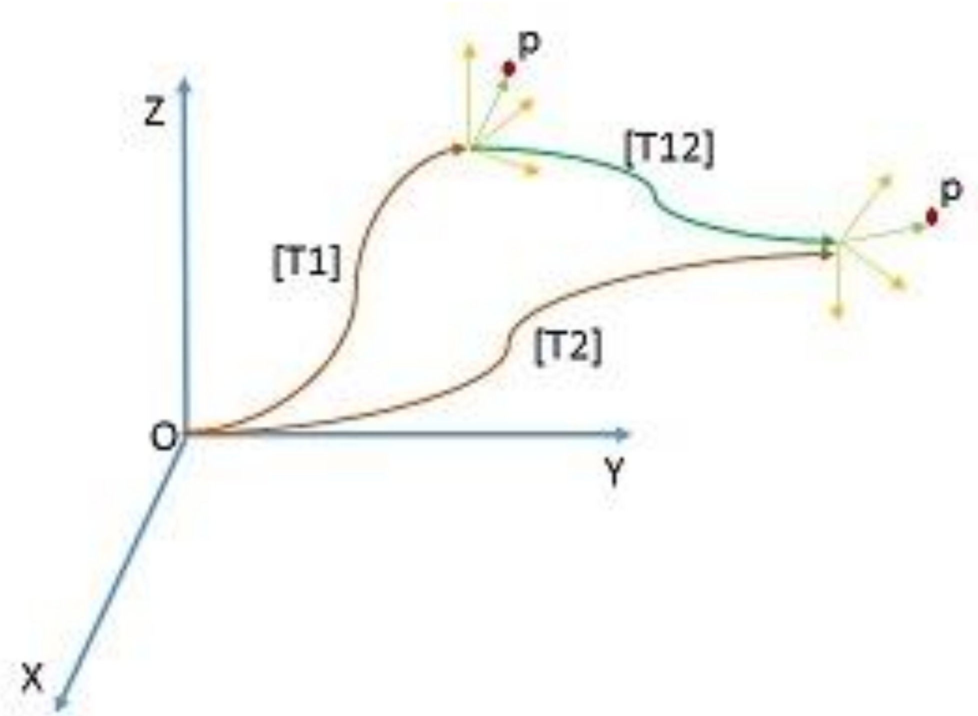


Figure 12 : *Relative Displacement*

$$[T_{12}] = [T_2][T_1]^{-1} \Longrightarrow \quad Where\ [T_{12}]\ is\ reletive\ displacement \qquad \textbf{\textit{Equation (12)}}$$

In this chapter, various types of displacement and motion such as Finite Displacement, Inverse Displacement, and Relative Displacement were explained which will be used in the final design. In the next chapter, we will talk about Mobility and Reachable Positions.

4 Mobility and Reachable Positions

4.1 Number of Reachable Positions

4.1.1 Serial Robot

As shown in part (2-4), one of the methods for design of a robotic hand is kinematic synthesis. In this method we assume that we have the positions of end-effectors, then equate Forward Kinematics (FK) for task displacements and solve the equations for all unknowns of FK. This also works with relative FKs

$[FK_{1i}] = [P_{1i}]$, $i = 2, \dots, m$ (where m is the number of positions the robot can reach defined by its topology)

For each relative position using matrix algebra, we can write 12 equations. 6 of these equations are independent with 3 translational equations and 3 rotational equations.

Using equation [13] below. The number of positions reachable by a serial R-Robot is calculated:

$$6(m-1) = 4n + n(m-1) \qquad \textbf{\textit{Equation (13)}}$$

$n = number\ of\ joints, m = number\ of\ positions$

Therefore,

For a RR robot : m=3 & we have 36 equations (18 independent equations)

For a RRR robot : m=5 & we have 60 equations (30 independent equations)

For a RRRR robot : m=9 & we have 108 equations (54 independent equations)

For a RRRRR robot : m=21 & we have 252 equations (126 independent equations)

And so on. Therefore, the number of equations that we need to solve, with adding each joint to the chain will increase rapidly.

Also, the number of reachable positions of the robot in the plane is calculated from equation [14]

$$m = \frac{r+3}{3-r-t}$$ ***Equation (14)***

$r = number\ of\ revolute\ joints, t = number\ prismatic\ joints$

For any $r + t \geq 3$ the robot can reach to any position in plane

4.1.2 Tree Topology

To find the maximum number of reachable positions for a tree topology, first we need to find the number of subgroups of that tree and then find the number of positions for each subgroup with equation [15]. The maximum number of reachable positions will be equal to minimum number of positions of these subgroups. For finding all subgroups, we need to eliminate edges one by one from the tree topology and make new subgroups.

$$m = \frac{D_s^e.E - D_c^n.B}{D_{ee}^n.B - D_j^e.E} + 1$$ ***Equation (15)***

Where,

D_s^e : Vector containing number of structural variables for each edge

E : Vector of 1's for edges

D_c^n : Vector of extra constraints for each branch

B : Vector of 1's for branches

D_{ee}^n : Vector of Degree of freedom for End-Effector

D_j^e : Vector wit number of joint variables per edge

4.2 Mobility Equations

We can find the degree of freedom (Mobility) of mechanisms with below equations:

A) Mobility equation in the **plane** [14] :

$$M = 3(n-1) - 2j_p - j_h \qquad \textit{Equation (16)}$$

$n =$ Total number of links in the mechanism

$j_p =$ Total number of lower pair$s$ $in\ the\ plane$ $*$ (R − joint, P − joint)

$j_h =$ Total number of higher pairs $**$ (Gear pair, Cam pair)

*** Lower pairs:**

In Plane : Revolute Joint (R),Prismatic Joint(P)

In Space : Cylindrical Joint, Universal Joint, Spherical Joint, Helical Joint, Plane Pair

**** Higher pairs:**

Gear Pair :

Cam Pair :

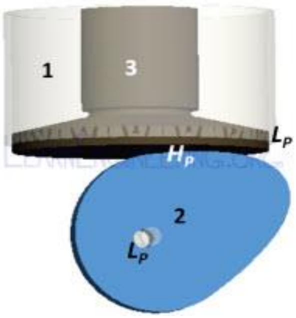


Figure 13 : Higher Pairs

B) The General Equation For degree of freedom (Mobility) :

$$M = D(n-1) - \sum_{i=1}^{j}(D - f_i) \qquad \textbf{\textit{Equation (17)}}$$

$D = dimention\ of\ subgroup\ of\ motion(3\ for\ plane, 6\ for\ space)$

$n = number\ of\ links$

$j = number\ of\ joints$

$f_i = Degree\ of\ freedom\ joint\ i$

4.3 Various Mobility Cases for Planar Robots

4.3.1 Reduced Mobility or Constrained Robot

When the DOF of the robot is less than the DOF of space, the robot can move but there are some barrier for designing. In this case, the robot mainly can perform a task with less DOF.

4.3.2 Full Mobility Robot

When the DOF of robot is equal to DOF of space, robot can reach all the positions.

4.3.3 Redundant Robot

When the DOF of robot is more than DOF of space, even with some constrains, robot can reach all the positions. Actually, this allows the designer to more configurations for a single end-effector position, avoiding obstacles, etc.

4.3.4 Over Constrained Robot

When the mobility of a robot is less than zero (M <0), that means this robot can't move (locked). Therefore, we need to eliminate some constraints or change the work space (for example maybe a robot be over constrained in space when it can move in plane).

4.4 Closed Chains

Closed Chain or Hybrid Chain is when there is more than one path along the chain to reach the end-effector. In Figure [14] a 2R serial Robot and a 4R Closed Chain Robot are shown:

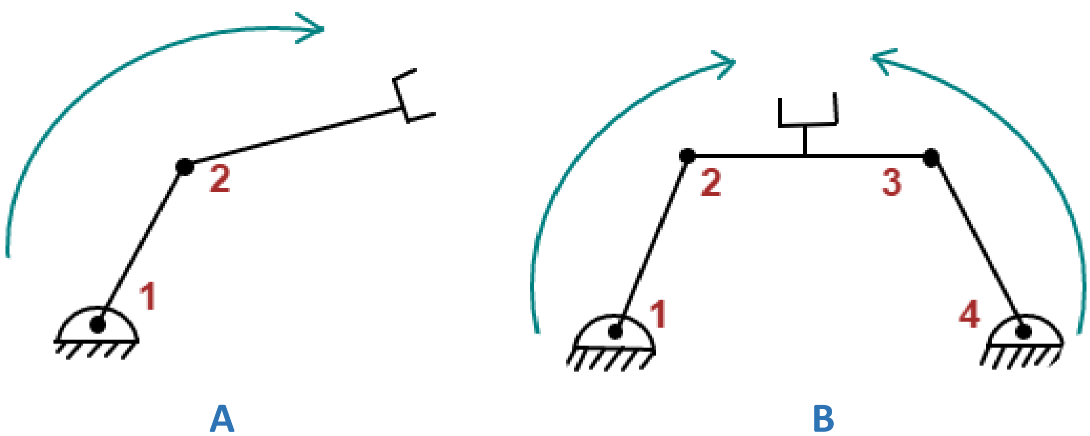


Figure 14 : *A) RR Serial Robot B) RRRR Closed Chain Robot*

Now if we calculate the mobility of these two mechanisms in space, we show that the 4R closed chain is an Over Constrained linkages and just movable for some special geometries.

A) Serial 2R $M_6 = 6(3-1) - 2(6-1) = 2$

B) Closed 4R $M_6 = 6(4-1) - 4(6-1) = -2$

In the next two sections, two special geometries for a 4R Closed Chain Robot which is movable are explained.

4.4.1 Planar Case

As shown in part 4-3-1, a Constrained Robot can change to a movable robot with a decrease in the DOF of work space. Here we will make our robot with four parallel revolute joints (Figure [15]). When all four revolute joints in a RRRR closed chain are parallel to each other, we can consider it as a planar case. So mobility in this special case becomes equal to one. Therefore, it will be solvable.

Mobility of Planar Closed 4R Robot: $M_3 = 3(4-1) - 4(3-1) = 1$

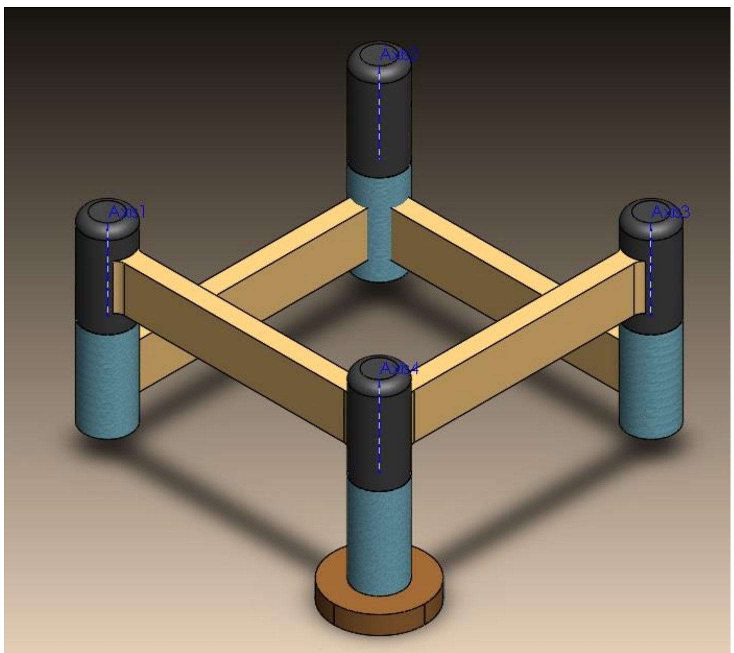


Figure 15 : *RRRR close chain parallel Robot*

4.4.2 Bennet Linkage

Another famous movable special geometry for a RRRR closed chain robot is called Bennet Linkage. For Bennet Linkages [15], [16], we need to have some relations between angle and distance of revolute joints. According to parameters in the Figure [16], for bennet linkage we need to have the following two equations.

1. $$\frac{a}{Sin\ \alpha} = \frac{g}{Sin\ \gamma}$$ ***Equation (18)***

2. $$G\,.\,H\ = W\,.\,U$$ ***Equation (19)***

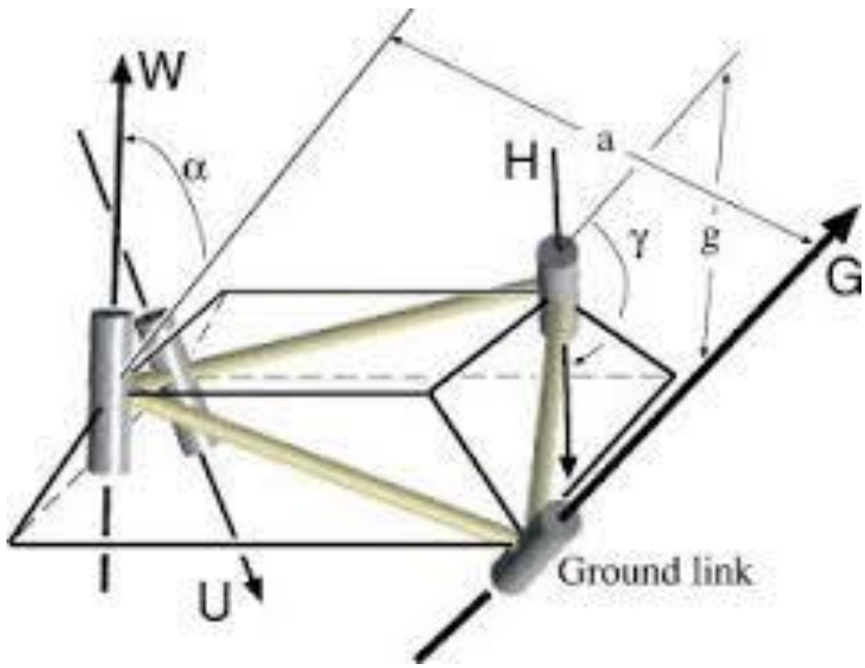


Figure 16: *Bennet Linkage*

In this chapter, equations for calculating the number of reachable positions and degree of freedom were explained. Also, various type of mobility cases were defined. Next chapter, will be talked more about different types of robotic hands.

5 Robotic Hands

5.1 Human Hand

In this part we analyze a real human hand. In Figure [17] is shown skeleton of a human hand.

- The human hand has five fingers and **27** bones, not including the sesamoid bone [1]. (29 including the sesamoid bone)
- The human hand has 27 degrees of freedom [2]
- Four DOF for each finger, three for extension and flexion and one for abduction and adduction
- The thumb is more complicated and has five degrees of freedom
- Leaving 6 DOF for the rotation and translation of the wrist

Figure 17 : *Joints and Bones in human hand*

5.2 Problems with designing a robotic hand similar to human hand

According to part 4-1, the addition of revolute joints to the robotic design result in a dramatic increase of the number of equations. Therefore, designing a robotic hand with same shape as human hand becomes challenging.

Also as shown in 5-1, a human hand has 27 DOF. Therefore to design a robotic hand with the same shape of the human hand, at least we need 27 actuators (motors), because we need one actuator for each DOF. A robotic hand with at least 27 actuators is also too heavy compared to a human hand (for male 0.65% & for female 0.5% of total body [3] weight which become about 0.5 Kg). Another problem with the large number of actuators is space limitation. The only space available for actuators is in the palm of the robotic hand which is a small space for 27 regular actuators. Also the price of small actuators is high so the final design with this actuators would be expensive.

5.3 Various Robotic Hands

5.3.1 Commercial Robotic Hands

A good example of a robotic hand with advanced technology is Barret Hand (Figure [18]). Barret hand is a 3-fingered robotic hand which is available on the market. This robot uses small actuators and advanced technology and is able to do combination of many different movements. However, its high price makes it disadvantaged for common use.

Figure 18 : *Barret Hand*

5.3.2 Robotic Hands with External Actuators

In this method, the user carries the actuators in a bag pack and they are connected to the joints with wire. However, because the user always needs to carry a bag pack, maybe that is not very comfortable in long time!

Figure 19 : Scott-Hand

5.3.3 Simple Robotic Hand Design

The final aim for designing a robotic system is making an artificial mechanism which is capable of performing a particular tasks. About the robotic hand, usually we want to make a mechanism which is able to perform specific tasks similar to the human hand. For instance, grasping something or showing a sign. As a result, the best design is a design where the robotic hand using any hand topology can perform the task with less complexity and fewer actuators. So, if we can design a robotic hand with only one DOF, it needs only one actuator and therefore would be advantageous to pervious methods (price, weight, space, etc.)

6 Design of a Simple Wristed, Five-Fingered Robotic Hand

6.1 Introduction

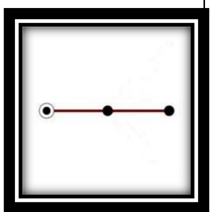
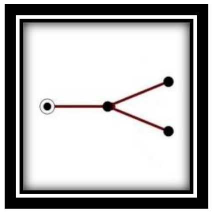
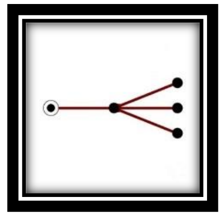
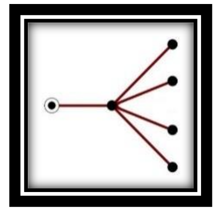
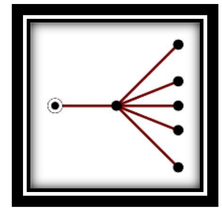
In this chapter we discuss designing a simple robotic hand with minimum DOF. We will use the basic knowledge from previous chapters and design the process step by step followed by an example for clarification. First of all, we need to select a hand topology for our robot and find solvability. Then, define a specific task for the hand. Finally, in this chapter, I will talk about two different methods which we use for connectivity between joints in the wrist and fingers.

6.2 Hand Topology and Solvability

As we saw in part 5.3.3, usually the simpler hand design with less number of joints and minimum DOF will be better design. By this we mean, if a hand topology with just one DOF can perform the task for us, it will by definition be one of the best designs.

Also, from Solvable multi-fingered hands for exact kinematic synthesis by A. Makhal and A. Perez-Gracia [6], it can be seen that the family of one-jointed hands is solvable for many topologies. We can find the number of positions solvable by the robot using equations [13], [14] and [15]. You can see the family of wristed-robotic hands with one joint for each finger in Table [1]. For more than five fingers, number of reachable positions for the robot becomes less than two, which means that the robot just reaches to one position and making it useless. In this family, the number of reachable positions of the last one, a five finger robotic hand, is sufficient for pick-and-place or showing two signs with fingers. Additionally, it is the most similar to a human hand with sufficient fingers for grasping objects properly or showing various types of common signs.

Table 1 : *Family of wristed-robotic hand with one joint for each finger*

Graph					
Number of Fingers	1-Finger	2-Finger	3-Finger	4-Finger	5-Finger
Hand Topology	$1-(1)$	$1-(1,1)$	$1-(1,1,1)$	$1-(1,1,1,1)$	$1-(1,1,1,1,1)$
Reachable Positions	m=3	m=2.33	m=2.14	m=2.05	m=2
Solvability	Solvable	Solvable	Solvable	Solvable	Solvable

6.3 Task

The final goal of this project is designing a robotic hand which can pick and place an object or show a sign with fingers. In this case we need to have two sets of data for positions of the end-effectors. We use Motion Capture System to record these data. The Motion Capture System is a set of high speed cameras which can record the positions and orientations of points in space. As you can see in the Figure [20], we stick a sensor to each end-effector and one sensor to somewhere on the forearm (fixed point) as the Rote Node which allows for comparison of position and orientation of each point to the Rote Node as Origin. We selected two different finger signs as a task for the robot as shown in Figure [21]. The data for positions of the end-effector for this task are given in the Table [2].

Figure 20 : *Motion Capture System*

Table 2 : *Initial and final position of fingertips (End-Effectors) in dual quaternion form*

Finger	**Initial Position**	**Final Position**
Finger 1	$-0.30 - 0.26i + 0.30j + 0.87k + \varepsilon(-3.81i - 3.38j + 3.90k - 3.70)$	$-0.08 + 0.05i - 0.14j + 0.98k + \varepsilon(-1.70i - 0.65j - 5.05k - 0.85)$
Finger 2	$-0.13 - 0.04i + 0.17j + 0.97k + \varepsilon(-6.14i - 1.71j + 7.99k - 2.33)$	$0.03 - 0.01i - 0.08j + 1.00k + \varepsilon(3.00i - 1.28j - 7.48k - 0.71)$
Finger 3	$-0.25 - 0.04i + 0.97k + \varepsilon(-2.32i - 0.39j - 0.63)$	$-0.04 - 0.01i + 0.11j + 0.99k + \varepsilon(-3.27i - 0.45j + 9.16k - 1.18)$
Finger 4	$-0.19 + 0.05i + 0.98k + \varepsilon(-2.35i + 0.64j + -0.48)$	$-0.09 - 0.02i + 0.24j + 0.97k + \varepsilon(-2.68i + 0.63j + 8.26k - 2.31)$
Finger 5	$-0.07 + 0.02i + 0.08j + 0.99k + \varepsilon(-5.86i + 1.34j + 6.91k - 1.03)$	$0.09 - 0.002i - 0.29j + 0.95k + \varepsilon(1.14i - 2.26j - 6.35k - 2.09)$

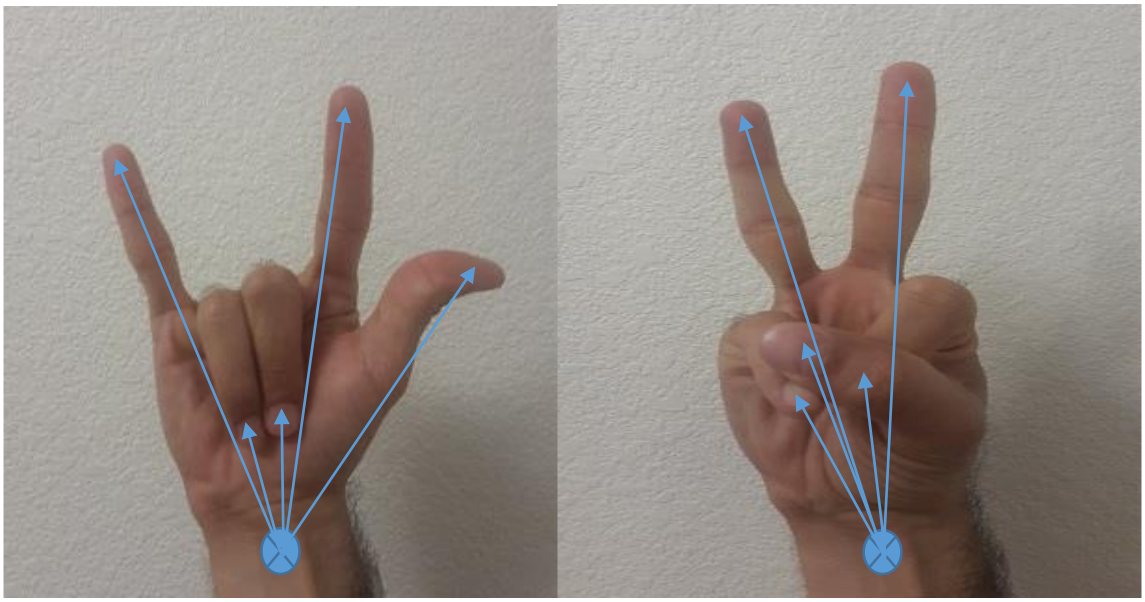


Figure 21 : Two Finger signs as a task for robot

6.4 Methodology of Finding the Positions and Orientations of Joints

In this section, there is a general information about the method we will use in the next section. Here we use synthesis methods to find the positions of the joints. In this method we need to know the task generated by the positions of end-effectors from motion capture system. Here we prefer to use the quaternion form of positions rather than matrix form, so using Mathematica transform the positions to dual quaternion form (Table [2]). After this step, we need to find the relative displacements, since each fingertip reach to only two positions, there will be just one relative displacement for each finger. Then relative displacements are set equal to relative Forward Kinematic and because the last component of the dual quaternion for each joint solution must be equal to zero, we will have 5 new set of equations. With eliminating the common part from equations ($cot\frac{\Delta\theta_0}{2}$), they result in four sets of equations (Equation [33]). In addition, we have two extra constraint equations for the wrist screw (Equation [34]). On the other hand, we have six unknowns for the wrist. With solving these equations at same time, we find the dual quaternion of joint of the wrist. Finally, with using the information from joint of the wrist and connection between this joint and all joints of the fingers, we can make new sets of equations between the joints of fingers and joint of the wrist and solve them to find the dual quaternion form of the joints of the fingers as well (Equation [28]). In reality, we need to add some constraints to make the design as the shape we want.

6.5 Constrain the Motion

There are two popular methods to make a connection between joints in a robotic hand. In this project, we work on both methods and the following parts will explain each of these methods in planar case and spatial case.

6.5.1 Pulley and Belt System in Plane

In this method, due to space limitations in the area where the fingers connect to the palm, we will use one revolute joint with a standard size and shape for each finger. On the other side, we will use a pulley system in the wrist (Figure [22]). Therefore, our design has one revolute joint in complex with five pulleys with various ratios. All revolute joints in fingers are connected to this pulley system with belts. Therefore, when the pulley system connects to an actuator and rotates, each finger will move in specific ratio-metric form. Therefore, for any given task, we can calculate and design a different pulley system with a different radius within each pulley (Figure [23], Equation [20]). Also, in the final design between the wrist and each finger we added two extra rollers which allow us to change direction of belts to desired way.

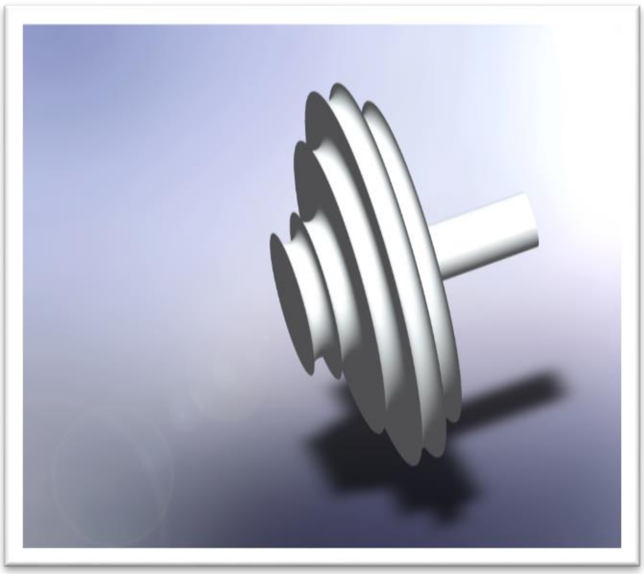


Figure 22 : *Pulley System*

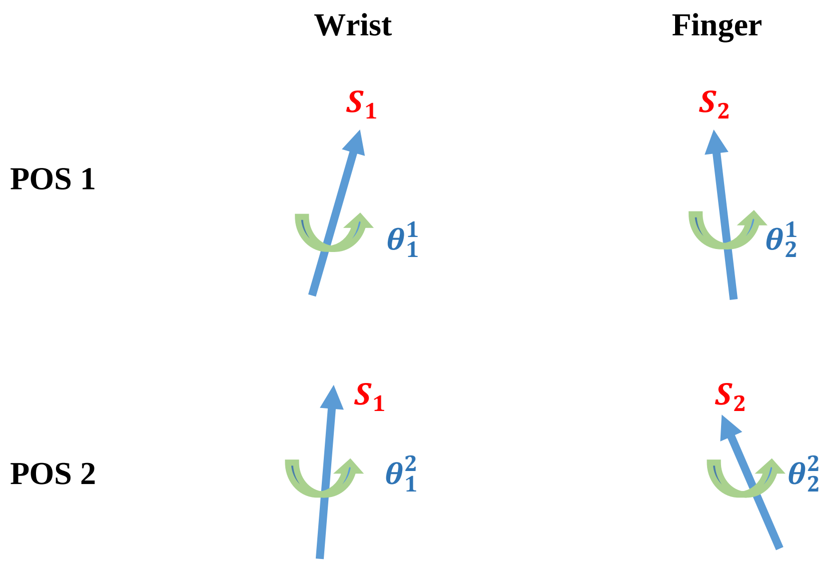


Figure 23

$$ratio = \frac{\Delta\theta_1}{\Delta\theta_2} = \frac{r_{pulley2}}{r_{pulley1}} \qquad \text{Equation (20)}$$

6.5.2 Result of in Plane Pulley Belt System

In the section 6.4, it was explained how we can find the positions of wrist and fingers joints. We then proceeded to find the positions of guide rollers when the length of belt between the pulleys become minimum, the guide rollers don't have any intersection with rollers and joints, and they are in a specific area (palm). For getting to final design, it is necessary to add some extra constraints too. Let us start with a simple example. As you can see in the Figure [24], the Pulley C1 connected to Pulley C2. In this figure we assume position of C1, radius R1 and R2 are known. In this problem we want to find the position of C2 when the length of t1t21 become minimum (α1 and α21 are unknown).

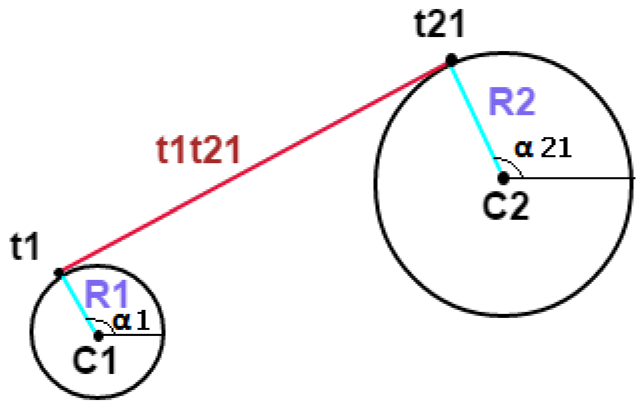


Figure 24

First we need to find t1 and t21 from equations [21] and [22].

$t1{=}c1+\{R1{*}Cos\ [\alpha 1], R1{*}Sin\ [\alpha 1]\}$ ***Equation (21)***

$t21 = c2 + \{R2 * Cos\ [\alpha 21], R2 * Sin\ [\alpha 21]\}$ ***Equation (22)***

Then from equations [23], [24], and [25], we can find a formula for length of t1t21.

$t1c1 = t1 - c1$ ***Equation (23)***

$t21c2 = t21 - c2$ ***Eauation (24)***

$t1t21 = [(c2 + t21c2) - (c1 + t1c1)].[(c2 + t21c2) - (c1 + t1c1)]$ ***Equation (25)***

Now let us go back to our five fingered hand and continue with that. As you can see in the Figure [25], there are four pulleys for each finger in the real case. Therefore, we will have three belt length calculation same as equation [25] for each finger (Equation [26]).

$Len\ 1 = [t1t21 + t23t32 + t34t4]$ ***Equation (26)***

Then we can write the same equation for all five finger and solve them at same time for minimum length of belt to find the positions of the guide rollers (Equation [27]).

$$a = \mathrm{Minimize}[\{\mathrm{len1} + \mathrm{len2} + \mathrm{len3} + \mathrm{len4} + \mathrm{len5}\}, \{\mathrm{x21}, \mathrm{x31}, \mathrm{y21}$$
$$, \mathrm{y31}, \alpha 11, \alpha 211, \alpha 231, \alpha 321, \alpha 341, \alpha 41, \mathrm{x22}, \mathrm{x32}, \mathrm{y22}, \mathrm{y32}, \alpha 12, \alpha 212$$
$$, \alpha 232, \alpha 322, \alpha 342, \alpha 42, \mathrm{x23}, \mathrm{x33}, \mathrm{y23}, \mathrm{y33}, \alpha 13, \alpha 213, \alpha 233, \alpha 323, \alpha 343,$$
$$\alpha 43, \mathrm{x24}, \mathrm{x34}, \mathrm{y24}, \mathrm{y34}, \alpha 14, \alpha 214, \alpha 234, \alpha 324, \alpha 344, \alpha 44, \mathrm{x25}, \mathrm{x35}, \mathrm{y25},$$
$$\mathrm{y35}, \alpha 15, \alpha 215, \alpha 235, \alpha 325, \alpha 345, \alpha 45\}]$$

Equation (27)

As it mentioned before for preventing the intersection between pulleys and rollers, we can add several extra constraints to Equation [27] and then solve it for minimum length of belt (Equations [28]).

Equation (28)

$$\mathrm{con1} = (\mathrm{c21} - \mathrm{c22}).(\mathrm{c21} - \mathrm{c22}) - (\mathrm{R21} + \mathrm{R22} + 0.2)\text{^}2$$

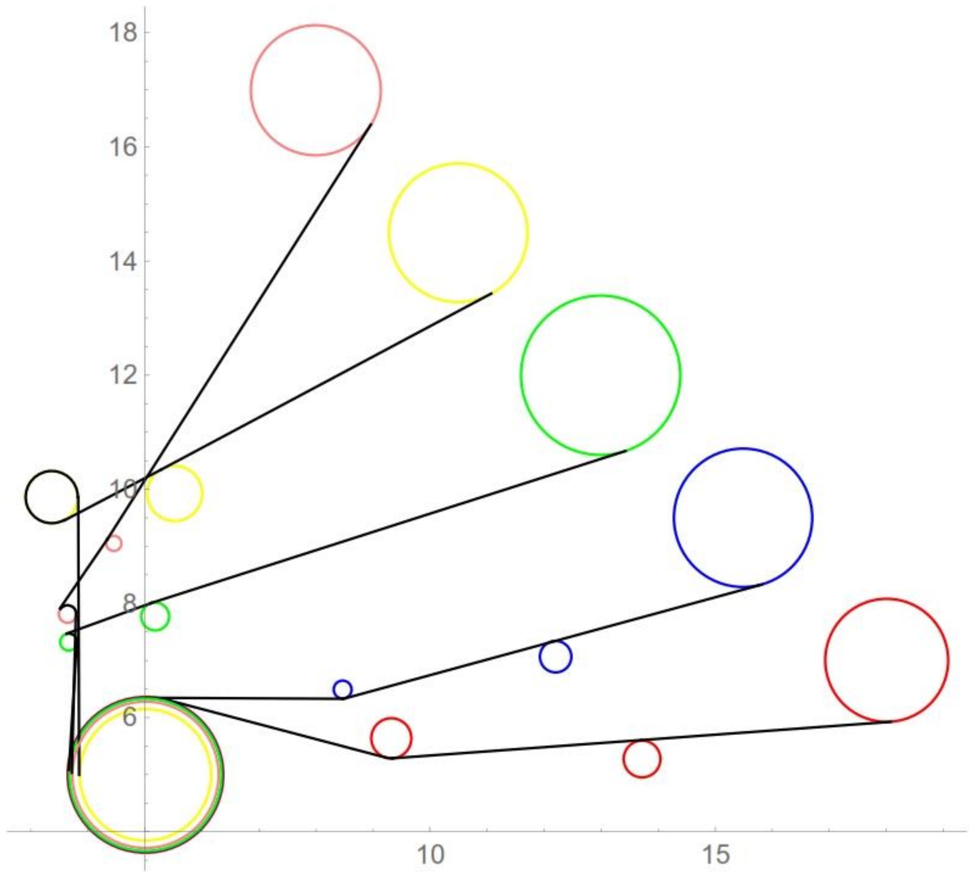


Figure 25

6.5.3 Pulley and Belt System in Space

In the process of modeling this system, there were many problems. The first problem was the connection between pulleys and belts. Actually, when two revolute joints (wrist & a finger) are not in the same plane and they are not parallel to each other, the twist force over belt causes it to come out of the rail during rotation (Figure [26]). For solving this problem first we designed a new pulley with covered parts which prevented the belt from coming out of the rail (Figure [27]).

Figure 26

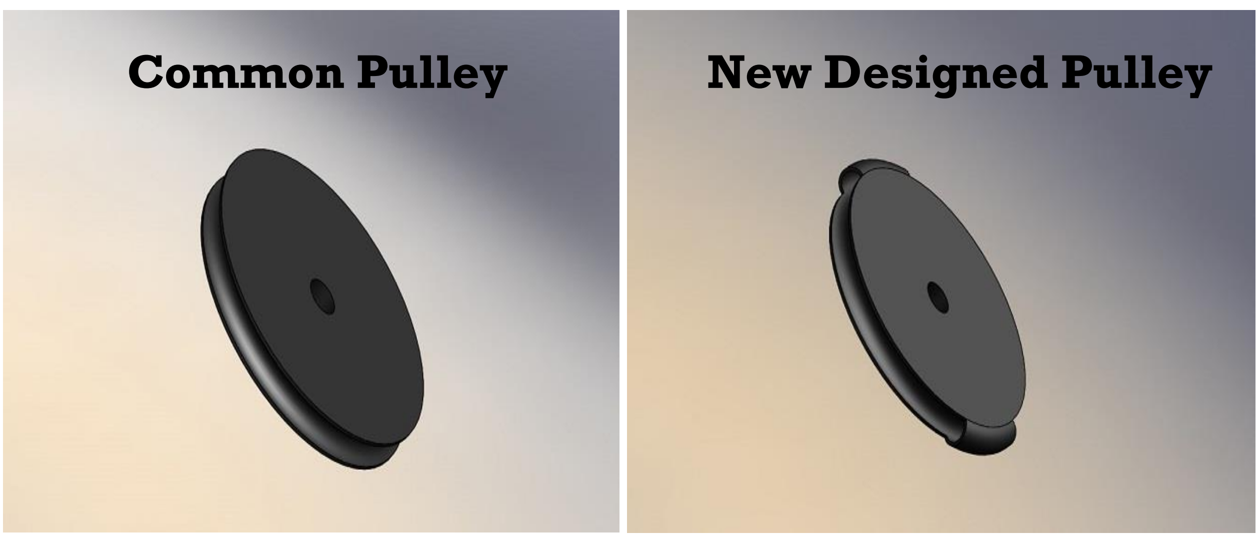


Figure 27

However, this new designed only allows the pulley to rotate for 45° and after that the covered edges prevent the belt from moving around the pulley.

A solution for this problem is mixing three revolute joints (each one can rotate around one axis) and made a complex joint (Figure [28]). Because this joint is free to rotate in any direction, the twist force by the belt causes two joints to rotate and stay in the same plane. By changing the position and oriantation of the joints, they can adapt themselves to new conditions. Using this design, the belts remain within the joints. Because of rotation, angular velocity of each joint and the belt velocity will be different. In this project we just analyzed the pulley and belt system up to this point and leave it for more research in future work.

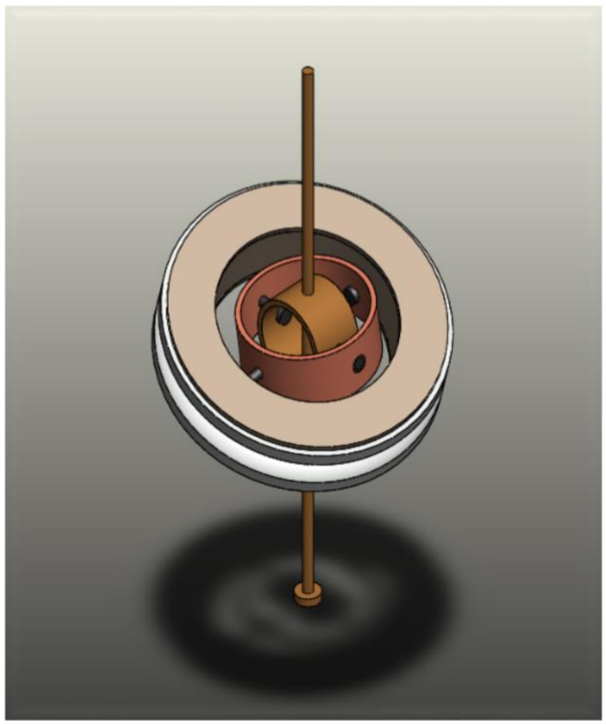


Figure 28

6.5.4 Linkage System in Space

As shown before, in reality during design of a robotic hand for showing several sings or grasping something, we usually encounter a spatial case. We present description of one finger and after that we will analyze the whole hand at once. In this design we use one joint for each finger which is connected to the one common joint in the wrist. So, we have a serial RR robot for each finger. As we know from chapter 4, this robot has 2 DOFs and according to the final aim we want to design a robot with just one DOF. Therefore, we can use the same trick that we used in the planar case and with adding some links to system decrease the number of DOFs.

One of the simplest linkages we can use in this case is a four bar linkage. As we saw in chapter 4 the mobility of this linkage in space is negative (Over Constrained Robot) and just movable in some special cases (parallel or Bennet linkage). Because in designing a hand we can't put the finger joints and wrist joint parallel to each other, the Bennet linkage will be the best option.

For linkage systems in space, same as pulley-belt system, we used synthesis method. Thus, using the positions of end-effectors in space from motion capture system. After transforming the positions of end-effectors to dual quaternion, we calculated the relative displacements. Because this hand has five fingers (five end-effectors), we have four relative displacements. The relative displacement of each end effector will be equal to relative displacement of the wrist multiplied by the relative displacement of finger joint of the same end effector [7], as shown in equation [29]. In this equation we have used exponential model of joints relative displacement.

$$e^{\hat{S}_j \frac{\Delta\theta_j}{2}} = (e^{\hat{S}_0 \frac{\Delta\theta_0}{2}}) * \hat{P}_j \qquad \textbf{\textit{Equation (29)}}$$

* Donates the conjugate unit dual quaternion

From [7], we can expand equation [29] and extract equation [30],

$$\left(cos\ \frac{\Delta\theta_j}{2} + \varepsilon sin\frac{\Delta\theta_j}{2}\right)\hat{S}_j = \left(cos\ \frac{\Delta\theta_0}{2} - sin\frac{\Delta\theta_0}{2}S_0\right)(p_{j0} + \varepsilon p_{j7} + P_j)$$

$$j = 1, \dots, 5$$

Equation (30)

Where,

$$\hat{S}_j = 0 + \varepsilon 0 + S_j = 0 + s_{jx}i + s_{jy}j + s_{jz}k + \varepsilon(s_{jx}^0 i + s_{jy}^0 j + s_{jz}^0 k + 0)$$

Equation (31)

According to equation [30], the last component of the dual quaternion must be equal to zero for each joint solution. As a result, the second part of equation [29], $(e^{\hat{S}_0\frac{\Delta\theta_0}{2}}) * \hat{P}_j$ to have also last component equal to zero and creates one equation on the parameters of the first axis S_0 from each finger equations [7],

$$cos\ \frac{\Delta\theta_0}{2}p_{j7} + sin\frac{\Delta\theta_0}{2}\left(s_0.P_j^0 + s_0^0.P_j\right) = 0, \quad j = 1, \dots, 5$$

Equation (32)

Or

$$cot\frac{\Delta\theta_0}{2} = \frac{-\left(s_0.P_j^0 + s_0^0.P_j\right)}{p_{j7}}, \qquad j = 1, \dots, 5$$

Equation (33)

Because $cot\frac{\Delta\theta_0}{2}$ is same for all five equations, we can eliminate it and end up with four linear equations in the Plucker coordinates of the axes (equation [34]) plus two Plucker constraints (equation [35]) and solve them for six unknown parameters.

$$\frac{\left(s_0.P_j^0+s_0^0.P_j\right)}{p_{j7}}=\frac{\left(s_0.P_{j+1}^0+s_0^0.P_{j+1}\right)}{p_{(j+1)7}},\quad j=1,\ldots.,4$$ ***Equation (34)***

$$s_0.s_0=1,\quad s_0.s_0^0=0$$ ***Equation (35)***

With solving these equations, we will find four solutions which two of them are repeated because of double covering of SO(3), so finally we will have two different solutions. For each of these solutions, we calculate the value of θ_0 from equation [33]. Then, with these value, we can make the quaternion form of s_0 (wrist axis) and find the quaternion of each finger axis from equation [29].

With finding the positions and orientations of wrist and finger axis, now we just need to find the position and orientation of the extra joints of Bennet Linkage for each finger. In this case we have 14 unknown (10 independent) and 17 equations (10 independent), five Bennet Linkage constraint equations (equations [18], [19]), relative displacement equality between wrist-finger joints and two other joints (8 equations) and four Plucker constraint equations (Equation [36]).

Rel01= quatmultSep [S0,S1]
Rel3141= quatmultSep [S31, S41]
Eq01= (Rel3141-Rel0) = 0 ***Equation (36)***

In the next step, we solve these equations and find the positions and orientations of whole joints of the hand. For getting to this aim, one of the Matlab toolboxes (optimtool) were used. fsolve-Nonlinear equation solving and Levenberg-Marquardt were chosen as the solver and algorithm respectively. The final error was less than 10^{-15}, so we can disregard it. You can see the dual quaternion form of all four joints of Bennet Linkage for each finger in Table [3] and Table [4] for first and second solution respectively. Notice the first joint for all Bennet Linkage is for the wrist, so it will be similar for all fingers.

Table 3 : *Dual quaternion form of all four joints of Bennet Linkage for all fingers for the First solution*

First Finger	
Wrist Joint	$0.08 + 0.02i - 0.37j + 0.92k + \varepsilon(-23.64i + 16.52j - 3.93k)$
Finger Joint	$-0.28 - 0.40i + 0.47j + 0.73k + \varepsilon(19.87i - 26.52j - 10.56k)$
Bennet 13 Joint	$0.23 + 0.01i + 0.09j + 0.95k + \varepsilon(0.01i + 0.03j + 0.26k)$
Bennet 14 Joint	$0.06 + 0.03i + 0.08j + 1.00k + \varepsilon(0.003i + 0.04j + 0.03k)$
Second Finger	
Wrist Joint	$0.08 + 0.02i - 0.37j + 0.92k + \varepsilon(-23.64i + 16.52j - 3.93k)$
Finger Joint	$-0.15 - 0.10i + 0.46j + 0.87k + \varepsilon(21.11i - 23.19j + 2.07k)$
Bennet 23 Joint	$0.27 + 0.10i + 0.35j + 0.89k + \varepsilon(0.12i + 0.23j + 0.32k)$
Bennet 24 Joint	$0.15 + 0.16i + 0.09j + 0.98k + \varepsilon(0.02i + 0.02j + 0.04k)$
Third Finger	
Wrist Joint	$0.08 + 0.02i - 0.37j + 0.92k + \varepsilon(-23.64i + 16.52j - 3.93k)$
Finger Joint	$-0.34 - 0.15i + 0.45j + 0.81k + \varepsilon(14.87i - 19.07j + 5.06k)$
Bennet 33 Joint	$0.001 + 0.005i + 0.002j + 1.00k + \varepsilon(0.003i + 0.001j + 0.004k)$
Bennet 34 Joint	$0.04 + 0.11i + 0.12j + 0.99k + \varepsilon(0.12i + 0.08j + 0.01k)$
Forth Finger	
Wrist Joint	$0.08 + 0.02i - 0.37j + 0.92k + \varepsilon(-23.64i + 16.52j - 3.93k)$
Finger Joint	$-0.35 - 0.02i + 0.56j + 0.75k + \varepsilon(12.97i - 19.90j + 7.13k)$
Bennet 43 Joint	$0.25 + 0.07i + 0.13j + 0.93k + \varepsilon(0.23i + 0.28j + 0.03k)$
Bennet 44 Joint	$0.09 + 0.07i + 0.05j + 0.99k + \varepsilon(0.04i + 0.04j + 0.03k)$
Fifth Finger	
Wrist Joint	$0.08 + 0.02i - 0.37j + 0.92k + \varepsilon(-23.64i + 16.52j - 3.93k)$
Finger Joint	$-0.05 - 0.03i + 0.16j + 0.98k + \varepsilon(23.19i - 14.98j + 4.67k)$
Bennet 53 Joint	$0.05 + 0.02i + 0.09j + 0.99k + \varepsilon(0.02i + 0.02j + 0.02k)$
Bennet 54 Joint	$0.03 + 0.06i + 0.05j + 0.99k + \varepsilon(0.14i + 0.06j + 0.03k)$

Table 4 : Dual quaternion form of all four joints of Bennet Linkage for all fingers for the Second solution

First Finger	
Wrist Joint	$-0.49 + 0.40i - 0.46j + 0.63k + \varepsilon(-5.44i - 0.56j + 5.39k)$
Finger Joint	$0.29 - 0.75i + 0.21j + 0.55k + \varepsilon(3.50i - 2.19j - 12.61k)$
Bennet 13 Joint	$0.07 + 0.18i + 0.17j + 0.93k + \varepsilon(0.24i + 0.26j + 0.28k)$
Bennet 14 Joint	$0.11 + 0.26i + 0.25j + 0.92k + \varepsilon(0.06i + 0.05j + 0.19k)$
Second Finger	
Wrist Joint	$-0.49 + 0.40i - 0.46j + 0.63k + \varepsilon(-5.44i - 0.56j + 5.39k)$
Finger Joint	$0.41 - 0.52i + 0.44j + 0.61k + \varepsilon(4.57i - 3.16j - 7.88k)$
Bennet 23 Joint	$0.33 + 0.10i + 0.38j + 0.91k + \varepsilon(0.14i + 0.08j + 0.10k)$
Bennet 24 Joint	$0.08 + 0.06i + 0.04j + 0.99k + \varepsilon(0.10i + 0.07j + 0.07k)$
Third Finger	
Wrist Joint	$-0.49 + 0.40i - 0.46j + 0.63k + \varepsilon(-5.44i - 0.56j + 5.39k)$
Finger Joint	$0.25 - 0.58i + 0.37j + 0.68k + \varepsilon(-4.06i - 3.10j - 2.13k)$
Bennet 33 Joint	$0.08 + 0.22i + 0.09j + 0.97k + \varepsilon(0.03i + 0.19j + 0.09k)$
Bennet 34 Joint	$0.06 + 0.10i + 0.02j + 0.97k + \varepsilon(0.03i + 0.23j + 0.23k)$
Forth Finger	
Wrist Joint	$-0.49 + 0.40i - 0.46j + 0.63k + \varepsilon(-5.44i - 0.56j + 5.39k)$
Finger Joint	$0.16 - 0.56i + 0.51j + 0.63k + \varepsilon(-3.19i - 1.75j - 0.88k)$
Bennet 43 Joint	$0.21 + 0.20i + 0.16j + 0.97k + \varepsilon(0.09i + 0.20j + 0.13k)$
Bennet 44 Joint	$0.09 + 0.24i + 0.22j + 0.97k + \varepsilon(0.14i + 0.16j + 0.15k)$
Fifth Finger	
Wrist Joint	$-0.49 + 0.40i - 0.46j + 0.63k + \varepsilon(-5.44i - 0.56j + 5.39k)$
Finger Joint	$0.58 - 0.27i + 0.32j + 0.70k + \varepsilon(3.30i - 2.33j - 8.00k)$
Bennet 53 Joint	$0.19 + 0.05i + 0.11j + 0.90k + \varepsilon(0.18i + 0.26j + 0.11k)$
Bennet 54 Joint	$0.28 + 0.33i + 0.10j + 0.88k + \varepsilon(0.05i + 0.14j + 0.15k)$

Finally, we can start to model the hand with one of the CAD software. In this project we use SolidWorks for modeling. As you can see in the Figure [29], we have 20 links and 16 joints which is really difficult and time consuming to model and assemble it. Therefore, in this project with help of Macro in SolidWorks and automatic hand and skeleton design program [9], I designed a five finger hand with one revolute joint for each finger and one common revolute joint in the wrist (Figure [30]).

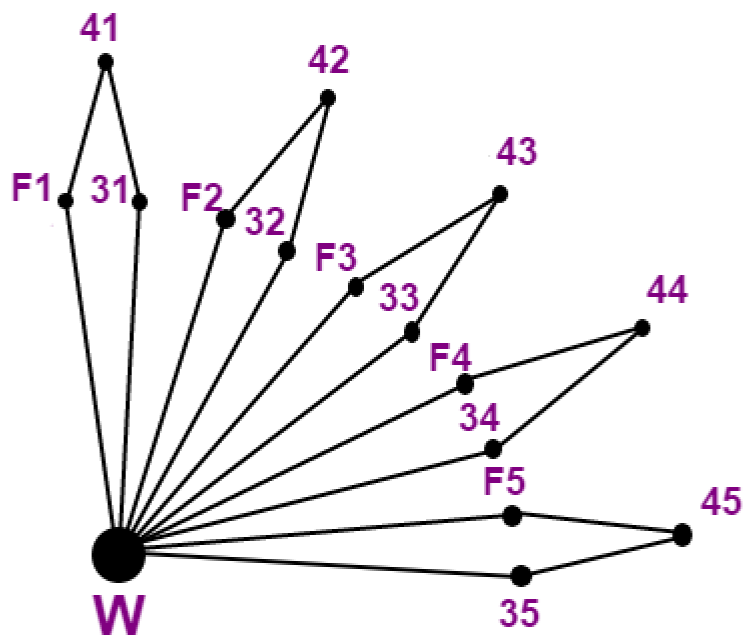


Figure 29 : *Five finger robotic hand with Bennet Linkage*

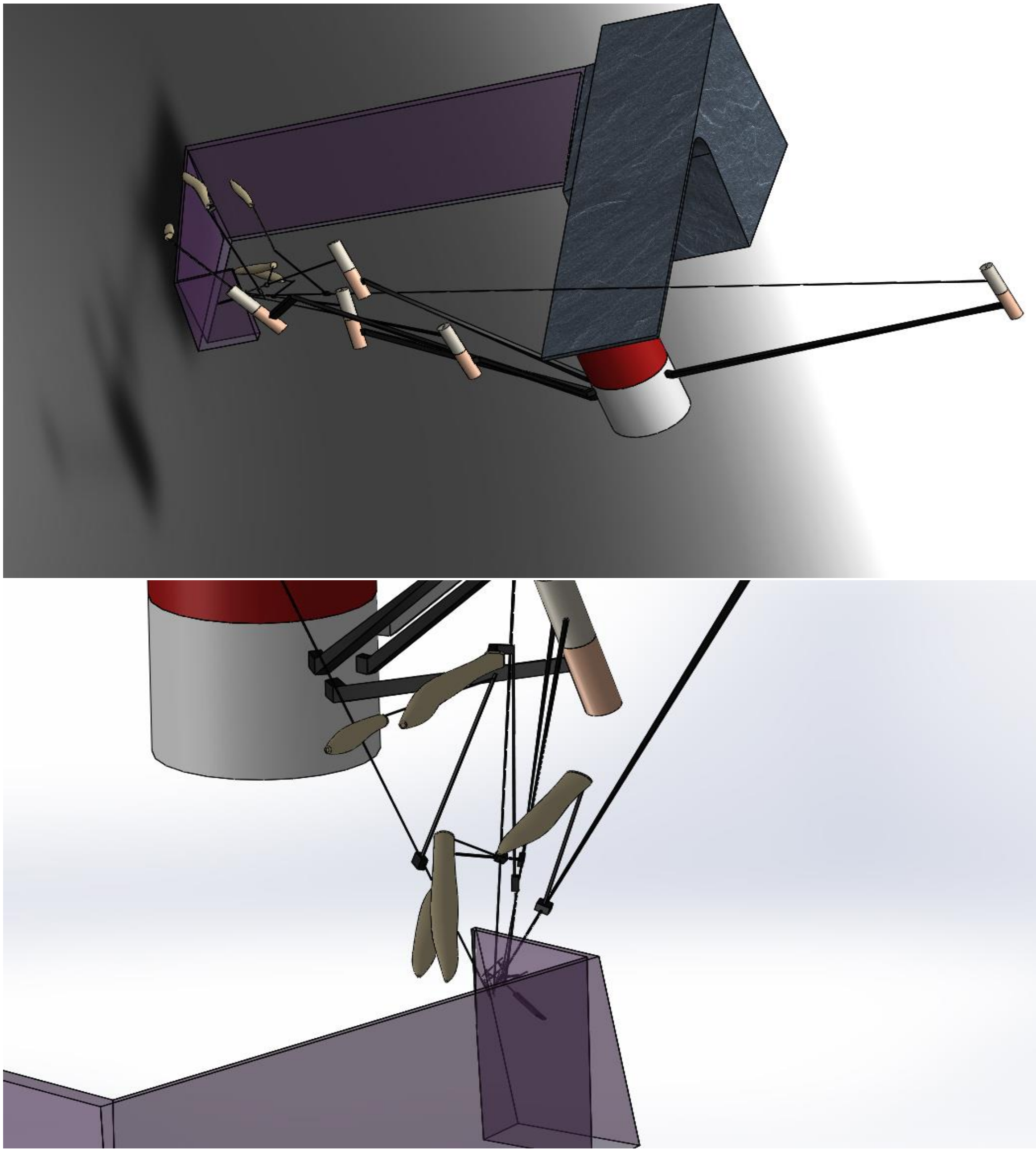


Figure 30 : *SolidWorks Model of a five finger hand with one revolute joint for each finger and one common revolute joint in the wrist in the First position*

In this chapter, first hand topology and solvability was checked for the robot and task was defined. Then, two different types of constrain the motion for planar case and spatial case were analyzed. Finally, a five-fingered robotic hand with using bennet linkage was designed.

7 Conclusions

In this project we design a five finger robotic hand with one revolute joint for each finger and one revolute joint in the wrist. We selected this hand topology for our robot to decrease the DOFs. Since in this robot, each finger can reach to two positions, this robot can perform simple tasks such as pick and place an object or showing two signs with fingers.

On the other hand, we analyze two methods for connection between joints, Pulley-Belt System and Linkage System, in the planar case and spatial case. Finally, we designed a hand which could show two different signs using Bennet Linkages for finger and wrist connections.

In the Pulley-Belt System, we chose a pulley system in the wrist with five different ratio with each one connecting to a revolute joint in a finger. The joints in the fingers had the same size, so while rotating the pulley system in the wrist, all fingers moved with various ratios to perform the task. Therefore, to perform different tasks we just need to change the pulley system. However, in the Linkage System, for designing a hand to perform different task, we need to design a new Bennet Linkage, change the length of links and angle between joints.

In future, we decide to work on Pulley-Belt System in the space and find the relation between joints and belt angular velocity. Try to design an adjustable pulley system in the wrist which we can easily switch to different tasks. On the other side, we have plan to design a hand to perform more complicated tasks with using Bennet Linkage or other Linkage systems.

8 References

[1] Tubiana, Thomine & Mackin 1998, p. 4

[2] A. M. R. Agur and M. J. Lee. Grant's Atlas of Anatomy. Lippincott Williams and Wilkins, 10th Edition, 1999.

[3] *Plagenhoef et al., 1983*

[4] E. Simo-Serra, A. Perez-Gracia, H. Moon, and N. Robson. Design of multi fingered robotic hands for finite and in-finitesimal tasks using kinematic synthesis. In Advances in

Robot Kinematics, Innsbruck, Austria, June 2012.

[5] E. Simo-Serra and A. Perez-Gracia. Kinematic synthesis using tree topologies. Mechanism and Machine Theory, 72 C:94–113, 2014.

[6] A. Makhal and A. Perez-Gracia. Solvable multi-fingered hands for exact kinematic synthesis. In *Advances in Robot Kinematics*. Ljubljiana, Slovenia, June 2014.

[7] A. Perez Gracia. Dimensional synthesis of one-jointed, multi-fingered hands. *In 3rd IFToMM Symposium on Mechanism Design for Robotics*, Aalborg, Denmark, 2-4 June, 2015.

[8] Gim Song Soh, J. Michael McCarthy. The synthesis of six-bar linkages as constrained planar 3R chains, Elsevier, February 2007

[9] Hassanzadeh, N., He, X. and Perez Gracia, A., "A Design Implementation Process for Robotic Hand Synthesis", ASME IDETC/CIE conferences, Boston, USA, August 2-4, 2015

[10] N. Robson, J. Allington, and G.S. Soh. Development of Underactuated Mechanical Fingers Based on Anthropometric Data and Anthropomorphic Tasks. Buffalo, USA, 2014. ASME.

[11] M. Ceccarelli and M. Zottola. Design and simulation of an underactuated finger mecanism for larm hand. *Robotica*, 2015.

[12] A. Ciocarlie and P. Allen. Data-driven optimization for underactuated robotic hands. In *Proc. of the 2010 International Conference on Robotics and Automation*, pages 1292–1299, Anchorage, Alaska, USA, 2010.

[13] Lung Wen Tsai, Mechanism Design, Enumeration of Kinematic Structures According to Function

[14] David H. Myszka, Machines & Mechanisms, Applied Kinematic Analysis, Forth Edition

[15] A. Perez Gracia and J. Michael McCarthy, Dimensional Synthesis of Bennett Linkages, 2003

[16] A. Perez Gracia and J. Michael McCarthy, Bennett's linkage and the cylindroid, 2002

[17] Steven M. LaValle, Planning Algorithms, Cambridge University Press, 2006

[18] Richard P. Paul, Robot Manipulators: Mathematics, Programming, and Control. The Computer Control of Robot Manipulators

[19] K.B. Shimoga, Robot Grasp Synthesis Algorithms, The International Journal of Robotics, June 1996

[20] Marco Ceccarelli, Nestor Eduardo Nava Rodriguez, and Giuseppe Carbone, Design and tests of a three finger hand with 1-DOF articulated fingers, 2006

[21] Akhil J. Madhani, Robot hand with humanoid fingers, 2011