Multi Objective Workspace Optimization of a Tricept Parallel Manipulator using Evolutionary Algorithm

Author

Syed Saad Farooq

NUST201261185MCEME35112F

Supervisor

Dr Aamer Ahmed Baqai

DEPARTMENT OF MECHANICAL ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

JAN, 2015

Multi Objective workspace Optimization of a Tricept Parallel Manipulator using Evolutionary Algorithm

Author

Syed Saad Farooq

NUST201261185MCEME35112F

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

MS Mechanical Engineering

Thesis Supervisor:

Dr Aamer Ahmed Baqai

Thesis Supervisor’s Signature:____

DEPARTMENT OF MECHANICAL ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY, ISLAMABAD

JAN, 2015

Declaration

I certify that this research work titled "*Multi Objective Optimization of a Tricept Parallel Manipulator using Evolutionary Algorithm"* is my own work and based on my thesis original work. Work has been acknowledged through proper references. It is clearly attributed the work of others whenever consultation is required and the work is purely allocated to this research degree and none of the work written here is presented to anyone else.

Signature of Student

Syed Saad Farooq

NUST201261185MCEME35112F

Language Correctness Certificate

This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university.

Signature of Student

Syed Saad Farooq

NUST201261185MCEME35112F

Signature of Supervisor

Dr Aamer Ahmed Baqai

Copyright Statement

Acknowledgement

In the Name of the ALLAH who has been so beneficent and merciful. It is he who has given me the power to complete this research work and with the blessings of ALLAH I have succeeded to defend my research work. It is that courage, self-belief and confidence that ALLAH ALMIGHTY has injected in myself to put it in that useful effort which eventually has become a source of happiness for me and for all my well-wishers. I would like to thank NUST and people associated directly or indirectly to my degree for their moral support and belief on my capabilities. I would like to mention some of these personals below.

Firstly my source of inspiration needs some serious gratitude and that is my supervisor Dr Aamer Ahmed Baqai. I like to thank him for his ultimate guidance and his credence on me and motivation lectures that has given me reasons to go swiftly with my thesis and also I would like to apologize for taking his time on many odd occasions. But I really appreciate his kindness and I always enjoyed his friendly behavior with myself.

Secondly I would like to thank Dr Sajid ullah butt, Dr Hasan Aftab and Dr Khalid. I always felt proud to have their useful tips and guidance around. I really appreciate their politeness that has given me strength to learn a lot from them.

I would like to express my feelings for my childhood friend Adnan Mehmood for his moral support and his prayers. I am grateful to him. I would also like to thank my NUST Research fellows Engr Asad Shafiq, Sardar Ameer, Nouman Masood, Ahsan, Ehsaan for the help and motivation. They always pushed me up whenever I got stuck in my thesis and I always felt lively in your company guys. I would like to thank Kamran Nazir one of my senior whose research struggle has given me a right way to excel in the future studies. I like to appreciate Haseeb Qureshi and Shaheer Hussain for their well wishes that has been always around me during my whole research work.

Finally I am very grateful to my mother who has been waiting since my whole research to see me a ms scholar. It is indeed her prayers that has ended my work with a lot of success.

Dedicated to my beloved mother whose care and prayers lead me towards that accomplishment

Abstract

Optimization in every research work is now mandatory for the purpose of manufacturing and design. Robotics is fastest growing field in the context of manipulators and the actuators. Industrialists, engineers and doctors now are well groomed in their expertise tries to adopt fast and reliable methods to improve their production by saving a reasonable amount of time. Under this interest, the researchers went into the study of power and time saving methods that can help industrialists to enhance their production rate. They felt the need to introduce the manipulators. The serial manipulators were in fashion previously but these were not meeting the requirement of the users. The parallel manipulators with small alteration in their design came and with their precision manufacturing advantages. Researchers then welcomed the optimization for the design and manufacturing. So that the users may get the enough benefits by getting the best optimized product which will eventually save time and hence increase their income as well.

The research in parallel manipulators are being done on RPS, UPS, SPS and some special structures like orthoglide and tricept mechanism. Our main focus in this work will be on tricept mechanism. We will cater on the multi objective optimization of parallel tricept mechanism. Actually it is UPS mechanism having three legs with one centered leg and having static platform on the base and moving platform on the head.

Previously it has been done on single objective optimization and genetic algorithm in evolutionary algorithm was targeted. Here it has tried to perform optimization with particle swarm algorithm and claimed it fast in some terms with genetic algorithm. Initially the jacobian and the inverse kinematic solutions has been performed. It should be noted that it is using here three performance parameters conditioning index, workspace volume and global conditioning index to optimize to design variables used in tricept mechanism. It is prismatic actuated and it has two rotations and one translation which prove the fact it has 3 DOFs (degree of freedom). Then multi objective optimization has been introduced and two strategies weighted and epsilon constraint strategies performed to give the corresponding near optimum design values for tricept mechanism.

Key Words: Inverse kinematics, Conditioning index, Workspace volume, GCI, particle swarm Optimization (PSO), Multi objective, Weighted, Epsilon Constraint strategy.

Table of Contents

List of Figures

List of Tables

INTRODUCTION

Background

In the history of the automation and industrial revolution, there was huge number of work being done by the labors and it had usually become very hectic to work for the mass production with hand and time consumed was much higher than their production. The need for the fast work was demanded by the industrialists to the researchers and the scientists to introduce such machines which can reduce their work and hence can play a vital role in increasing their production. That was the challenge for the researchers and scientists to really show their level of work and understanding in the field of control and automation. Scientists come to the conclusion that the machine should be human friendly and computerized. After many years of research, they produced a link or arm which can be controlled numerically and computerized. But that was actually not enough for the industrialists to install and take the benefits which they required. After wards they collectively produced theories for the manipulation and they finally produced robot which didn’t behave exactly like humans but it was very much useful and industrialists welcomed the achievement by the researchers. Actually what theories showed was the robot which was made up of many links and the manipulators and which was programmed numerically by the computers and later on now researches have put artificial intelligence and now a robot can behave like humans. Extensively now work is on Neural networks and the scientists are very much have approached to that work where sooner or later all the work will be performed by the robots and manipulators and humans will only inspect behind the screen.

Theory of Manipulators

Manipulators as expressed earlier are sub part of the robot and many manipulators are combining to form a robot. Further if we divide for the composition of the Manipulators which are actually made of number of links and joints [1]. One link is connected to the other link through the joint. Manipulators are controlled by the motors and drives which have computer based numerical control. [2]

Here there is a need to put some lights on the joints and their degrees of freedom. Some common joints which may be using in my work of thesis will be extensively the revolute joint, spherical joint, prismatic joints and the universal joints.

Degree of freedom (DOF) is actually the independent movement allowed by any link with respect to the other link and roughly we can estimate the number of links to the degree of freedom. Revolute joint denoted by the symbol 'R' sometimes also be said as pin joint has the rotary movement with respect to the other link and it has 1 degree of freedom. Spherical joint which is commonly known of its free rotations along the x, y, z directions and wrist joint is an example of the spherical joint. It permits the three rotations and has 3 DOF [3].Prismatic is the one used term for the sliding of the one link to the other link and can allow the translation in the one direction at a time and have 1 DOF. Universal joint is comprised of the 2 revolute joint and therefore have 2 DOF.

Applications and Motivation

- In Cable driven robot and is used to carry high loads and in configuration it has 'm' no of motors with 'm' number of strings. Also cable suspension robot is another area of parallel manipulators [4].
- Widely used in the machining operations and used for positioning and fixturing.
- Most commonly used in satellite better orientations where it has fixed base and a moving platform.
- When we talk about their application in the medical. It has amazing areas like ENT, heart problems now days, neurosurgery, artificial leg, prosthesis, CPR (cardio-pulmonary resuscitation operation) [4]. See in figure below.
- Hexapod and delta robot is one parallel configuration used for the precision and rapid manufacturing reduces the overall cost and time of manufacturing.
- Used in packaging and food industries.
- Used in the nuclear reactions where extreme attention requires for reactive material handling, used for cleaning and inspecting of pipes.
- Other application areas includes the civil works like bridge building, maintaining the huge architecture of commercial buildings, maintenance of big machines and aircrafts,

cargo services of ships and effectively in deep sea explorations and used for maintaining oil and gas facilities under sea at rigs.

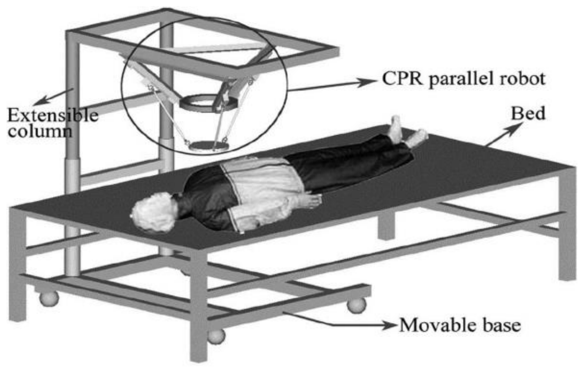


Figure CPR [4]

Contribution of Thesis

The work which will be presented here will enhance the knowledge regarding the parallel manipulators. Tricept mechanism has been chosen here as it has many application areas in medical and defense sectors. The contribution will be on Tricept mechanism optimum performance parameters that are necessary for the choice of the design variables. This work will revolve around their design that will allow the researcher to cater for their manufacturing and design perspective.

The design that is chosen here has already been optimized through the genetic algorithms. In this thesis we ought to make sure the same design through the particle swarm optimization (PSO) technique and we have brought here the optimum values of the performance index using PSO. Conditioning index and workspace volume were major performance index that will be presented later. Corresponding design variables also been optimized and shown later in the chapters. Finally analysis has been made that will correlate the results with the genetic algorithm previously done. Future suggestions will be made in the end.

CHAPTER 1: KINEMATIC CHAIN AND MANIPULATORS

1.1 Introduction to Kinematic chain

When the number of links and joints are combined, they formed a kinematic chain. Kinematic chain can be of the open loop kinematic chain and closed loop kinematic structure. Open loop kinematic chain is such a structure when one link is attached to the base and other end effecter is free. Number of links proceeds serially to the e effectors and end effectors is not attached to further link. It is commonly being known as serial link manipulators [5]. Whereas the closed loop structure comprise of the closed structure from the base to the end effectors and you can reach to end from either directions [6].These are the parallel manipulators which will be discuss in detail extensively throughout this literature. If open and closed loop both are in the structure it forms a hybrid kinematic chain manipulator.

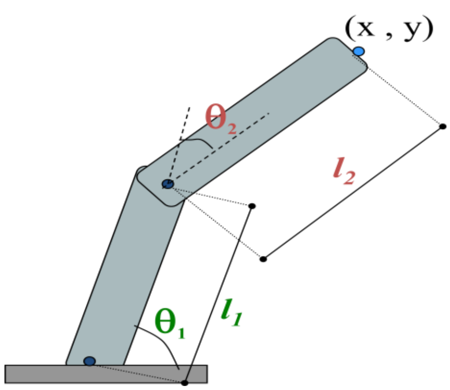


Figure 1.1 Open loop structure

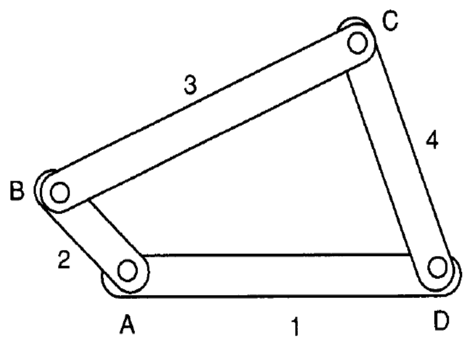


Figure 1.2 Closed loop Structure

There are two main types of manipulators

- Serial manipulator
- Parallel manipulator

1.2 Serial Manipulator

A serial manipulator is simply an open loop structure which has links interconnected by the number of joints and every joint is being actuated by the motor or drive [8].

1.21 Types of Serial Manipulators

Here are listed two types of serial manipulators which are as follows:

- Direct drive manipulator
- Conventional serial manipulator

In the direct drive, the movements of the links are dependent on one another whereas the conventional serial manipulator has the movement of links independent of one another. Direct drives has only the first motor at the start of the first link and have faster speed as compare to the conventional. Although they are bulky and heavy but in conventional machining usually the gear reduction unit is introduced [9].

In simple words this unit has small many numbers of motors separately attached to the joint of very link and that's why it can control every link independently according to our need. This usually requires great skills of programming.

Here new concept lies for every manipulator for serial as well as parallel about the workspace of the manipulator.

1.22 Advantages of serial manipulator

The main advantage of the serial is the larger workspace it occupies within its entire movement. As it is open loop structure their end effectors is mainly used as a pick or drop the materials from far off places. It has very easy structure to understand and it can easily determine the input output relationships [10]. The singularities are relatively lesser and sometimes it is observed that its singularities by just looking at the geometric configurations. It is easier to understand and can change the design of the manipulator and we can easily manipulate its mechanical design [11].

1.23 Disadvantages of the Serial Manipulator

When the topic is all about its disadvantages it came to know that apart from its larger workspace there are certain other things which include its high inertia, low stiffness and most important are low precision. It cannot get the desired precision when deals with serial manipulators [11].

1.3 Kinematics of the Serial Manipulator

The position, orientation and the frames are the main ingredients to describe the kinematics of the manipulator. In the study of kinematics locate the bodies in space. For that matter first it has to introduce some systems which are basics to describe manipulator position and orientations.

1.31 Position Analysis

Fixed frame of reference plane is first made to describe the initial location of the manipulator. Reference frame can be a bed of the lathe on which the whole body is attached or any moving CNC tool holder part. The point is that it can first assign the reference point on which the manipulator is moving. In this frame we can assign reference coordinates (x, y, z) as shown in figure 1.3. Now this is the part which is static. Now we have another end effecter frame which is Cartesian frame and at that point we have the coordinates assign as (u , v ,w) and is commonly known as the moving frame. Moving frame is denoted because end effecter is always be in moving state as shown in the figure 1.3.

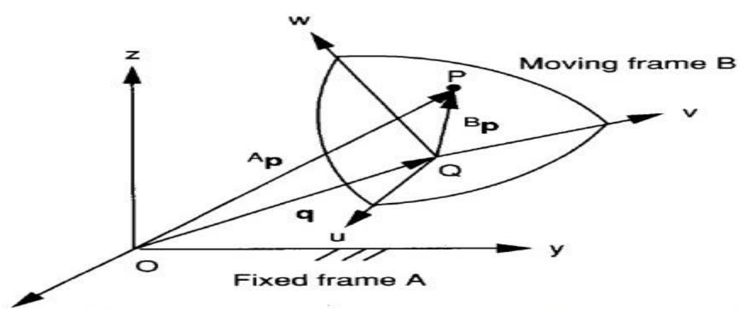


Figure 1.3 Spatial Displacement [9]

Now here it examines that the coordinates of the moving frame B is known if the reference position is known. O is its base point and on the fixed frame as A and again the (u ,v, w) are the Cartesian coordinates and P is the position. Mathematically the position vector at point P with respect to fixed frame A is:

$$P_A=\begin{bmatrix} Px \\ Py \\ Pz \end{bmatrix} \tag{1.1}$$

Where the subscript Px , Py, Pz are the projections of point P onto the fixed frame A and hence figure 1.4 demonstrates it clearly.

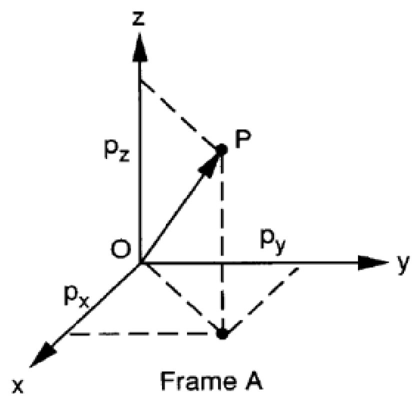


Figure 1.4 Position of P in Frame A

1.32 Orientation Analysis

Orientation can be expressed in different ways.

- Direction cosines representation
- Screw axis representation
- Euler angle representation

To describe the orientations firstly it is to consider that the moving frame B with respect to fixed frame A. direction cosines and screw angle representation is already described by [9].It will proceed through the Euler angle representation.

In Euler angle representation, three successive orientations have been used to describe rotation in the with respect to the fixed frame or the moving frame.

$$\text{Rz} = \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.2}$$

Rotation along z with angle θ

Where cθ, sθ is sinθ and cosθ.

$$\text{Rx} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha' & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \tag{1.3}$$

Rotation along x with an angle α

$$Ry = \begin{bmatrix} c\varphi & 0 & s\varphi \\ 0 & 1 & 0 \\ -s\varphi & 0 & c\varphi \end{bmatrix} \tag{1.4}$$

Rotation along y with an angle φ

These are the three successive rotations about the coordinate axis of the fixed frame.
The multiplying of all three will result in one final matrix which will result in first rotation to the last rotation. For example take the convention of rotation as follows.

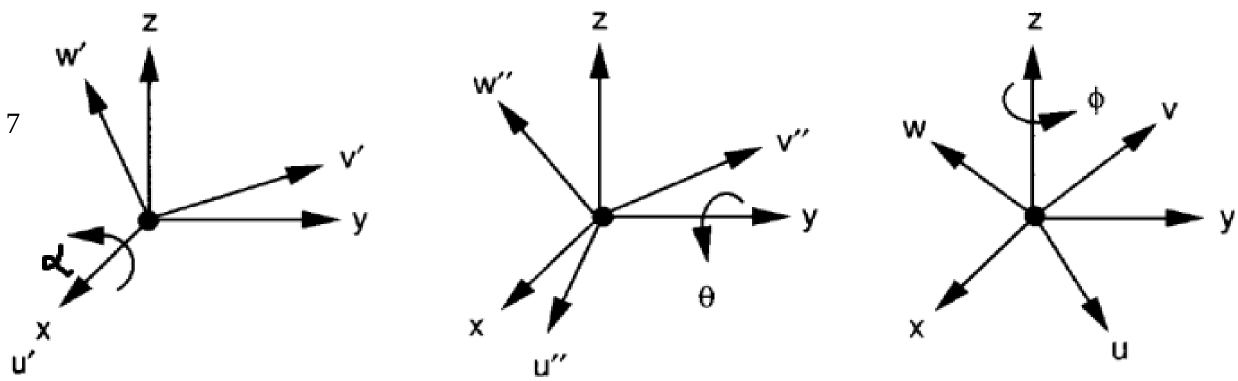


Figure 1.5 Three Successive rotation about the Fixed Frame (x, y, z) [9]

Where the (u´, v´, w´) are the axis rotated along x which is also known as roll. Second rotation results in (u´´, v´´, w´´) are the axis rotated along y and is known as pitch and finally results in (u, v, w) when rotated along the z axis which is known as yaw. This representation is also known as roll pitch yaw representation. Now R will be:

R=Rx* Ry* Rz

Mathematically after solving and multiplying three matrices R will become

$$R= \begin{bmatrix} c\varphi c\theta & c\varphi s\theta s\alpha - s\varphi c\alpha & c\varphi s\theta c\alpha + s\varphi s\alpha \\ s\varphi c\theta & s\varphi s\theta s\alpha + c\varphi c\alpha & s\varphi s\theta c\alpha - c\varphi s\alpha \\ -s\theta & c\theta s\alpha & c\theta c\alpha \end{bmatrix} \tag{1.5}$$

1.33 Homogenous Transformations

Homogenous transformation matrix is the combination of both translation and rotation and describes the mapping of one frame to another frame which has undergone the translation and rotation simultaneously.

T is the transformation matrix and it maps the vector coordinates of frame A to frame B [12].

$$T=\begin{bmatrix} R(3*3) & \vdots & P(3*1) \\ \dots\dots\dots & \vdots & \cdots\cdots\cdots \\ \gamma(1*3) & \vdots & \rho(1*1) \end{bmatrix} \tag{1.6}$$

Where R describes the rotation matrix from A to B and P shows the translation matrix from A to B. $\gamma = [0 \quad 0 \quad 0]$ and $\rho = [1]$.

1.4 Basics of Workspace

The workspace is defined as the volume of the region end effectors that can occupy throughout its maximum reach [13].Workspace is very beneficial in finding the robot trajectories. What will be its movement and it will further find its more parameters like dexterity, singularities etc. It will cover in the optimization of the workspaces and we have to deal with the jacobians.

Here now some terms and are very necessary to explain which will be useful in concepts of workspaces.

Reachable workspace is that volume of space in which the end effecter can reach all its points through at least from one orientation, whereas the most important term the dexterous workspace is the volume of the space in which the end effecter can reach its all points from all possible orientations. Basically the dexterous is the subset of the reachable workspace. It will proceed throughout this report by talking again and again of the dexterous workspace because it clearly tells the researcher about the trajectory and many other parameters by which it will estimate its movement by considering its singularities.

1.5 Parallel manipulators

Before going to the details of parallel manipulators, let readers know about the history of parallel. When the researchers went deep into the serial manipulators they thought that there is something missing in the serial manipulators like precision machining which will make the manipulators theory versatile. They undergo different experimentations by altering the links and the joint movements. Many prominent names are in the field of research. Details of research will be provided in the second chapter. Finally researchers found the new combinations of links in a different manner and they named it parallel manipulators.

Parallel manipulators are combination of links and manipulators attached in a parallel fashion. It has a moving platform and a static platform. There are some joints which are actually actuated and rests are passive to the actuated joint [14].

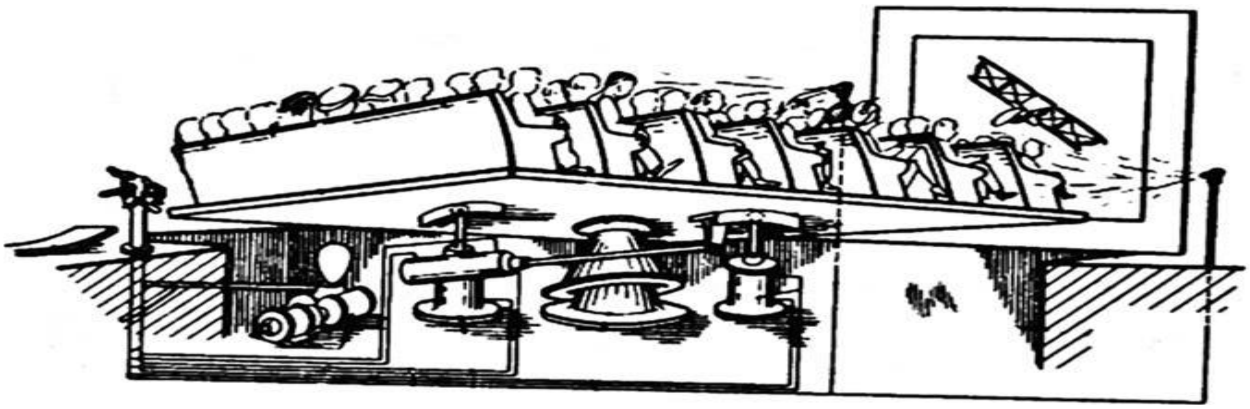


Figure 1.6 First spatial parallel mechanism patented in 1931(US patent No. 1789680)

The invention of parallel manipulators paved the way for industrials for their ultimate needs. Need of Adaptive automation was solved by this research of parallel manipulators. Parallel manipulators are famous for its rapid acceleration and immediate precise movements as compared to the conventional machining. It has mechanical simplicity in its structure and requires less installation efforts. It has less moving weight [12].

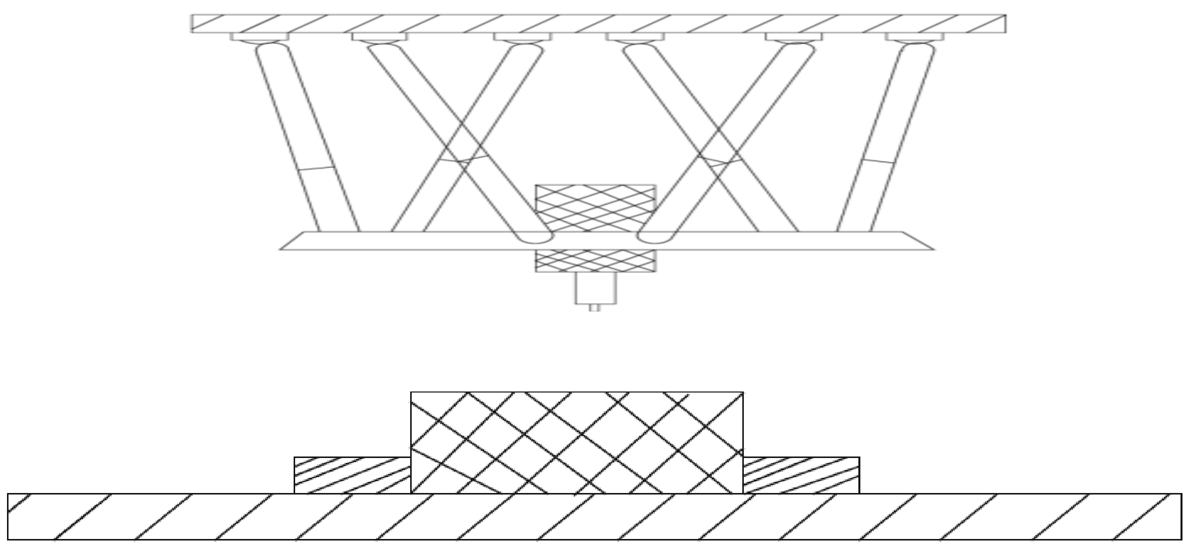


Figure 1.7 six-axis parallel hexapod [15]

Here are some advantages of parallel manipulators.

1.51 Advantages of Parallel Manipulators

- High rigidity.
- High precision.
- More flexible than serial manipulator.
- More capable of bearing high loads.
- Highly used for accuracy of machining.

If there are some advantages there is always the faulty side which is expressed as disadvantages.

1.52 Disadvantages of Parallel Manipulator

The main drawback of using parallel manipulator is a very limited workspace. It cannot perform too far off places. End effecter is limited to a certain workspace as it is a combination of links parallel attached. Parallel manipulators have complex inputs and outputs solutions and very difficult to find singularities as it has high number of singularities [16].

1.53 Types of Parallel Manipulator

There are several types based on the joints and links formation and as these types are made on the combinations of the joints so these are not only types. These are just described to get the overall introduction of the parallel manipulators.

- Gough Stewart Platform:

This form of structure is a basic purposed architecture and is of 6dof and has spherical prismatic spherical architecture [14].

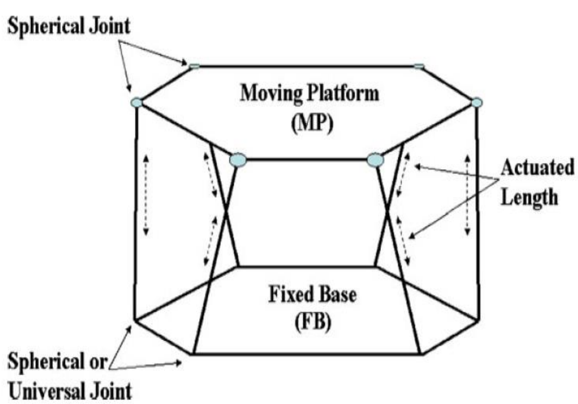


Figure 1.8 Stewart platform [14]

- 3 DOF UPU (universal prismatic universal)

It has been explained by many researchers and has illustrated its workspace and singularities concept [19].

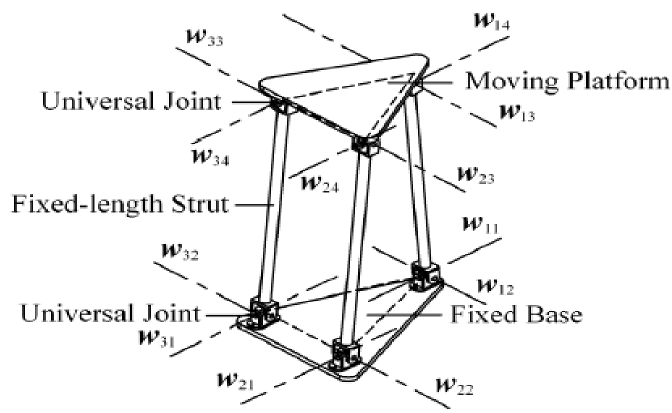


Figure 1.9 UPU [19]

- 3DOF PRS (prismatic revolute spherical)

This has explained by [20] and it has described the optimization of its workspace based on interval analysis. Basic structure is as follows.

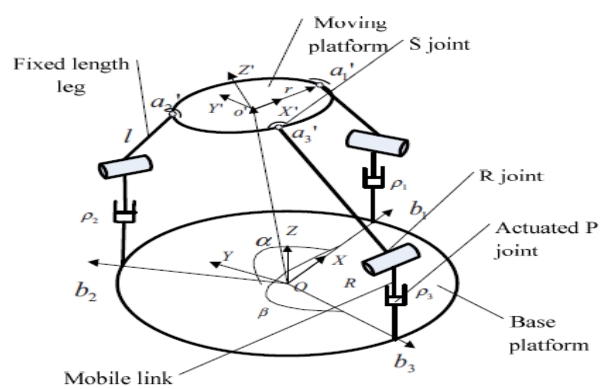


Figure 1.10 PRS [20]

- 3 DOF RRR manipulator(Revolute Revolute Revolute)

This form of architecture has been well explained by [18] and [16]. This architecture has all joints revolute and does not possess translations.

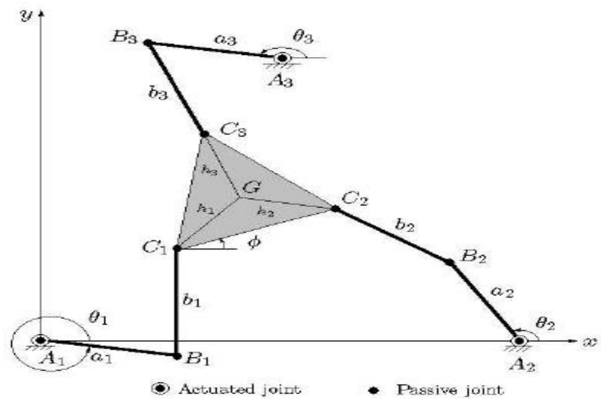


Figure 1.11 RRR [16]

- 6 DOF SPS architecture(Spherical Prismatic Spherical)

This structure has 6 degree of freedom and has explained by [14] and has described the overall kinematic design and its workspace.

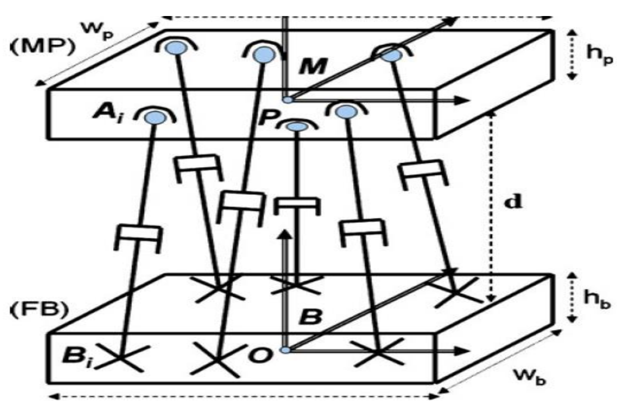


Figure 1.12 SPS [14]

Latest additions in the research of the parallel manipulators are Tricept and Orthoglide parallel manipulators. These both are very useful in practical implementations in industries as well as medical usage.

- 3 DOF Tricept manipulators

This type of structures has been well explained by [21] and [22]. Actually it has three legs with prismatic actuated and center leg has UPS architecture which is connected from base to the moving platform above.it has prismatic actuators.

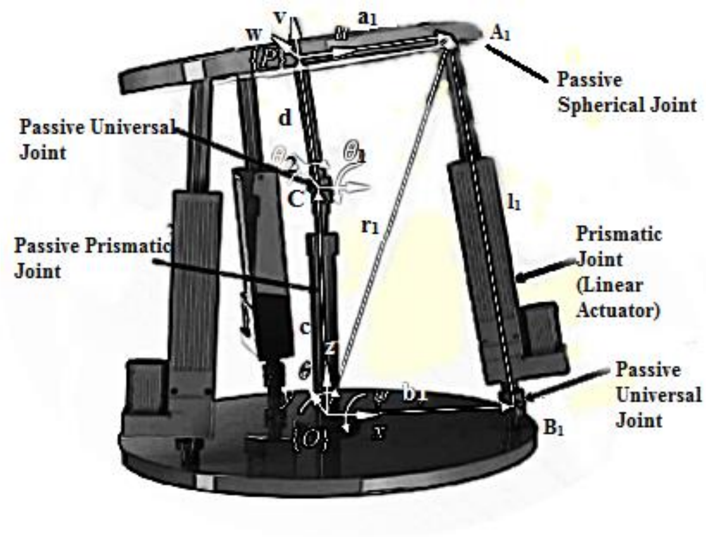


Figure 1.13 Tricept [22]

- 3 DOF Orthoglide parallel mechanism

This mechanism has been illustrated by [23]. This structure moves in the x, y, z directions having fixed orientation and heavily used for the machining purposes.

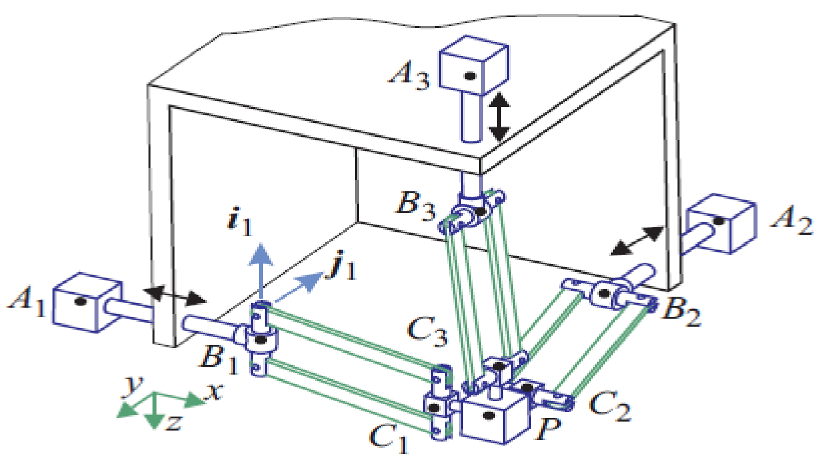


Figure 1.14 Orthoglide [23]

1.6 Workspace of Parallel Manipulators

The workspace of a parallel manipulator is very useful feature in determine its trajectories and the characteristics. There are two main types of workspace. First one is the finding the design of the manipulators with the prescribed workspace. [17] and [18] and second is to maximize the workspace by changing the geometry of a workspace [16].

CHAPTER 2: PARALLEL MANIPULATOR AND ITS WORKSPACE

2.1 Literature Survey

Parallel manipulators caught many scientists attention towards itself in the mid-nineties. They started working on the precision machining and the parallel manipulators which can be applied for the industrial purposes.

There is no need of going behind the early nineties. Here it will elaborate working on parallel manipulators after 1940's. Literature shows that people were not satisfied with the use of serial manipulators and of the draw back that serial are not accurate enough to apply in the industries. In 1931 Man named James E Gwinnett firstly made a thing which was indeed very amazing invention for the entertainment industry. He made a spherical parallel robot (Fig 1.6) platform. It was a huge effort as at that time movies had just started their journey towards the colorful pictures and sound effects. It was a random invention and people didn't understand his invention as it had complex degree of freedom.

After that people collectively sit together to excel in that field and for the first time in 1942 pollard structure broke the silence and produced his three branch parallel robot having the 3 DOFs. It has a structure which has the universal joints as well as the ball joints. Both the arms are controlled the back end motors and strings are attached which connects base to the end effecter. It was used primarily for the spray painting but it was never put into practice.

It was a time when the researchers were putting their level best to produce one such parallel machine which can be presented as a sample for the further research. In the mean while a man in 1947 named Dr Eric Gough presented his research on parallel robots and it was functional in 1954 and that was variable six struts octahedral hexapod. It was a huge invention and paved the way for the scientists to have that strong existing base for the future of parallel robots.

The universal tire testing machine was built to made use of this phenomena by Dr Eric Gough and it has the property of inspecting the tire characteristics under combined loading effects. When asked for the origin of this hexapod, Dr Eric Gough replied that it was tried by many researchers but it was long forbidden. Multi axis simulation table was of that kind and was built by Dr Hubert long ago this Gough platform.

After that Stewart in 1965 made a flight simulator model which was likely to be that octahedral hexapod. In the main while Klaus Cappel made a motion simulator. It was actually the same as octahedral hexapod.

After that many researchers come and enhanced their work.

2.11 Rapid Contributions

[24] Presented the general idea for the calculation and gave the idea to condition numbers and global indexes later. Also they had used searching technique for the optimization. Tsai also made a valuable contribution in describing the difference between the serial and parallel manipulators and has its famous book namely robot analysis. "The mechanics of serial and parallel manipulators". Some others used sequential quadratic and some had used the genetic algorithms for the tasks. In 1997 Richard optimized the workspace based on dexterity and GCI and system have 3dof translational platform. After that [21] proposed the Tricept robot kinematics and did its workspace optimization using different parameters. In December 2003 Hui Cheng worked on the dynamics and control of parallel manipulators [16]. In 2005 compared different manipulators and their optimization based on dexterity. While some other like Tanio tanev proposed a paper for the hybrid serial and parallel manipulator optimization in 2006. In 2007, G.Pond and J.A.Carretero expressed the quantitative analysis of manipulators based on the dexterity and dexterous workspace. In 2010 [14] proposed an optimization for the 6 DOF manipulator. In 2011 Ming Z. Huang designs the planar manipulator on the basis of dexterity. In 2011 [19] developed kinematic structure of 3DOF UPU structure and worked on the optimization and singularities. In 2012 [20] proposed a 3 DOF PRS optimization based on interval analysis.

2.2 New Research and Advancements

Gough Stewart platform lead researchers to go rapid through their work [25, 26, 27]. Here there is a need to describe some important terms which will be very common in mostly research works and people will be using frequently those terms to describe the working of the parallel structures.

As it has been previously described the examples and types of parallel manipulators. Now it will be describing the important term workspaces and the parameter on which it will check either the workspace is right within its constraints is dexterity. With these basic terms there are also other parameters for the checking of workspace and optimize them. Before the dexterity understanding condition number firstly should be known. Jacobian matrix is a one which is the base matrix for all the calculations. From now onwards we are starting the general scheme for the manipulators movement, their inverse kinematic and forward kinematic solutions and also difference between the closed form of solutions and also there will be some concept about the DH parameters to find the inverse solutions, some concepts about the singularities and what are the minimum and maximum singular values and how they are beneficial for the kinematic analysis of manipulators.

First step is to find the DOFs of the manipulator if planer and if the links are attached with each other in a complex fashion.

Hunt in 1978 purposed a general mobility criterion of the planer manipulator and tries to simulate it to complex ones.

$$L=6(n-g-1) + \sum_{i=1}^{g} fi \tag{2.1}$$

Where

L=total DOFs no in kinematic chain

n=number of rigid bodies in closed form chain

g=no of joints

fi= DOF of ith joint

It was used by many researchers and proved to bd very beneficial and an accurate one but it later found some errors when we go to the complex kinematic chains. It failed to give results in helical and circular type chains and it usually gave the inaccurate results and comparisons. The idea reveals around the circle to use the jacobian matrix to find the DOFs of the kinematic chain and it was first discussed by Freudenstein in 1962 and later by Angeles in 1987. Then later through many mathematical calculations it was found that actually ‘L’ is actually equals to the dimensions of null space of a jacobian [24].

$$L=\dim [(N (J)]$$

Where L is equal to the DOF and N denotes the null space of J which is jacobian. Null space can be found when it has found the jacobian matrix of the manipulators. Mathematically if it has the no of column of matrix which are independent of each other, it can say that the vector or matrix has full rank. Rank is the no of non-zero columns or rows in a matrix. Null space is actually the set of values or vectors (x) of matrix when multiply with Matrix A produces zero.

$$A(x) = 0$$

And actually dimensions are the sets of the vectors which will satisfy above relation and it is also called the nullity of the matrix. Actually the rank and nullity combines to form a matrix. Detail information about the origination of the concept can be reviewed in [24].

2.3 Velocity Analysis

Velocity analysis is the building point to create the jacobians which will give the reader many important concepts and useful information. Let do some steps to go towards the finding of jacobian of serial manipulators.

If it moves from joint space to the Cartesian space it is known as direct kinematics but if it moves from Cartesian space to the joint space it is commonly known as inverse kinematics. Jacobian is a relationship between the Cartesian and the joint space. Values of the jacobians are differential. Basic relation of which jacobian can be made is as follows:

Let

$$Y_i = f_i\,(x_j) \tag{2.2}$$

Where i=1,…,m;

And j=1,…n

If we take the differential of above equation on both sides, we get:

$$\partial \mathrm{Yi} = \sum_{j=1}^{n} (\partial \mathrm{fi} / \partial \mathrm{Xj}).\partial \mathrm{Xj} \tag{2.3}$$

Where if it writes this in matrix form, relation becomes $\partial Y = J\partial X$ and J here describes the jacobian matrix.

In robotics it deals with the jacobian matrix in terms of velocity and the relation becomes

$$V=J\dot{q} \tag{2.4}$$

Where q is the rotational and translational speeds at the joints and v is the velocities of the end effecter in Cartesian space. Here note that it is interested in calculating the inverse of the jacobian and it comes to know that the relation

$$V=J^{-1}\dot{q} \tag{2.4a}$$

Where J^{-1} only exists when there 6 joints in a robot or machine which concludes that if there is a robot less than 6 joints its inverse will give the joint space in the subspace of Cartesian space. Here one more concept lies when we talk about the singularities in a manipulator and which states that if

$$\det (J)=0 \tag{2.4b}$$

There prevails a singular relationship between the links and the inverse will not exist in fact it cannot move for the inverse kinematics [24].Actually singularity plays a vital role for the translation and rotational analysis and velocity analysis. Most importantly singularity occurs when the center line joining the two links becomes parallel or become co linear then both the links will stuck and will exhibit no movement at that very pose. It will be explaining some more singularity concepts and how to find that poses of singularity later on.

This concludes here the basic jacobian theory of serial manipulators.

Now it will proceed towards the parallel manipulator singularities and further kinematics.

If it can estimates the degrees of freedom we can just see the actuated joint in the whole structure. Also the constraint equation is

$$F(x, q) =0 \tag{2.5}$$

Then if differentiate it with respect to time it will give us the two new generations of jacobian in parallel robotics. Equation becomes

$$J_x \dot{x} = J_q \dot{q} \tag{2.6}$$

Where the J_x denotes the forward jacobian and J_q represents the inverse jacobian [24] [16] .These forward and inverse matrices plays an extensive role in the overall analysis of the condition numbers as well as global conditioning indices as well as inverse condition number singularities determination plus the stiffness and dexterity evaluation [16].

If it simplifies the above equation it will find the jacobian matrix to be rearranged in the form

$$J\dot{x} = \dot{q} \tag{2.6a}$$

Where $J=J_q^{-1}J_x$

Here note that the equation 2.6a is already in the inverse desired form so no need to again take inverse form. As the jacobian describes the rate of movement between the joint and Cartesian space and the notice able comparison between these parallel calculations with serial one as it has to take inverse of jacobians and the joint rate has to be determine for its movement in Cartesian space and that create more effort and more calculations between the joint and Cartesian workspace as compared to parallel robots.

Important point to understand that the manipulators tasks are all performed in the Cartesian space and the actuators work in joint space [28].

2.4 Singularities Concepts

Some small concepts are revised here for better physical understanding. Some points are as follows:

- When the lines of axis of links intersect at one common point, singularity occurs [24]
- Whenever the links becomes collinear during its movement.
- Whenever links become parallel in their design or movement [9]

  Now if it mathematically elaborates the characteristics of jacobian matrix. It will come to know that:
- Firstly if the determinant of the jacobian matrix becomes zero. Singularity occurs.
- When the jacobian matrix determinant reaches infinity, it can say that any of the elements in the jacobian matrix has a denominator zero. Singularity occurs [29]
- Now another state if the jacobian matrix when the element in the jacobian matrix exhibits sudden value of 0/0. That will create the rank deficient matrix and that matrix vectors are now dependent of each other. At this stage the motor at joints becomes over controlled and its arm can harm people around working on this as motor begun to rotate irregularly [9] .

Now after it come to know about the singularity existence. A relation is also existed in the literature by which it can find that poses on which the singularities exist. Many people used their own numerical as well geometric procedures for the singular values. It is stated by [29] that

$$J=U\epsilon V^T \tag{2.7}$$

V=Eigen vectors of J^TJ

U=Eigen vectors of JJ^T

ϵ =diagonal matrix containing singular values of J

If see geometrically, the transformations occurring by the figure

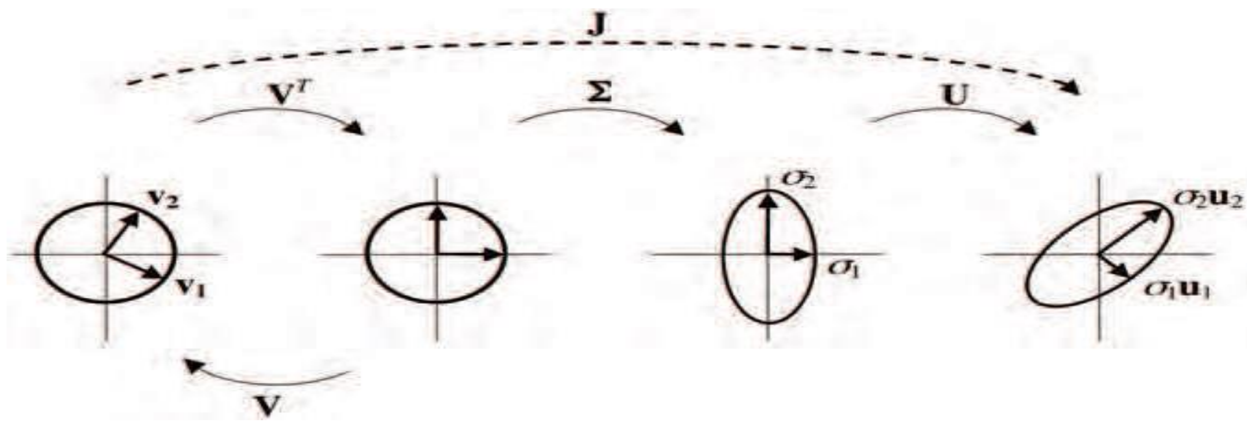


Figure 2.1 The geometrical view of starting from L.H.S to R.H.S 1) Rotation 2) Scaling 3) Rotation [29]

2.5 Types of Approaches to Find the Kinematic Solutions

- Geometrical
- Analytical
- Numerical

The approach which it will be using in this report is geometric approach and analytical simultaneously. The numerical approach is adopted by many people and they have used the Denavit Hartenberg conventions commonly known as DH parameters. Now it will explain some important steps to find the kinematics of solutions by DH parameters for forward kinematics.

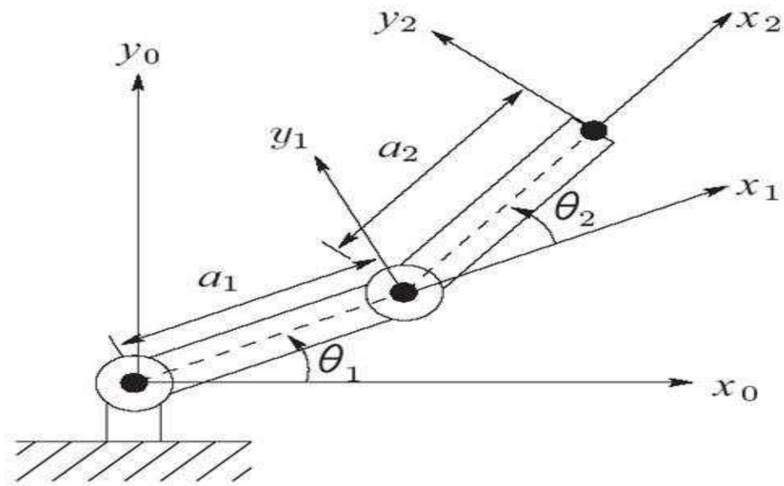


Figure 2.2 Planer 2DOF serial manipulator where z is outward of the plane [28].

Nomenclature basis for DH parameters:

a_i : denotes for the link length between O_0 and O_1 (projected along X_1)

α_i : represents link twist , angle between Z_o and Z_1 (measures along X_1)

d_i : represents link offset, distance between O_0 and O_1(projected along Z_0)

θ_I : represent the joint angle, distance between X_0 and X_1(measured along Z_0)

Link i	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1
2	a_2	0	0	θ_2

Table 2.1 D-H parameters for a 2 DOF planer serial robot

The overall transformation matrix will be as follows: [28]

$$\mathbf{A}_i^{i-1}(q_i) = \mathbf{A}_{i'}^{i-1}\mathbf{A}_i^{i'} = \begin{bmatrix} c_{\vartheta_i} & -s_{\vartheta_i}c_{\alpha_i} & s_{\vartheta_i}s_{\alpha_i} & a_i c_{\vartheta_i} \\ s_{\vartheta_i} & c_{\vartheta_i}c_{\alpha_i} & -c_{\vartheta_i}s_{\alpha_i} & a_i s_{\vartheta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.8}$$

D-H parameters are usually avoided when there come across complex kinematic chains. They usually show the degeneracy and cannot useful for the dexterity analysis. After this calculation it will easily find the inverse solutions.

Until now it has described basics to calculate the forward and inverse kinematic of serial manipulators and parallel manipulators. Now after calculating the singularities now looking forward to increase our understanding toward the condition number.

2.6 Basic Parameters for the Analysis of Parallel Manipulator Workspace

2.61 Condition number or Inverse Condition number

Condition number actually explains the regularity of the workspace which is an important feature of dexterity. It is actually a check of dexterity. Condition number can be calculated by a formula [16] [29].

$$K(J) = \frac{\sigma\max(J)}{\sigma\min(J)} \tag{2.9}$$

σmin is the minimum singular value of the jacobian matrix and σmax is the maximum singular values of jacobian matrix. Its values ranges between 1 to ∞ .And inverse condition number can be expressed then as

Inverse condition number 'k' =1/K

Thus k always lies between [0, 1].

This index is very useful and some writers also uses local conditioning index to describe the k. this condition number as explains from the formula uses the singular values of jacobian so it better explains the singularities and links nearness to singularities. Regularity and the mesh of neighborly values and also explains the properties of Isotropy. Furthermore it explains the error in the design and stiffness. Then also it explains the acceleration and velocity analysis [16].

There is also a derived mathematical formula to define this condition number [24].

$$K = ||J||*||J^{-1}|| \tag{2.10}$$

Where ||.|| denotes the Frobenius norm of matrix J and J^{-1}

2.62 More concepts on Condition Number

Question arise in the minds of a reader and somehow we will try to explain it through quantitatively.

What is dexterous workspace and what is its correspondence with the condition number

Answer somehow needs analytical thinking here. First as we have described the above relation of singularities. When values in the jacobian matrix become infinity we will find that system out of control so in that case condition number also approached infinity [29].

Isotropy or Isotropic conditions:

Isotropy is the term used to describe the ideal continuous workspace of the manipulator and if we relate this isotropy with the concept of condition number it closely relates in such a manner that condition number is 1we can say that the link is isotropic.

2.63 Dexterity and Dexterous workspace

Dexterous workspace is the workspace or volume calculated by such poses through which we are getting the condition numbers below a certain ending value of the condition set number. Dexterity take the effect of this analysis in such a way if the condition number is lower we get the dexterous workspace lower but the dexterity will be higher. There is inversely relationship between the dexterity and the Dexterous workspace [29]. Also we can get the Ideal dexterity at isotropic conditions. Now if we talk about the physical state of the actuators at this ideal dexterity. We come to the conclusion that all the actuators are putting equally the same work or effort at this state.

When we talk of the inverse condition number we can say that at K=0 the singular configuration has achieved and also when K=1 we can say that ideal dexterity has reached. Angeles (1991).

2.64 Global Conditioning Index (GCI)

It is based on the requirement whether the user needs the local conditioning or the global conditioning. If the user want the results to be with respect to global. He should use the global conditioning index for its simulation results [24].

As we know that from equation 2.5

$$F(x,q)=0$$

Then by differentiating we get the relation

$$A\dot{x}=-B\dot{q}$$

Also we get the jacobian as

$$J=-B^{-1}A$$

Where B represents the area od the workspace and A can be found as:

$$A=\int_{\varphi}\int_{y}\int_{x}\left(\frac{1}{K}\right)dx\,dy d\varphi \tag{2.11}$$

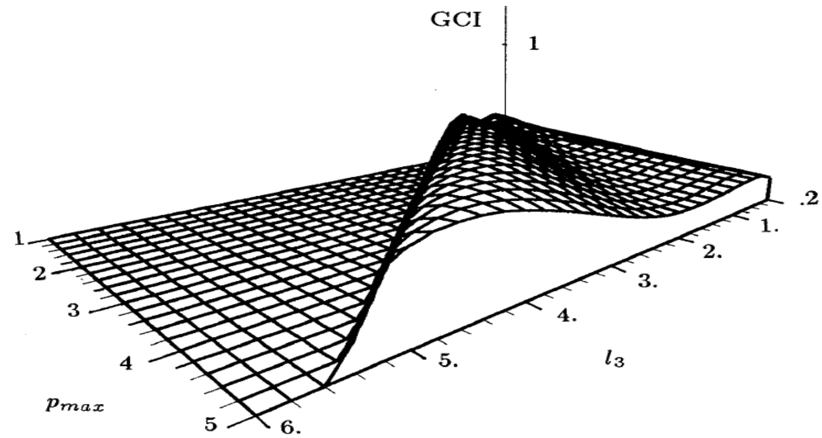


Figure 2.3 GCI of the planer manipulator with P_{max} and l_3 on other axis [24]

Actually the global index is the average of the conditioning index over a certain volume.

2.7 Examples to calculate the Workspaces of Parallel Manipulators based on the Condition Numbers

1) Planer 5 bar parallel manipulator

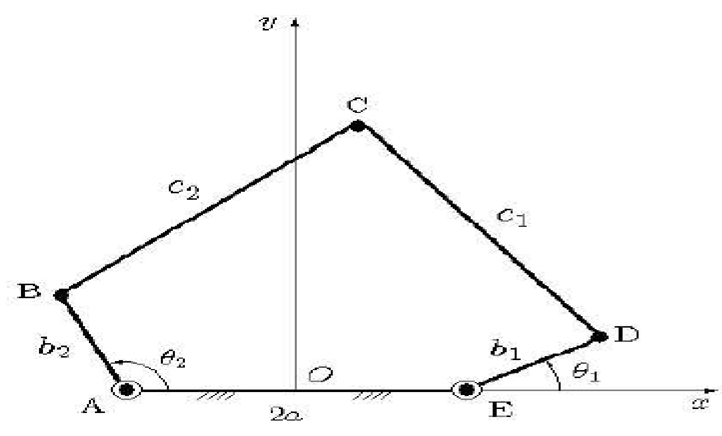


Figure 2.4 Five Bar Parallel Planer linkage

This linkage has been explained by [16]. Where A and E are actuated joints while others are passive. The writer used here closed loop equations:

$$(x-a-b_1\cos\theta_1)^2+(y-b_1\sin\theta_1)^2=c_1^2$$
$$(x+a-b_2\cos\theta_2)^2+(y-b_2\cos\theta_2)^2=c_2^2$$

Now for the jacobian matrix differentiate it with respect to time we get the following relation as

$$J_x\dot{x} = J_q\dot{q}$$

it found that J_x and J_q as the matrix of 2 by 2

$$J_x=\begin{bmatrix} x-a-b\cos\theta 1 & y-b\sin\theta 1 \\ x+a-b\cos\theta 2 & y-b\sin\theta 2 \end{bmatrix} \tag{2.12}$$

$$J_q=\begin{bmatrix} by\cos\theta 1-(x-a)\sin\theta 1 & 0 \\ 0 & by\cos\theta 2-(x+a)b\sin\theta 2 \end{bmatrix} \tag{2.13}$$

Here he has used the inverse condition number and the space utilization index to make the effective regular workspace.

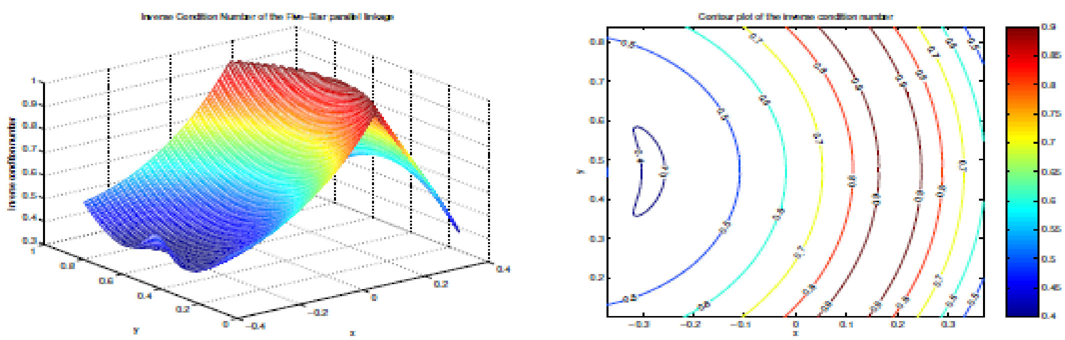


Figure 2.5 left side: the inverse condition number versus the x and y. right side: the inverse condition number versus the x and y contour.

2) <u>RRR parallel manipulator</u>

Using figure 11 for analyzing the 3 revolute actuated joints. It will repeat the same procedure of the closed loop equations to get the jacobians [16] [24].

According the loop closure equation:

$A_iG+GC_i = A_iB_i+B_iCi$

After some calculations we found the J_x and J_q. [16]

Then calculating the inverse condition number ,the graph is shown as:

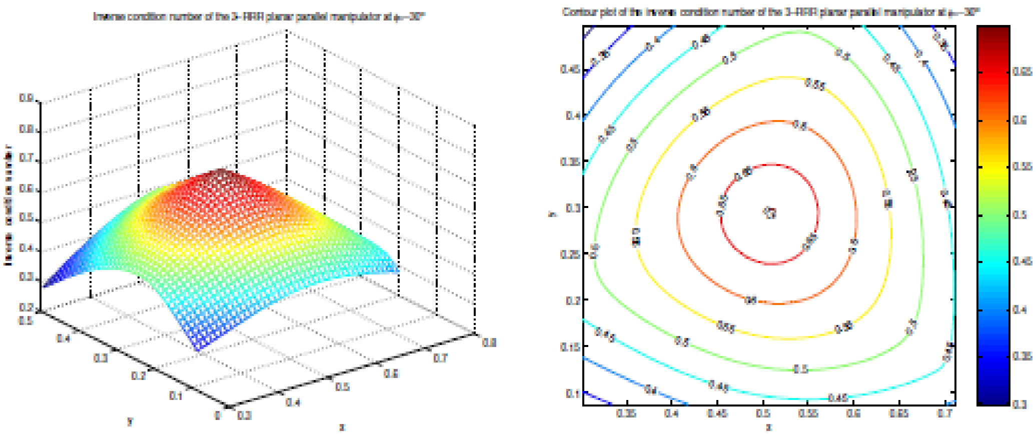


Figure 2.6 left side: inverse condition number versus x and y at $\varphi = -30°$ .Right side: inverse condition number versus x and y contour at $\varphi = -30°$

CHAPTER 3: TRICEPT PARALLEL MECHANISM AND OPTIMIZATION

3.1 Purpose to Select Tricept Manipulators

Tricept manipulators have become the source of interest for the researchers. Many writers are focusing on the complex kinematic research and some writers have moved their research to Orthoglide. Complex kinematic here means the whole structure is consisted of the combinations of different joints with one actuated joint. Tricept manipulators have much space to work on and as it is useful for the industrial and medical purposes. This Tricept being very interesting and therefore tried to enhance knowledge in the field of tricept manipulators.

3.2 Literature Survey on Tricept Mechanism

[21] Explains the kinematic solutions of the tricept robot but instead of moving platform upwards it has downward. Moving platform attached by the base upwards. Also it has three legs with three prismatic actuators. All links are attached to the moving platform through the 3 spherical joints. He had used the close loop direct kinematics approach for the workspace optimization and also based on the parameters like condition numbers, he had generated the certain graphs of velocity and time for manipulator movement.

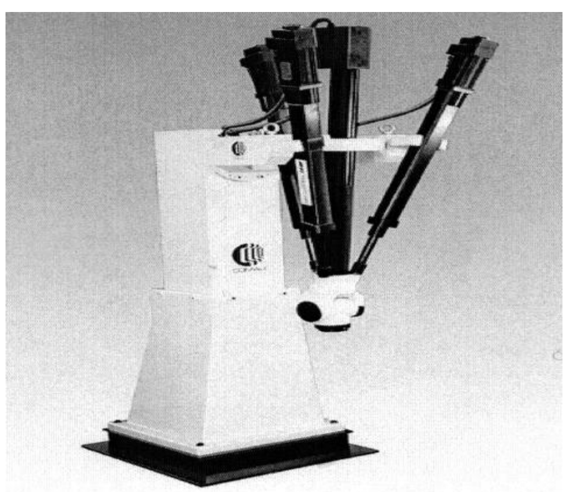


Figure 3.1 Tricept Robot [21]

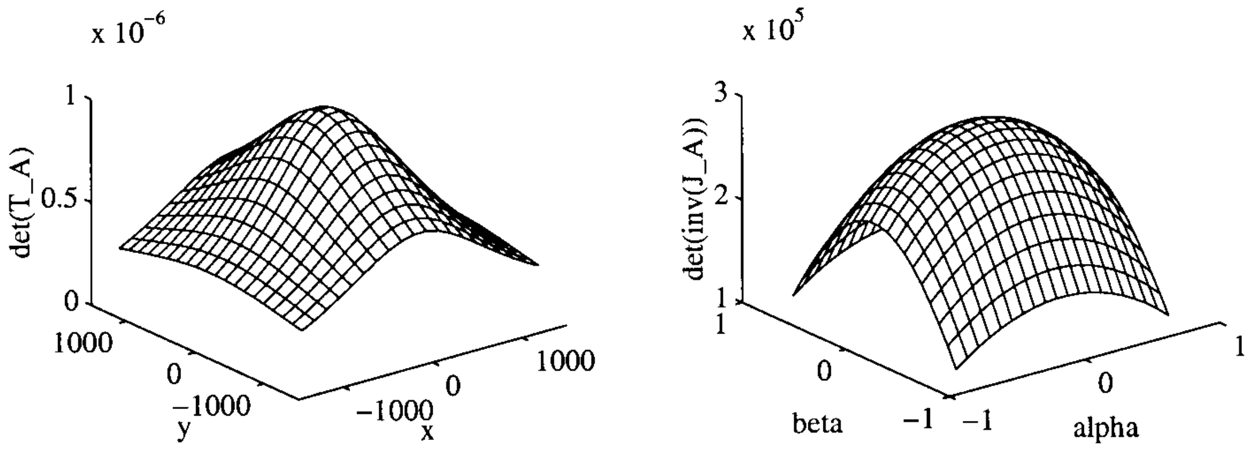


Figure 3.2 workspace measurement 1) left side: determinant (T_A) versus the x and y movement 2) right side: determinant inverse of (T_A) versus the beta and alpha.

[12] Elaborates in its topic 9.5 and case study of the Tricept manipulators in which he has used the 2 translations along y and z and also one rotation along y. He had formulated the kinematic solutions technique which later on I will be using for my tricept robot. It is closely related to this piece of work.

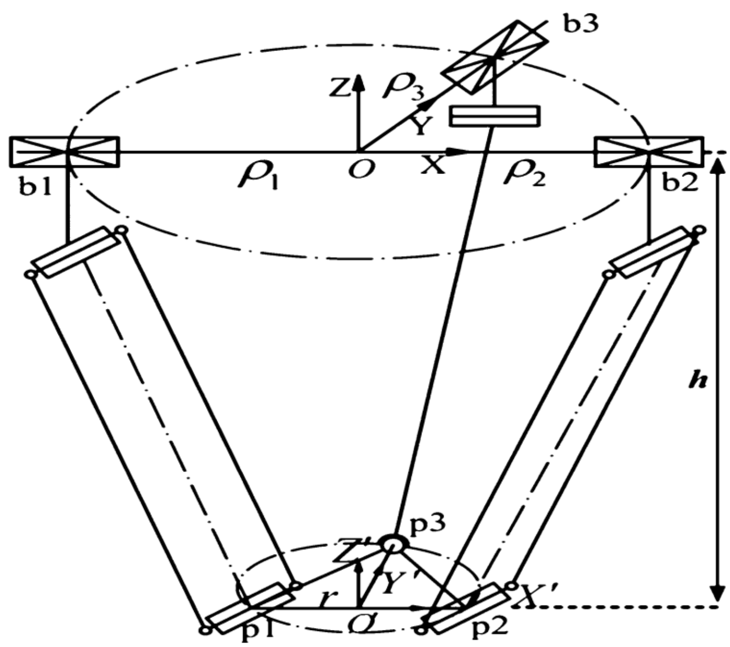


Figure 3.3 Tricept structure [12]

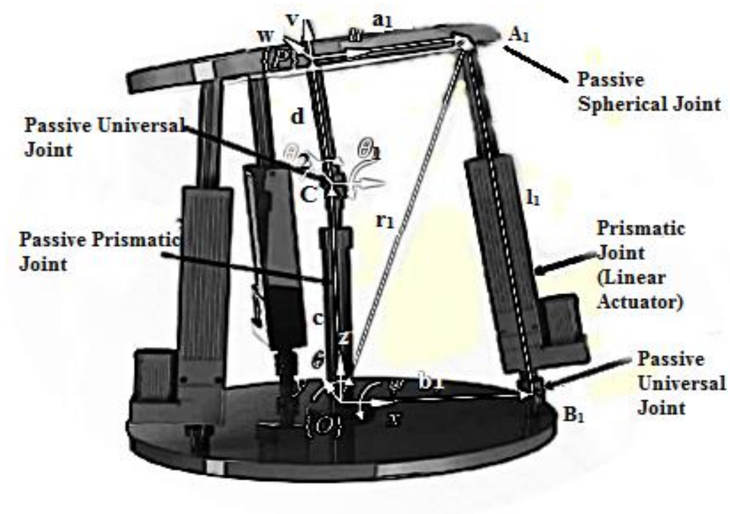
He had used the mean and standard deviation techniques to measure the workspace and by stiffness and condition number to enhance the quality and uniform graph of the workspace.

3.3 Bench Mark Paper and Its Description

Bench mark paper which is the "dexterous workspace optimization of tricept parallel manipulator" written by Mir Amin Hosseini and Hamid-reza M.Daniali in 2011. Hosseini [22]. He has explained the design of a tricept parallel manipulator with his weighing techniques, kinematic equations jacobians and Performance parameters to control the regularity is inverse condition number and the workspace optimization. He has used the genetic algorithm technique for its optimization and also has explained the concept of MSVs.

3.4 Tricept Mechanism

He has used the 3 DOf scheme and architectural view is explained under figure below.



Now let spend some time to describe this tricept mechanism. This mechanism has 3 DOF and combination of joints include the two rotations and one translation. Actuated joint is prismatic and it has SPS configuration but later on one spherical has been replaced by the Universal joint so then it became the UPS structure. The center link connects the base to the moving platform. When the structure is static the line passing through the universal joint of the moving platform is parallel with the x and y axis of base. When the prismatic joint is activated other universal and spherical joints are passive with that prismatic joint movement.

The procedure which this paper is following is the weighing factor and analytical method. Later on numerical search genetic algorithm is used for its workspace optimization. All dexterity concepts have been used in chapter no 2. The methodology will be discussed next to reach towards the optimization.

3.41 Kinematic Solutions and Jacobian

The main formulation of the parallel manipulator will be discussed in this section. In order to go toward the calculation of the performance parameters like conditioning index and global index, first find the inverse kinematics of the whole structure. Here are some steps to calculate the inverse kinematics and then to conditioning index which will finally to the global index.

1. Formulate position vectors of limbs with respect to frame 'O' which is base frame.

   OB1, OB2, OB3

2. Formulate position vectors of limbs with respect to 'P' frame which is the moving frame.

   PA1, PA2, PA3

3. Considering center link and make the rotation and translational matrices.
4. Then from closed loop procedure, find the position vector indicating from base to moving platform.

$$A_i = Q_O^P * PA_i + OP \qquad (3.1)$$

A_i=Transformation from base point 'O' to the moving 'P', i ranges from 1 to 3

Q_O^P =Rotation matrix from point of base to moving platform

OP= Position vector from base to moving platform.

5. Then from the constraint equations, proceed towards the inverse kinematics.

$$\| (A_i - B_i) \| = q_i \qquad (3.2)$$

   Where i approaches from 1 to 3,

   {q1, q2, q3} denotes the actuated lengths of joints configuration and $\{\varphi, \theta, c\}$ is the Cartesian coordinates. Where φ denotes the rotation angle along x axis and θdenotes the rotation angle along the y axis whereas c is the translation along z axis.

6. In this step first take the differentials on both side of equation 3.2, then finally rearrange the above 3.2 equation into 2.6 form where separate the inverse and forward kinematics matrices.

$$J_x \dot{x} = J_q \dot{q}$$

7. From this step onwards, way towards the conditioning index which is the first performance index is clear So by using the equation 2.10, conditioning number which by taking the inverse of K gives ‘k’ which is the conditioning index.

$$K= ||J||*||J^{-1}|$$

$$k=1/K$$

8. Further check some results globally by using the global indexing performance index as stated above in equation 2.11.

$$A=\int_{\varphi} \int_{y} \int_{x} (\frac{1}{K})\, dx\, dy d\varphi$$

Simply global index is the mean of the conditioning index in a prescribed volume around its workspace.

3.42 Procedure for the Workspace Volume

There are many methods adopted by many researchers for the calculation of workspace volumes. Analytical and numerical approaches have been introduced previously in [30]. Same concept has been used here. Firstly It takes the whole of the workspace as a cube which have three axis x, y and z respectively then it takes the subspace, a cylinder in particular for the workspace calculation. It restricts the legs and the platforms of the manipulator around a cylinder and from the inverse kinematic solutions of the parallel manipulator. By keeping in view the constraints, it searches each q’s in that subspace which forms the closed cylinder. After each z increasing, it tries to find out the solutions which are trapped inside or onto the surface of that subspace. [30]

Step 1: For a certain z, find the inverse kinematics solutions for a prescribed set of parameters and their Design and Geometric constraints.

Step 2: Do the necessary procedure to make a sampled hollow cylinder as a subspace.

Step 3: Start a check for the point of solutions to be in that cylinder and discard the remaining set of points as it is beyond our boundary conditions.

Step 4: Repeat the procedure from step 1 to 3 for z=z+1.

Step 5: Get a set for all z and save it a column matrix form to be used later on for analysis. Plot it to get a desired reachable workspace around a subspace.

Next example will be shown of the subspace created. Note that it is increasing along the z. more illustration of workspace will be stated later in chapter 4.

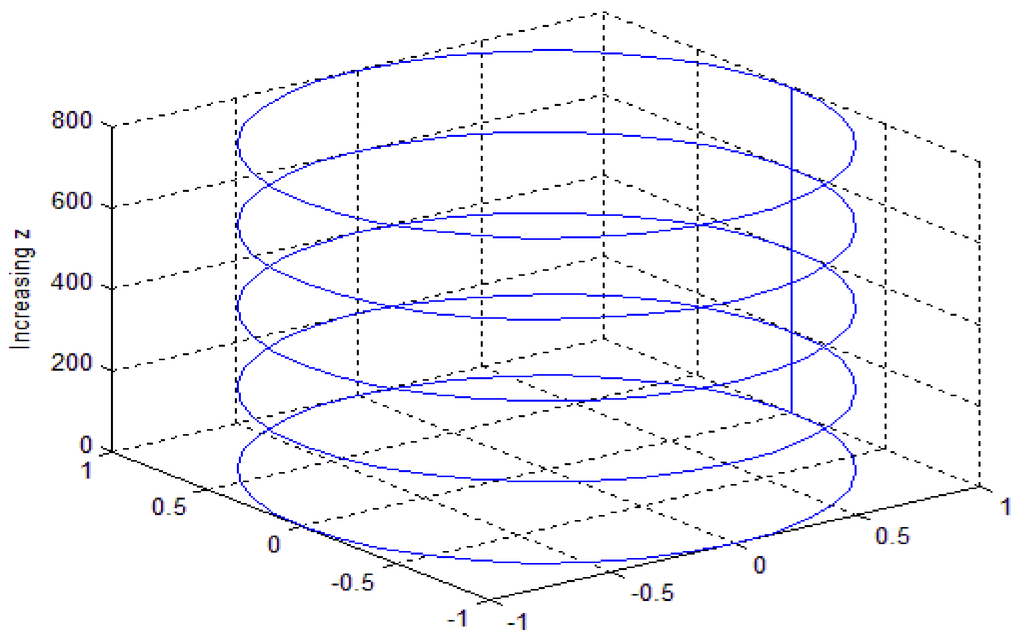


Figure 3.4 Illustration of the cylindrical subspace, several circular patterns are generated with increasing z.

3.43 Single Objective Optimization:

Single objective deals with the optimization of parameters independently. Like it undergo for optimization process of one parameter irrespective of other performance parameters. All the techniques whether numerical or analytical will apply for the optimization process here. Evolutionary algorithms (EA's) will be discussed next for the process to get optimized design variables finally. Here are some points for better understanding to use the evolutionary algorithms for our task. [31]

1. When found the uncertainties in solutions.
2. When there are random design variables involved.
3. Much complex constraints in their calculations.
4. If there are more numbers of local and global optimum points.

These are some reasons that usage of EA's has been increasing by many researchers for their convergence and computations. These algorithms perform swiftly for the findings of local and global minima and maximal points. Other traditional methods like bracketing and elimination optimization techniques does not guarantee findings of optimum points. They can skip their local and global points.

So in order to have that EV’s in our parallel manipulator calculations, some types are given below.

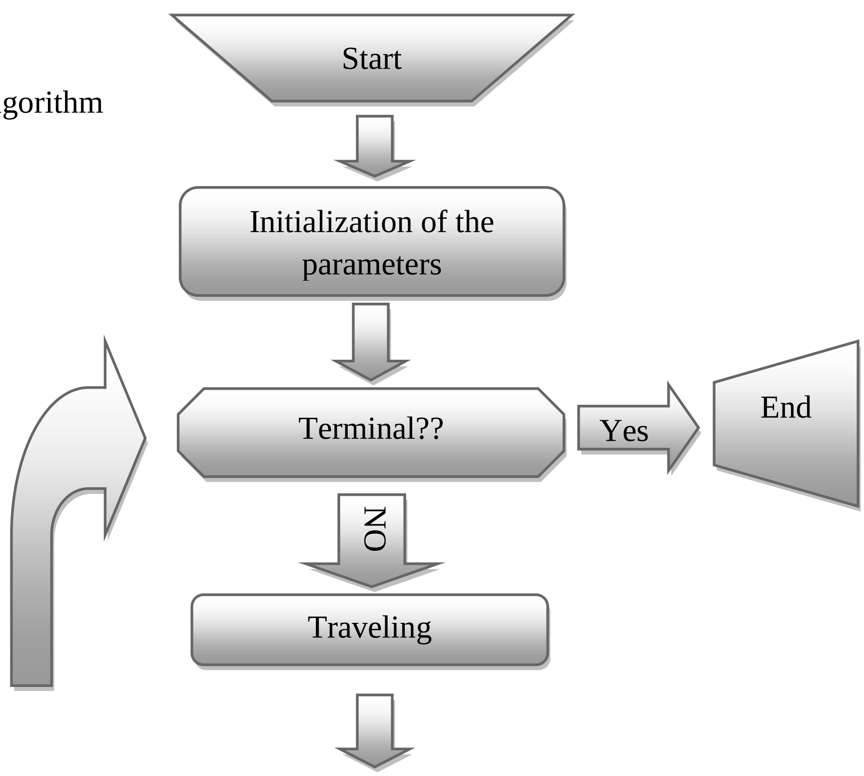
1. Ant colony Optimization
2. Genetic algorithms optimization.
3. Particle swarm optimization

So wide range of options are open to select any optimization technique and apply the data set according to user needs. Small overview of the above optimization techniques is as follows. Small overview of the above optimization techniques is as follows.

3.431 Ant Colony Optimization (ACO)

Ant colony uses the combinatorial optimization scheme for its task of finding the optimum points. It is a kind of Meta heuristic search. Motivation behind that study is a social behavior of insects and ant. This technique addresses the fact that ant always uses the shortest path to achieve certain food. Every ant follow that pheromone that is produced by the successor ant which has been passed from there some time before, leaving pheromone a sign to achieve that shortest path [32]. Traveling salesman is a latest application to this algorithm [32]. Basic flow chart has been given below. [33]

Figure 3.5 ACO Algorithm



3.432 Genetic Algorithms (GA's)

Genetic algorithms are famous for its task of converging results possibly to one global optimum. It uses the fact of human generations from the process of genes formation and new child population produces and so on. Formation of chromosome from genes includes numerical, binary or integer values. These chromosomes will recombine with other chromosomes from within the whole population through the process of cross over. Later on mutation occurs and offspring's generated will form a new set of population comprised of new chromosomes. The selection of a new chromosome will based on its fitness value. One which has higher fitness value will be selected as fixed for the new generation and is stated from the Darwinian evolution rule .After some iterations data will tend to converge to one global max or minima point according to the user defined criteria and the given geometrical constraints. Flow chart will better give an overview of this genetic algorithms. [34]

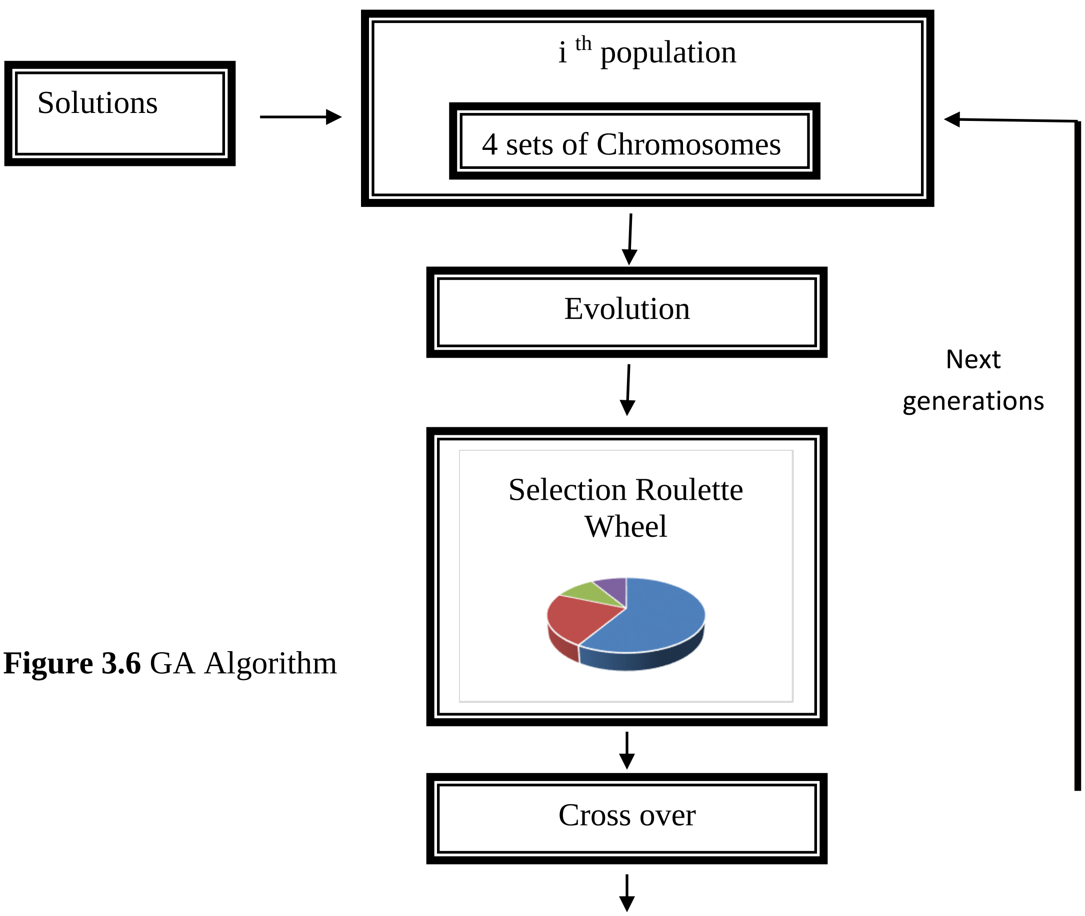
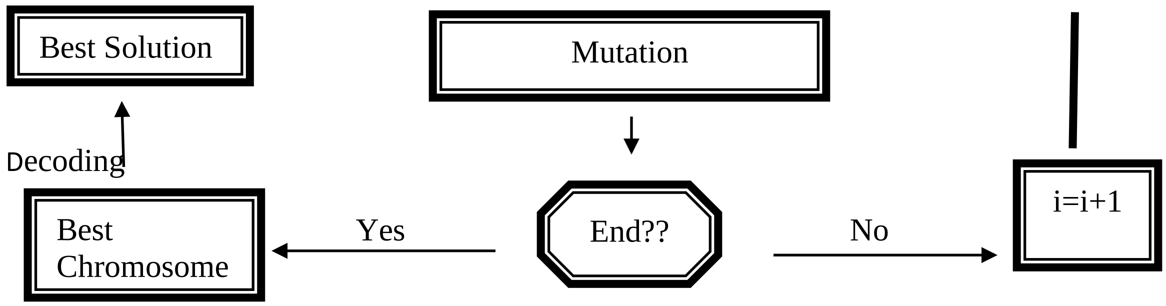


Figure 3.6 GA Algorithm



3.433 Particle Swarm Optimization (PSO)

Our main focus will be on particle swarm optimization and it will be dealing with this technique throughout our optimization. It is the social behavior of birds which this algorithm follows. When the birds move in search of food and all don't know the exact location of food. Finally the food is located by one bird and it is found to be nearest so now all the birds will follow that food which has been found by one of their bird. [34] That bird can be name as a leader. So PSO in fact is searching algorithm which usually take less iterations for their optimum points as compared to GA's. Both search in the same manner but the PSO takes one way for its results as all the other birds try to adopt the same velocity update with the leader one while the GA's have to go through the chromosome cross over and mutation processes and information has to be shared by the every group member in the population and whole group will go through the finding process as one. Both algorithms GA and PSO start with the same process of initialization. [35]

Some steps are described here below for better understanding of the algorithm.

1. Initialization of the population of design variables takes place randomly at first with range of variables in consideration.
   e.g. I can take a population set of 5x2 order. Such that we have 5 rows and 2 columns for the matrix. This means that we have a set of two particles and 5 swarm size taken as 'n'
2. Take the random velocities likewise the step 1. Remember the order of matrix which should remain same like the initialization of the population.
3. Calculate the fitness value by putting the design variables into the objective function.

4. Define the criteria whether the user want to maximize the objective function or minimize. E.g if I want to take the maxima optimum point. Check all the fitness value points and then take that value which is maximum of all.
5. After taking the highest value, put a check to examine whether that value is within the range or constraints of that maximum values. If yes then the corresponding value of the swarm will be taken as gbest for that swarm. Corresponding to that gbest value with the initial design variables taken at start, that particle chosen will be considered as pbest of that swarm. Similarly take the velocities corresponding to that pbest values.
6. In this step the velocity update procedure starts. Taking the length of the swarm to be i. start the update from the below update equation.[36]

$$vnew = vg(j)+c1*r1*(pbest-x(i,j))+c2*r2*(gbest(j)-x(i,j)) \tag{3.3}$$

Whereas

vnew = New velocity after update

vg = Global velocity of the particle

pbest = Particle best is same at start as x(i,j).

x(i,j) = It is the value of particle taken from i^{th} row and j^{th} column from the start to size of the swarm ‘n’

gbest = global best is the global best of the swarm corresponding to the fitness value of the objective function.

c1 = first constant

c2 = second constant

r1 = first random value

r2 = second random value

These c1 and c2 are the constants and weights assigned to each particle during its updating and usually these constants should both sum up to 4 in simulations whereas r 1 and r2 both are random values taken 0 to 1.

7. Similarly the position update of the particle takes place in accordance with the velocity update equation which represent in this form normally.[36]

$$xnew = x(i,j)+ vnew \tag{3.4}$$

xnew is the new position of the particle.

Here it is necessary to perform 2 checks to accept this xnew. First check it is in the limits taken as start and secondly check for maximum value whether xnew is more than the previous value. If yes accept that value otherwise this position will be replaced by the old x value. Go through the whole swarm and update accordingly and save the new updated swarm as a new population.

8. Proceed towards the new iteration and then go to 3 and perform the same process afterwards to 7. Make new gbest and new pbest if it is more than previous gbest and pbest for maximizing criteria. In this way we get the maximum swarm values in the end which will be converging and hence declare that value as maxima optimum point. Flow chart for easy understanding of this algorithm is as follows. [37]

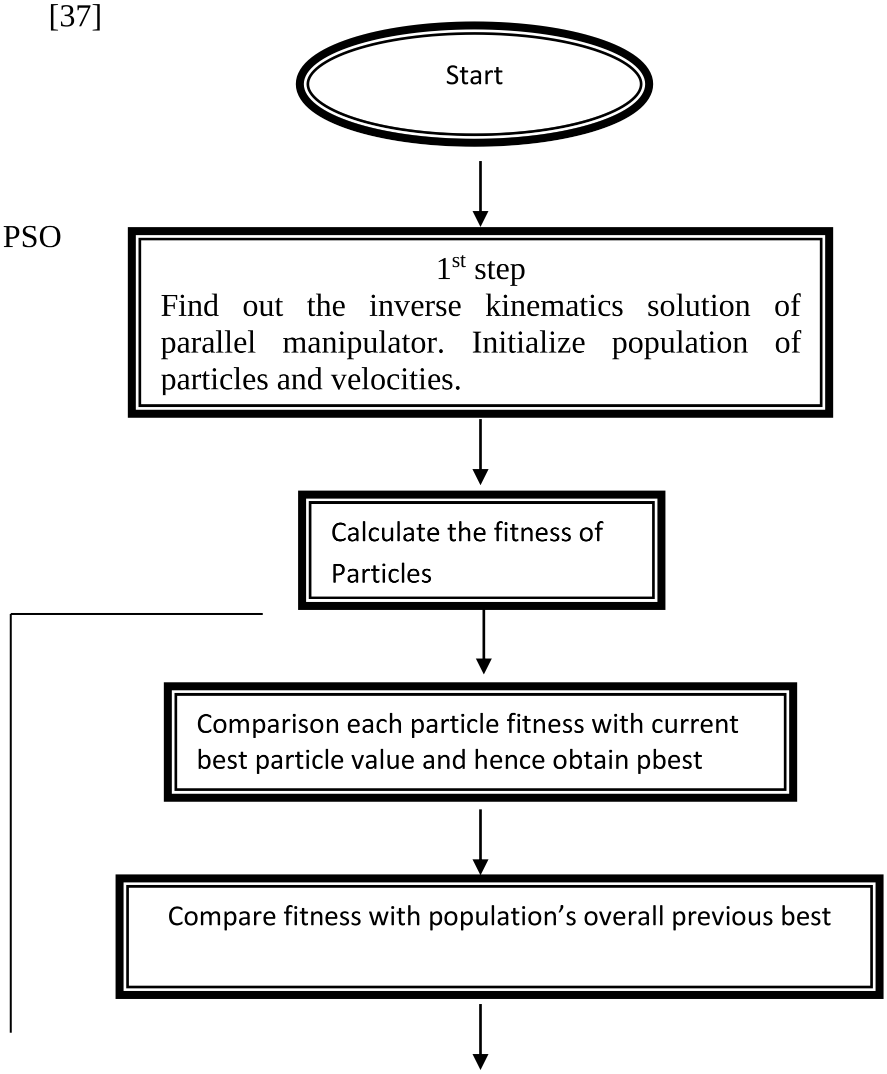
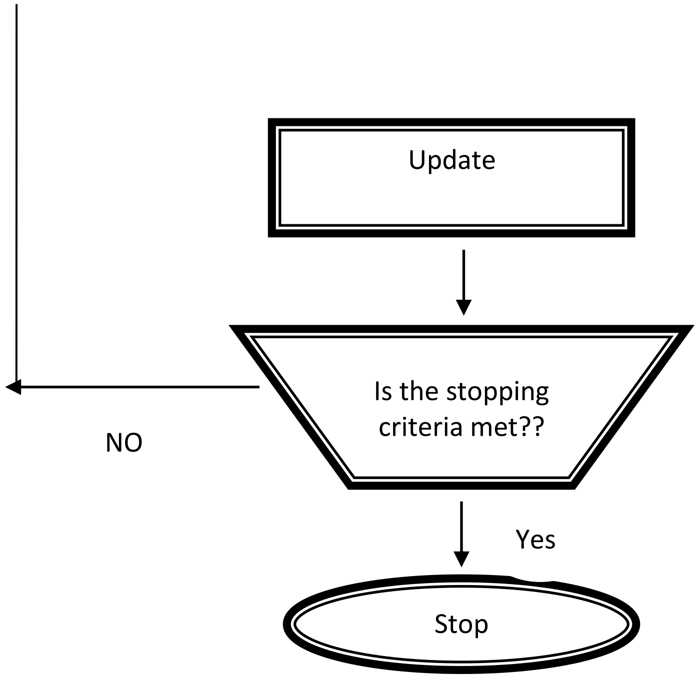


Figure 3.7 PSO



3.44 Multi objective Optimization

It deals with the multiple performance parameters to optimize simultaneously. In this optimization we will assign weights in order to perform one full or any ratio with cost of other parameters. We have to do because we have to check the relationships of parameters with each other.[38] When we take 2 to 3 parameters at same time. Their results usually conflicts so we will constantly evaluating their simulations and then we introduce the constraints like geometric and design constraints and finally try to find out the possibly optimum point from the data set of points. Optimum point will either be local or global maxima or vice versa [38] .and here it should be noted that it is using here the particle swarm technique for the multi objective optimization and shortly is termed as MOPSO.

3.441 Multi Objective Particle Swarm Optimization (MOPSO)

MOPSO Uses the technique of PSO stated previously in section 3.433. Here in multi objective make one more function which will have the all the performance parameters to be in function so that it has a variation of all the parameters at the same time through one function. Now new function will be treated as objective function for the job and the performance parameters act as function variables. After that all the process for the optimization remains the same. [39] There are many methods to form that new function. Two methods are discussed here for our calculation.

1. **Weighted Sum method**

   In this multi objective technique the function is formed by assigning the weights. Each variable has assigned weight which can be changeable according to the user demands. Check the simulation results by altering the weights [38]. E.g. in this study of work it has been used the two independent performance parameters. And the equation will be like the following. [38] Maximize y now and we will examine through this function, the fitness value.

$$y=w1*z\ (i,1)+w2*z\ (i,2) \tag{3.5}$$

   w1 and w2 are the two weights assigned to the conditioning index and the workspace volume. This method is good for continuous and convex problems however local optima usually achieve to discontinuous functions as well. [38]

2. **Epsilon constraint method**

   This method uses the one single function and restricts other functions through some constraints. Hence give as overall optimum desired results under that set of constraints. The user have to play through these set of constraints very carefully and need expertise knowledge about the boundaries of the solution. [38]

   Maximize y with respect to the other constraints for example, I just show 2 function as a constraints here C and L.

$$\begin{aligned} &\text{Function:} \quad Y=F_i\,(X) && i=1,2,\ldots\ldots,I \\ &\text{Constraints:}\ C_j(X) && j=1,2,\ldots\ldots,J \\ &\qquad\qquad\quad L_k(X) && k=1,2,\ldots\ldots,K \end{aligned} \tag{3.6}$$

3.5 Thesis Methodology Layout

The whole scheme in brief will be illustrated below in the form of a flow chart. The methodology and the sequential graph for our work. The detail has been previously mentioned in this chapter 3.

1^{st} step

Find out the inverse kinematics solution of parallel manipulator.

↓

2^{nd} step

Find workspace volume, conditioning index and global indexes.

↓

3^{rd} step

For a given set of data in an organized form, apply the PSO algorithm.

↓

4^{th} step

Single objective optimization occurs at each PSO on all the performance parameters.

↓

5^{th} step

Once we completed results on single objective go for multi objective. Implement Weighted PSO by introducing its new weighted objective function.

↓

6^{th} step

Apply now the epsilon constraint method for multi objective optimization and get the results.

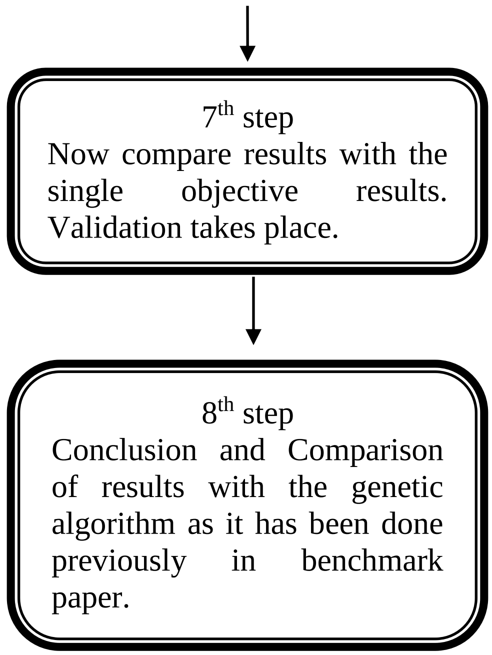


Figure 3.8 Proposed Methodology for the thesis

CHAPTER 4: APPLICATION AND RESULTS

In this chapter the simulation and results are evaluated. From the above methodology it follow the steps to show the results. Kinematic solutions will be described later. Table describe the geometric constraints.

Actuator lengths(mm)	**Angle (rad)**	**d (mm)**	**b(mm)**	**a(mm)**
400-750	-1 to +1	20-200	300-500	200-300

Table 4-1 Geometric constraints

Where d is the length of the joint from C to P point respectively and also 'b' is the length of the static platform from point O to B_1 whereas 'a' is the length from point P to A_1 of the moving platform.

4.1 Inverse Kinematics

Here the main objective is to go to the inverse kinematics equation, finding a relation between the Cartesian and joint coordinate system. Keeping in mind the actuator lengths (q's) joint coordinates and the passive x which is the Cartesian coordinates, we drive the jacobian equation separating these two terms. Recall those steps which topic 3.41 covers the tricept mechanism has two rotations which are along the x axis and y axis simultaneously so by applying the right hand rule it can show these rotations as in equation 1.3 and 1.4. Taking the clockwise positive and the right hand rule, we have the equations as below.

$$Rx = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\psi & S\psi \\ 0 & -S\psi & C\psi \end{bmatrix}$$

$$Ry = \begin{bmatrix} C\theta & 0 & -S\theta \\ 0 & 1 & 0 \\ S\theta & 0 & C\theta \end{bmatrix}$$

Let me now define the a_1, a_2 and a_3 which is in fact the PA_1, PA_2 and PA_3 respectively.

$$a_1=\begin{bmatrix} a/\sqrt{3} \\ 0 \\ d \end{bmatrix}, \quad a_2=\begin{bmatrix} -a/(2\sqrt{3}) \\ a/2 \\ d \end{bmatrix}, \quad a_3=\begin{bmatrix} -a/(2\sqrt{3}) \\ -a/2 \\ d \end{bmatrix} \tag{4.1}$$

Then finally according to the equation 3.1

$$A_i=Q_O^P * PA_i + OP$$

$$A=\begin{bmatrix} C\theta & S\psi S\theta & C\psi S\theta \\ 0 & C\psi & -S\psi \\ -S\theta & C\theta S\psi & C\theta C\psi \end{bmatrix}\begin{bmatrix} a/\sqrt{3} & -a/(2*\sqrt{3}) & -a/(2*\sqrt{3}) \\ 0 & a/2 & -a/2 \\ d & d & d \end{bmatrix}+\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ c & c & c \end{bmatrix} \tag{4.2}$$

Similarly

$$B=\begin{bmatrix} b/\sqrt{3} & -b/(2\sqrt{3}) & -b/(2\sqrt{3}) \\ 0 & b/2 & -b/2 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.3}$$

Where B is the position vector of B with respect of O.

According to the constraint equation 3.2.

$$\| (A_i - B_i) \| = q_i$$

Where || denotes the Euclidean norm.

$$q_1^2= \frac{a^2}{3} + \frac{b^2}{3} + c^2 + d^2 - \frac{2}{3} ab\, C\theta + 2cdC\theta C\psi - \frac{2bd}{\sqrt{3}} C\psi S\theta \tag{4.4}$$

$$q_2^2= \frac{a^2}{3} + \frac{b^2}{3} + c^2 + d^2 - \frac{1}{2}ab \left(\frac{1}{3} C\theta - \frac{1}{\sqrt{3}} S\psi\, S\theta + C\psi\right) + bd \left(\frac{C\psi S\theta}{\sqrt{3}} + S\psi\right) + \tag{4.5}$$

$$2cdC\theta C\psi + ac \left(\frac{S\theta}{\sqrt{3}} + C\theta S\psi\right)$$

$$q_3^2= \frac{a^2}{3} + \frac{b^2}{3} + c^2 + d^2 - \frac{1}{2}ab \left(\frac{1}{3} C\theta + \frac{1}{\sqrt{3}} S\psi\, S\theta + C\psi\right) + bd \left(\frac{C\psi S\theta}{\sqrt{3}} - S\psi\right) + \tag{4.6}$$

$$2cdC\theta C\psi + ac \left(\frac{S\theta}{\sqrt{3}} - C\theta S\psi\right)$$

Now up to here we have that three set of actuator lengths. Next section is of jacobian formation.

4.2 Jacobian Formulation and Conditioning number

Differentiation is the technique through which it will proceed towards the jacobian formulation.by taking differentiation on both sides of equation 4.4, 4.5, 4.6. As from the equation

$$J\dot{x} = \dot{q}$$

It came to know that jacobian is basically the relation to go to from the Cartesian (x) to the joint coordinate system (q). Note that the J is already in inverse jacobian form so the above relation in detail is elaborated as follows.

$$\dot{q} = \begin{bmatrix} \dot{q_1} \\ \dot{q_2} \\ \dot{q_3} \end{bmatrix}, \qquad \dot{x} = \begin{bmatrix} \dot{c} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{4.7}$$

Now after taking derivatives on both side of the above equation we separated $\dot{x}$ and $\dot{q}$. Let suppose three matrices.

$$P = \begin{bmatrix} 1 + dC\theta C\psi \\ 1 + dC\theta\psi + \frac{S\theta}{2\sqrt{3}} + C\theta S\psi \\ 1 + dC\theta C\psi + \frac{S\theta}{2\sqrt{3}} - C\theta S\psi \end{bmatrix}, \qquad Q = \begin{bmatrix} \frac{abS\theta}{3} - cdS\theta C\psi - \frac{2bdC\psi C\theta}{\sqrt{3}} \\ \frac{abS\theta}{12} + \frac{S\psi C\theta}{4\sqrt{3}} + \frac{bdC\psi C\theta}{2\sqrt{3}} - cdS\theta C\psi + \frac{acC\theta}{2\sqrt{3}} - \frac{acS\theta S\psi}{2} \\ \frac{abS\theta}{12} - \frac{S\psi C\theta}{4\sqrt{3}} + \frac{bdC\psi C\theta}{2\sqrt{3}} - cdS\theta C\psi + \frac{acC\theta}{2\sqrt{3}} + \frac{acS\theta S\psi}{2} \end{bmatrix}$$

$$R = \begin{bmatrix} -cdS\psi C\theta - \frac{2bdS\psi S\theta}{\sqrt{3}} \\ \frac{-C\psi S\theta}{2\sqrt{3}} + \frac{S\psi}{2\sqrt{3}} - \frac{bdS\psi S\theta}{2\sqrt{3}} + \frac{bdC\psi}{2} - cdS\psi C\theta + \frac{acC\theta C\psi}{2} \\ \frac{-C\psi S\theta}{2\sqrt{3}} + \frac{S\psi}{2\sqrt{3}} - \frac{bdS\psi S\theta}{2\sqrt{3}} - \frac{bdC\psi}{2} - cdS\psi C\theta - \frac{acC\theta C\psi}{2} \end{bmatrix}$$

Now combining P, Q and R. jacobian matrix is as follows:

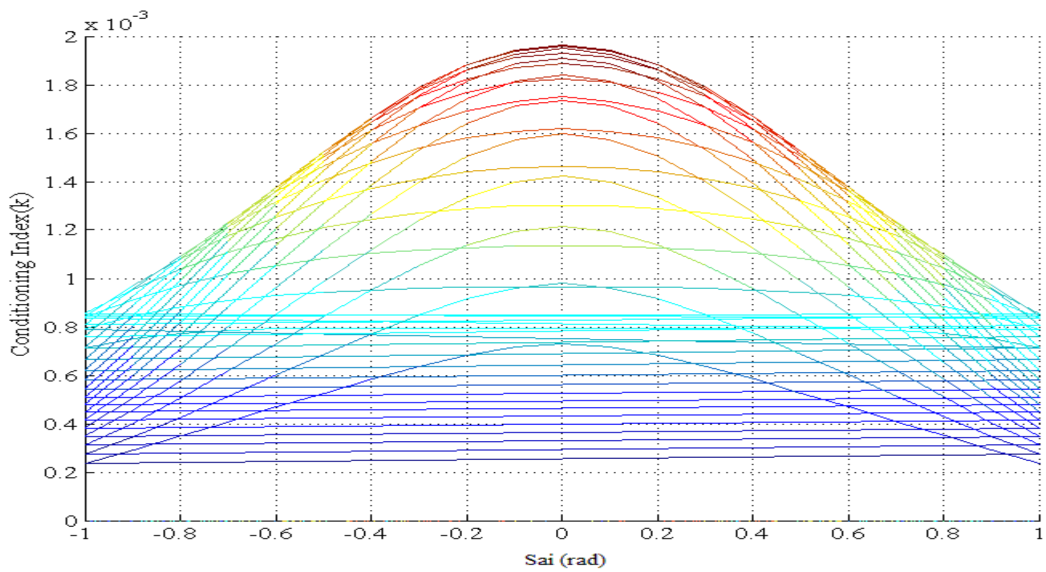
$$J = [P\ Q\ R] \tag{4.8}$$

The jacobian has been found and now there is need to introduce the first performance index into calculation that is the conditioning index from equation 2.9

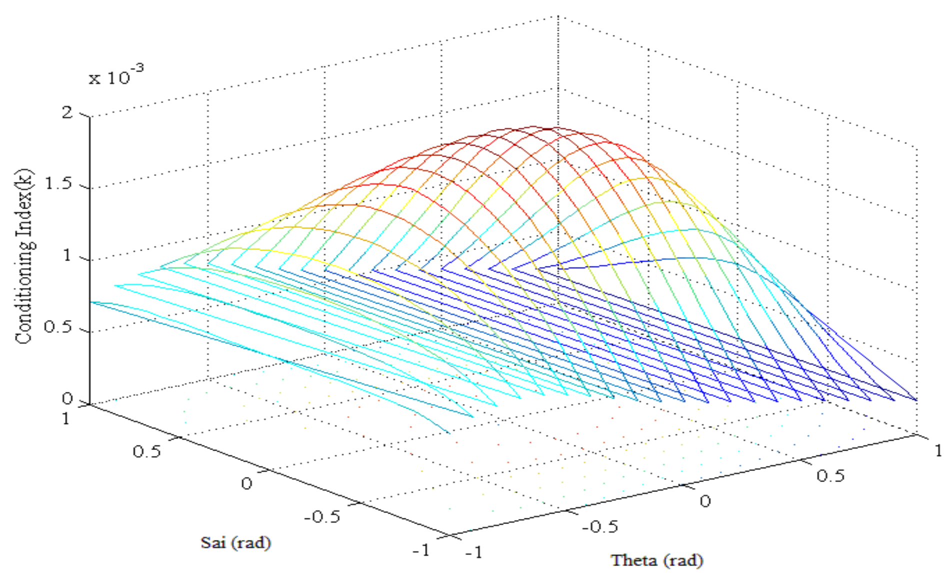
$$K(J) = \frac{\sigma max(J)}{\sigma min(J)}$$

Then k=1/K denotes the conditioning index.

Here it has taken only one elevation in the direction of ‘z’ for the plot of conditioning number against the given set of geometric constraints. Matlab code has been generated and illustrated in Apendix1. Iterations have been taken with a step size of 0.1 between -1 to 1.



(4.1a)



(4.1b)

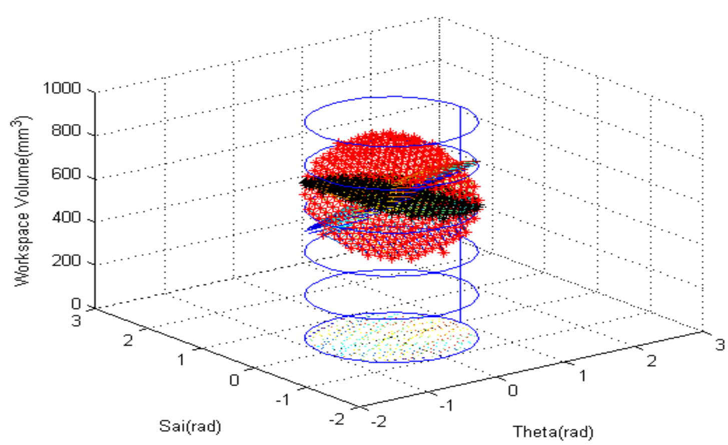
Figure 4.1 Conditioning index versus different orientation ‘theta’ and ‘sai’ at elevation of ‘500’mm (a) 2d view (b) 3d view

It is shown from the graph that the peak point of the curve is at $1.9e10^{-3}$. The step size taken between the set of constraints is 20.

Recall from the previous concepts of dexterous and the dexterity. Dexterity is actually the measure of sensitivity between the end effector and the actuator movement. Dexterity put a significant importance in design and control. Singularities are also there to control its movement around the prescribed task space. Larger singular values denotes the position and resolution control of the end effector while the smaller singular value is useful for the end effector velocities and hence it concludes that larger singular values guarantee accuracy and stiffness while smaller seems to get same outputs for relatively lower values which is important thing to cater for dexterity. Conditioning number just provides this sensitivity ratio for dexterity.

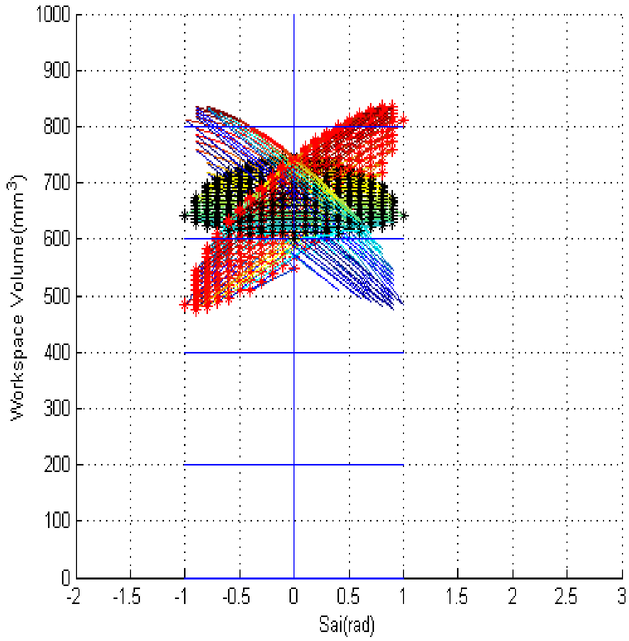
4.3 Workspace Volume

From topic 3.42 follow the steps that involve the calculation of the workspace volume. Considering the same set of geometric constraints of table 4.1 for volume analysis. Creating the subspace in the shape of cylinder. The step size taken between the set of geometric constraints is 20. In this algorithm it will search out the points that lies within the taken cylindrical subspace and discard the remaining set of points. 1123.2 mm^3 is the volume when only taking the values of inverse kinematic solutions q's under the subspace it has taken along z ranges from 0 to 1000 mm. step size of angles is 0.1 rad. Step size for geometric constraints is 10mm.



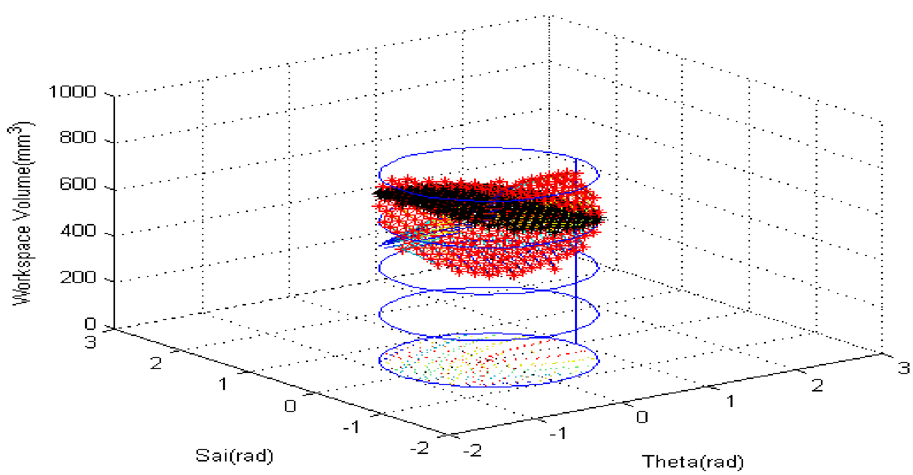
(4.2a)

Figure 4.2 a) Shows the 3d view of workspace volume without actuator limits in the subspace. **b)** Shows the 2d view of the workspace volume without actuator limits in the subspace.



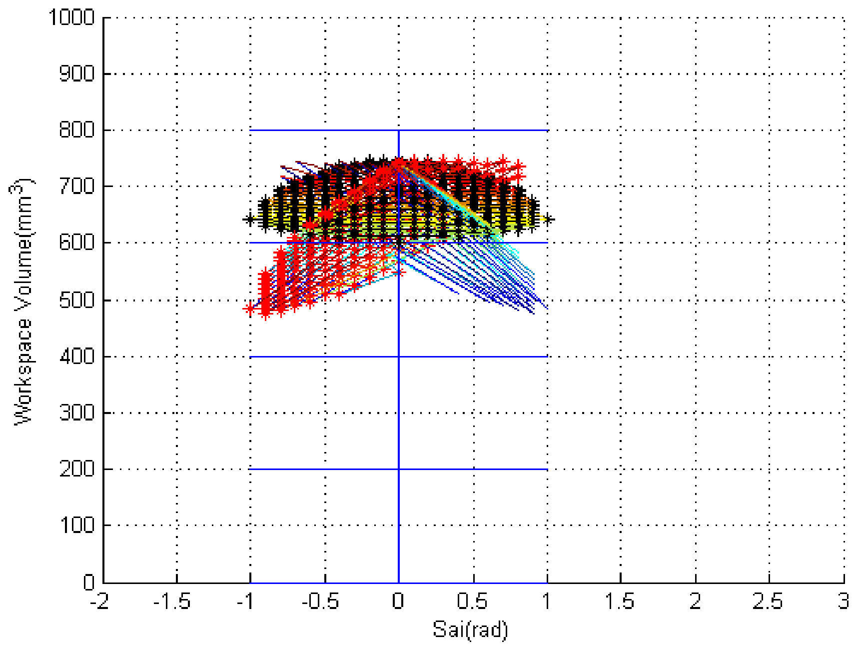
(4.2b)

It is obvious now that the region will be eliminated in the workspace calculation when keeping in view the actuator lengths ranging from 400 to 750mm and in the subspace. The results are as follows. Note that these both with or without constraints have been taken when it has the step size of geometric constraints equal to 10. Z ranges from 0 to 800 in this case. It may change according to your requirement. After considering the actuator constraints volume is found to be 845.2571mm$^{3.}$



(4.3a)

Figure 4.3 a) Shows the 3d view of workspace volume with actuator limits of 400 to 750mm in the subspace **b)** shows the 2d view of the workspace volume with actuator limits in the subspace.



(4.3b)

Next step is to evaluate the best parameters and optimum values using the evolutionary algorithms

4.4 Optimization

In optimization it has studied different techniques in topic 3.43. In this portion try to evaluate the graphs and design parameters by optimizing through the PSO algorithm. Firstly examine the single objective solutions and then proceed towards the multi objective optimization.

4.41 Single Objective

The conditioning index is being optimized for the set of design variables a, b and d and PSO algorithm is launched. Aim to find minimum point for this performance index was accomplished and the corresponding design variables saved against that best minimum point. The execution of the Matlab code reveals the results in the appendix 2 as follows.

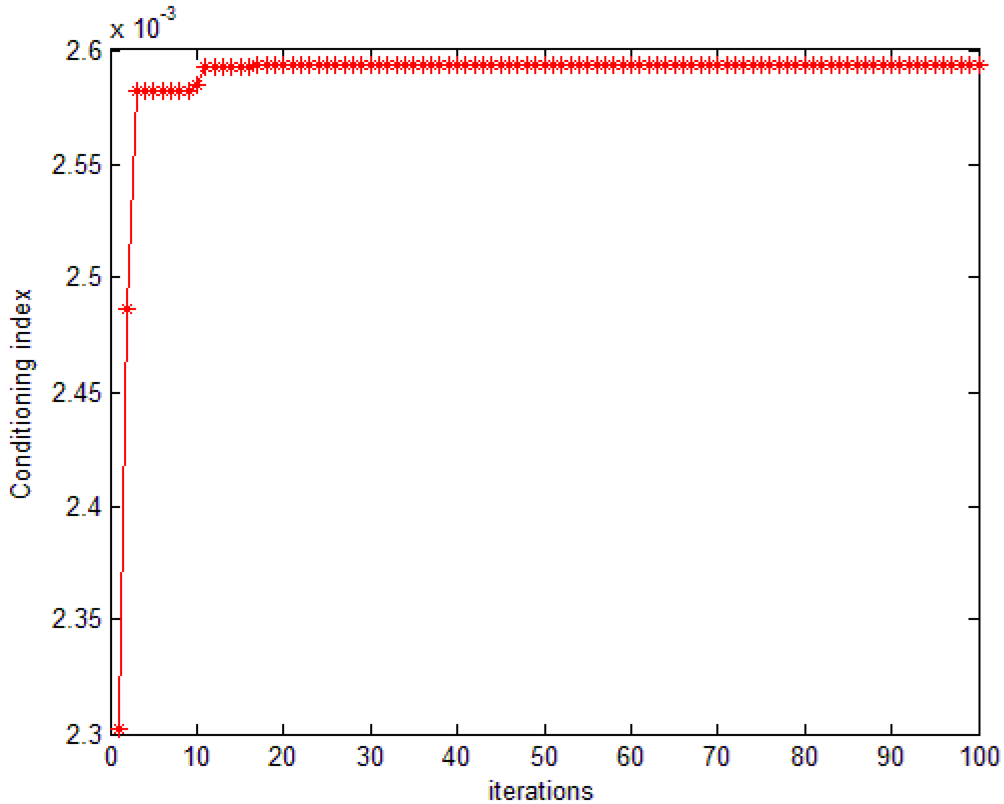


Figure 4.4 Condition number versus the iterations

The figure 4.4 result is demonstrated against the iterations of 100. For 20 intervals between the design variables the iteration started until a smooth constant line comes as it is a sign that the algorithm has found its most probably the optimum point. Iterations are being used here as a stopping criterion. It can be seen that graph has reached its maximum optimum value in near about 20 iterations. As graph showing the nearby optimum condition index point is $2.593x10^{-3.}$

1	a(mm)	b(mm)	d(mm)	Theta	Sai	CN	CI
4822	200.0313	305.1959	98.97398	0.1	0	553.2977	0.001807
4823	200.0645	304.7332	88.68052	0.1	0	611.3445	0.001636
4824	200.0367	305.2536	155.9036	0.1	0	385.642	0.002593
4825	200.0068	305.2877	155.0147	0.1	0	387.1652	0.002583
4826	200.0301	305.0755	53.63707	0.1	0	1014.79	0.000985

Table 4-2 gbest versus design variables

Now this minimum searching conditioning index showed the corresponding shaded values of the design variables in table4.2. Global conditioning index is just the mean of the values of conditioning index.

Now check for the maximum workspace values. Firstly run the algorithm and then check for the maximum values of the workspace that it computes. PSO algorithm compiles the searching in the form of code in appendix 3 and results are shown below.

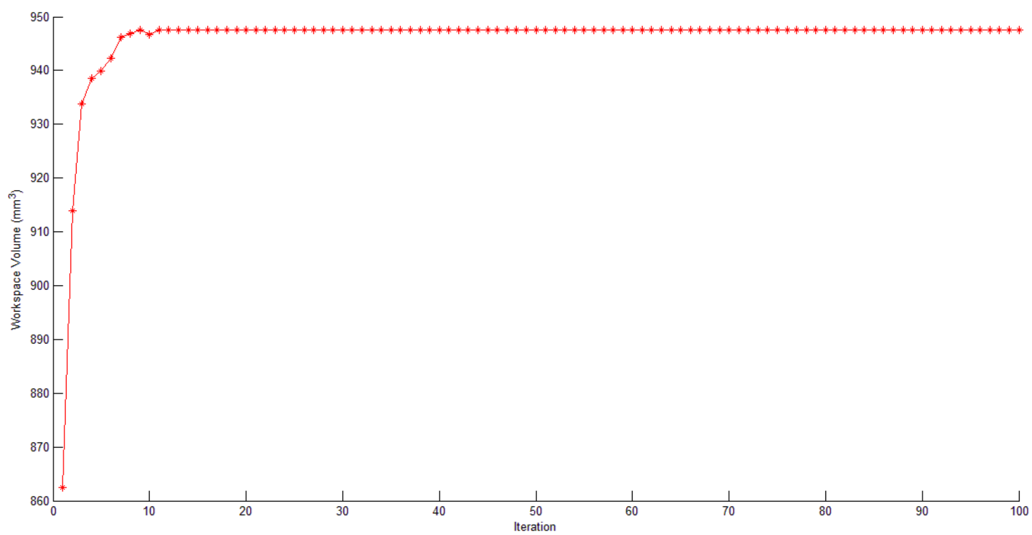


Figure 4.5 Maximum workspace values versus the iterations

The volume shown in the graph is against the number of iterations. As we have taken 100 iterations for the 20 intervals between the design variables the curve shown has made its own threshold under 20 iterations. We rely on this optimum point and regard as local maxima. PSO usually gives some different results of the optimum point at different tries. So sometimes it can expect that curve decrease to certain height and can end to lesser results than the previous one so it might give a wrong interpretation of the optimum point in prescribed number of iterations. After examine we came to know that the curve has reached its maximum at 947 volume but optimum has reached to its 947 cubic millimeters volume under 20 iterations.

1	a(mm)	b(mm)	d(mm)	Theta	Sai	Vol
3341	299.2915	499.9664	195.7733	-0.3	0.9	825.5052
3342	299.6175	499.385	178.5557	-0.3	1	921.3464
3343	299.279	499.8784	168.7831	-0.3	1	947.4832
3344	299.4126	498.2788	192.7489	-0.3	1	913.8174
3345	299.3272	487.1211	199.9535	-0.3	1	904.094

Table 4-3 Maximum volume versus design variables

It is seen that gbest volume approached to 932.8812mm^3 and it is considered to be near maxima optimum point and corresponding design variables are considered optimum which have to be designed to get that maximum volume value.

4.42 Multi objective Optimization

At this point it has been done through the optimization of single objective calculations. There remains the need to check the relationship of one performance parameter with the other so we have to produce the results which shows multi objective results. Multi objective is slightly different as it tries to occur the events simultaneously. Or say that it has to check one's performance parameter with respect to other at that point. Actually it is dealing with the three performance parameter at the same time. Three which includes conditioning index, global index and also the workspace volume. One function having these parameter acting as objective variables are evolved through the PSO algorithm and from running the MATLAB code in appendix 3. In this multi objective it is taking into account the two strategies one is weighted and the other is constraint epsilon strategy then it will conclude on the basis of results which one is more suitable.

4.421 Weighted Sum

In this technique it will have an objective function which have 3 different weights and by compromising on one's performance index, it will get another. Preference is set in the start to which performance index we want to check on behalf of others. The following figure relates the three performance index evaluated through the weighted sum strategy. 20 iterations have been made and the objective function shows constant behavior at the end of iterations. 20 intervals also taken with 0.1 step size of angles in the code. Maximization PSO runs and we have given preference to the conditioning index in the following graph such that set 1 for conditioning index and 0 for the other two parameters as we all know that weighted sum must always be equal to 1.

- Conditioning index =1
- Workspace volume =0
- Global conditioning index =0

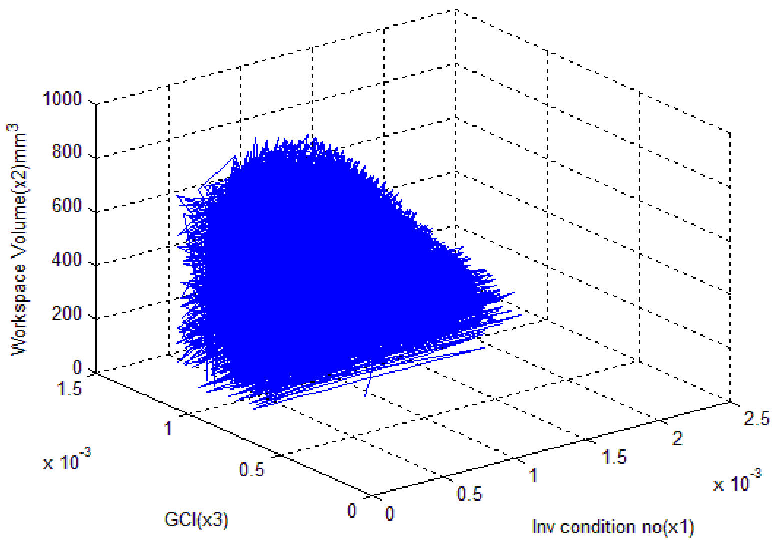


Figure 4.6 Multi objective maxima curve with conditioning number given preference of 1

It must be noted that this graph and values are taken by keeping in view all the geometric constraints. Maximizing conditioning index is .002593 through this optimization. It can be said that it is possibly the best optimum maxima point.

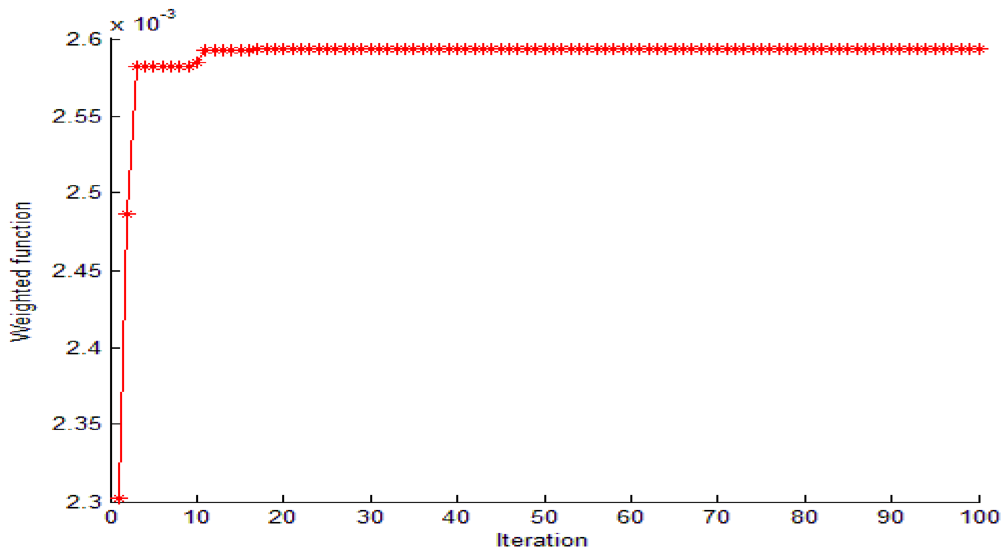


Figure 4.7 gbest conditioning index value increasing graph of the weighted objective function.

1	a(mm)	b(mm)	d(mm)	Theta	Sai	CI	vol	GCI
4822	200.0313	305.1959	98.97398	0.1	0	0.001807	28.21148	0.000984
4823	200.0645	304.7332	88.68052	0.1	0	0.001636	27.23871	0.000985
4824	200.0367	305.2536	155.9036	0.1	0	0.002593	32.97995	0.000985
4825	200.0068	305.2877	155.0147	0.1	0	0.002583	32.91182	0.000985
4826	200.0301	305.0755	53.63707	0.1	0	0.000985	23.72593	0.000985

Table 4-4 Conditioning Index against the set of GCI , workspace volume and the design variables for the multi objective maximum optimization.

Now for the case when workspace volume is in preference.

- Conditioning index =0
- Workspace volume =1
- Global conditioning index =0

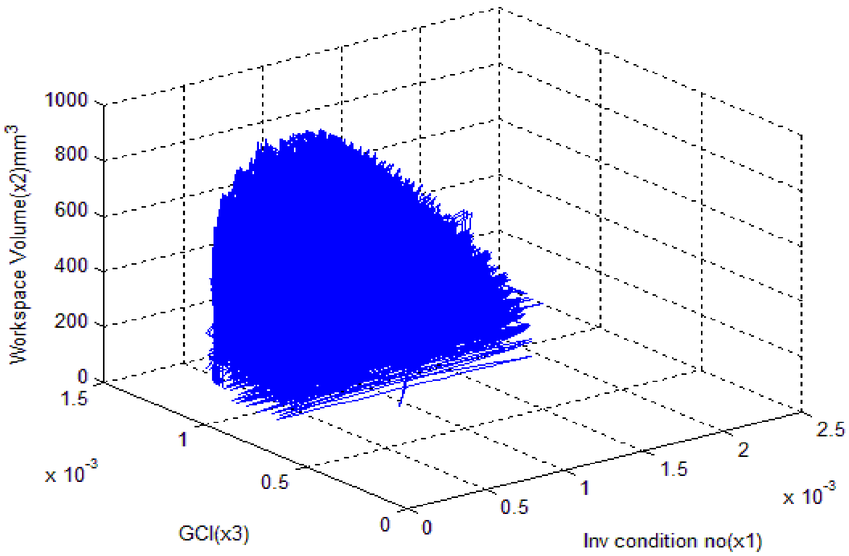


Figure 4.8 multi objective maxima curve with workspace volume given preference of 1

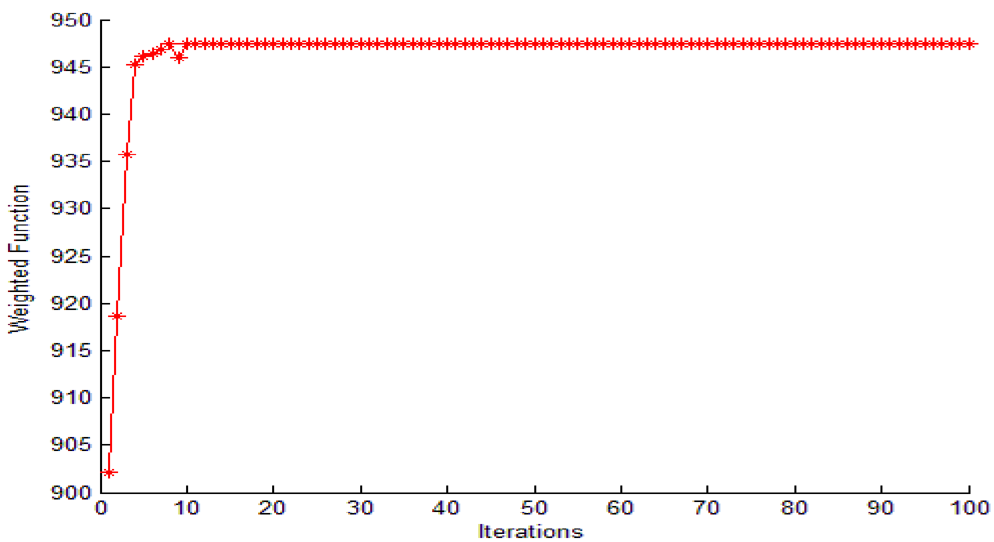


Figure 4.9 gbest volume value increasing graph of the weighted objective function.

1	a(mm)	b(mm)	d(mm)	Theta	Sai	Vol	CN	GCI
3346	299.6874	489.9829	189.5516	-0.3	1	898.3262	1417.812	0.001129
3347	298.1119	493.2253	198.189	-0.3	1	882.2358	1413.407	0.001129
3348	299.8518	486.945	172.3355	-0.3	1	947.46	1424.773	0.001128
3349	286.0697	489.0046	175.7706	-0.3	1	919.8657	1418.382	0.001128
3350	291.6233	453.9914	183.0379	-0.3	1	923.0926	1470.575	0.001128

Table 4-5: Workspace volume against the set of GCI , conditioning index and the design variables for the multi objective maximum optimization.

By the above results it is an analysis that workspace volume is inversely proportional to the conditioning index and conditioning index is directly proportional to global index. Results have been compared with single objective optimization and validated. Further results have been performed to check for giving the equal weightage to both conditioning index and the workspace volume.

Now for the case when workspace volume and the conditioning index both is given equal preference of half. Normalization process is carried out in which each performance index of swarm has been divided by the maximum value of each iteration to check the behavior of each performance index with respect to the other.

- Conditioning index =0.5
- Workspace volume =0.5
- Global conditioning index =0

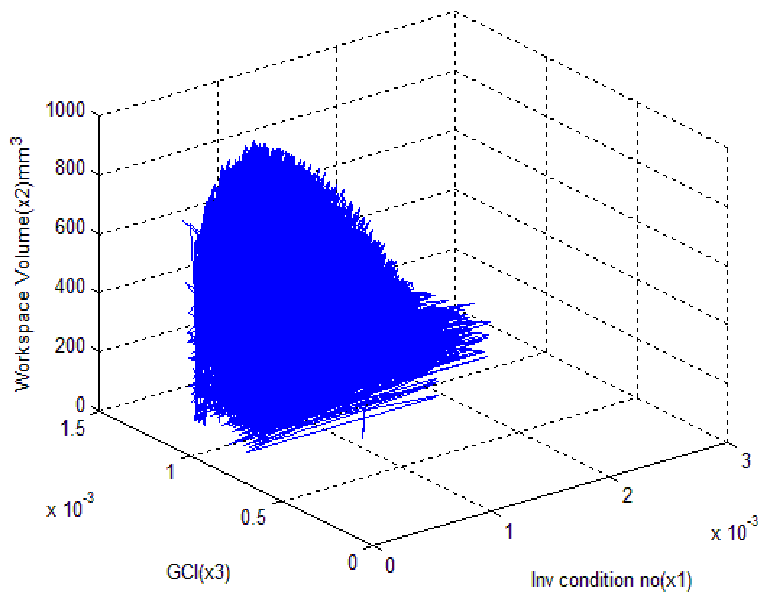


Figure 4.10 multi objective maxima curve with workspace volume and condition number given preference of 0.5.

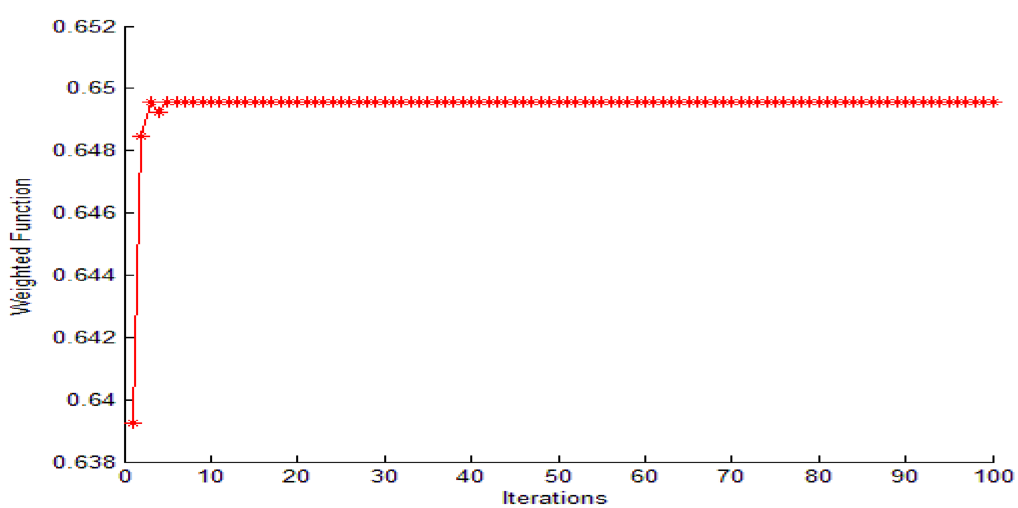


Figure 4.11 gbest compensation value increasing graph of the weighted objective function

The result obtained in figure 4.9 shows the compromising value when both the performance parameters are taken into the equal weightage. As this simulation is run for the 100 iterations. The code has find its optimum at about 20 iterations. This function reveals the result to be 466.45mm^3 which is an approximate value for the local maxima. As the result of conditioning index are in small fractions so the most of the value is in volumetric number but the participation of both the parameters are equal for the demonstration.

Here it must be clear that these values are near optimum local maxima for the tricept manipulator. These values are for the one elevation of ‘c’ for 500 mm elongation and the results shown previously is displayed for the multi objective algorithm to find for the local maxima. Other results can be shown for the local minima too in future and also some graphs for the global conditioning index can be taken into account. Well it usually exhibits the same results as it is inversely proportional to the workspace volume.

4.422 Epsilon Constraint Strategy

This strategy is introduced for the validation of the results of the multi objective weightage scheme and this technique is easy to control and hence have more reliable results. Sometimes weighted algorithm is challenging for the scholars and researchers. This technique basically reveals the result of performance parameters by restricting one of the all performance index as a constraint. It has more robustness and easy to control as its code can execute by controlling any of the parameter to certain limit and hence can check their responses. Here are some results below from this technique.

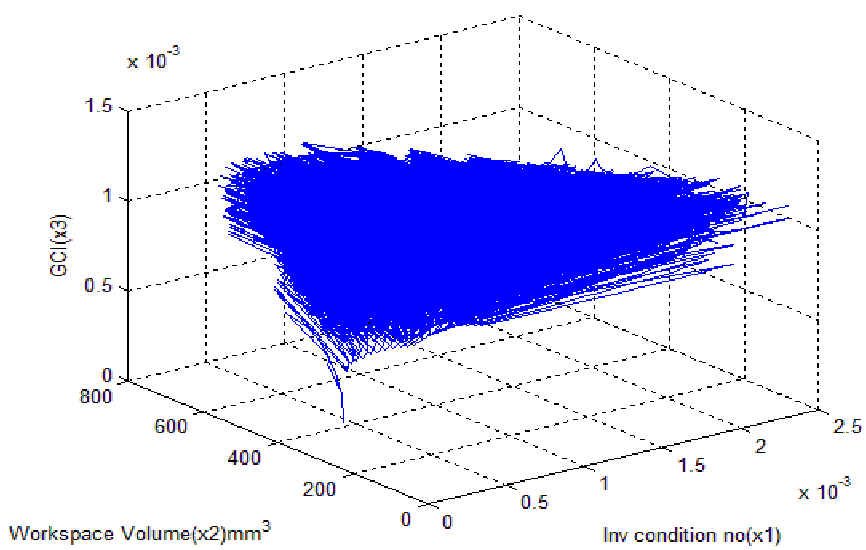


Figure 4.12 multi objective maxima curve with workspace volume restrict up to 700mm^3

From here workspace volume is being treated as a constraint and the conditioning index is shown to max under the 700 mm^3 volume restriction. Max Conditioning number at this point is .002593 which will regards as local maxima. This result is also shown from 100 iterations. It has got its optimum in 20 iterations.

1	a(mm)	b(mm)	d(mm)	Theta	Sai	Vol	CI	GCI
4411	200.1062	306.0383	88.68052	0.1	0	27.28973	0.001635	0.001012
4412	200.1541	306.1213	156.2243	0.1	0	33.06519	0.002593	0.001012
4413	200.141	306.069	156.2009	0.1	0	33.05932	0.002593	0.001012
4414	200.1247	306.0919	53.63707	0.1	0	23.75598	0.000985	0.001012
4415	200.1803	306.0872	108.1576	0.1	0	29.08988	0.00195	0.001012

Table 4-6 Workspace volume, conditioning index combined against the set of GCI and the design variables for the multi objective maximum constraint optimization.

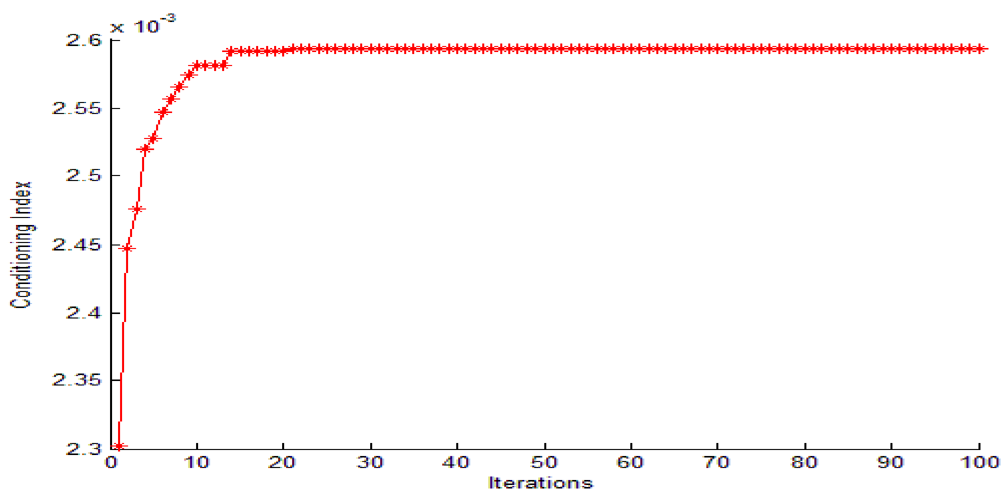


Figure 4.13 gbest compensation value increasing graph of conditioning index of the constraint volume up to 700mm^3

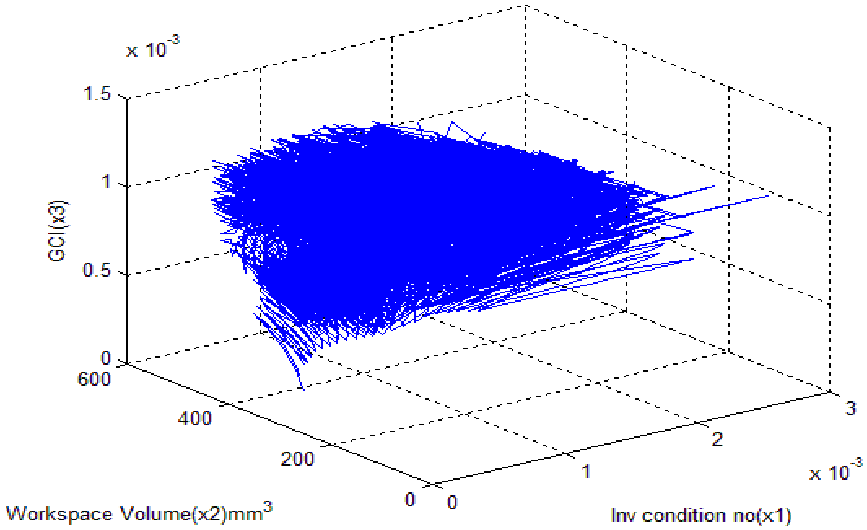


Figure 4.14 multi objective maxima curve with workspace volume restrict up to 500mm^3

From the simulation it come to know that the near optimum value up to 20 iterations is .0025 and we have given this time a limit of 500mm$^{3.}$ Now further more simulations can be run to make conditioning index as a constraint and check for the other performance index.

4.5 Conclusion and Discussion:

Now it is time to conclude the above thesis and gather all the points on which it ends up the discussion. Our bench mark paper [22] illustrates the single objective optimization using genetic algorithms. The multi objective optimization is an addition to the previous research work. Further it has been done through the particle swarm technique here with an ingredient of weighted and epsilon constraint strategy. Firstly this thesis work has given validation to the results shown in [22] for single objective optimization for volume and condition numbers optimization respectively. Next point to ponder is about the choice of best evolutionary algorithm that has been used in this paper (PSO) with a comparison of a (GA) used previously for this Tricept mechanism. It will summarize some remarks for this. In this work I have taken 100 iterations. It is random. It can try different iterations at start to see the variance of optimum points. The tries for PSO is clearly validated and we claim by [40, 41] that for my piece of work I found this number of iterations enough to attain any near optimum point. For this mechanism it is found that PSO actually evaluate the best optimum in about 20 iterations while the GA get close to the same optimum results in almost 40 to 50 iterations. Here it got the first point that the

PSO is relatively a fast one as compared to GA. Hence then This GA exerts more computation on the processor then PSO [42] .PSO has a higher convergence rate then GA for this task. On the other hand it have to accept the PSO on the cost of one thing and that is GA has the guarantee to find the best global maxima or minima point after that too much iterations while PSO find the best first optimum point and can find possibly the global optima in its search and declare it a best optimum point and [37] also quote this fact.

Secondly the results of weighted and the constraint epsilon optimum points are exactly the same for multi objective optimization when compared with the single objective optimization.

4.6 Future Suggestions:

This work has claimed to cover this tricept mechanism through the PSO algorithm. In future it can proceed our research through many other evolutionary algorithm techniques like ANT colony Optimization Technique and others. Secondly it can go for more performance parameters like stiffness index and the manipularity analysis. More constraints can be added. More structures can be added by small variation in the design of the parallel manipulators keeping in mind the actual concept of the tricept mechanism.

Appendix 1

```
               %% MATLAB CODE FOR THE CONDITIONING INDEX ANALYSIS
clear all
close all
clc
p=[];
q=[];
r=[];
n=input('enter the number of random values between lower and uper limits =
');
a=linspace(200,300,n)           %creating random values for a
a=transpose(a);
b=linspace(300,500,n)           %creating random values for b
b=transpose(b);
d=linspace(20,200,n)            %creating random values for d
d=transpose(d);
x=[a b d]
j=[];
ji=[];
ki=[];
thetas=[];
phis=[];
c=500
kif=[];
l=[];
         for theta=-1:.1:1
             for phi=-1:.1:1
             for l=1:1:n
  a=x(n,1);
  b=x(n,2);
  d=x(n,3);
           k=[];
           p=[(1+d*cos(theta)*cos(phi));
(1+d*cos(theta)*cos(phi)+sin(theta)/(2*sqrt(3))+cos(theta)*sin(phi));
(1+d*cos(theta)*cos(phi)+sin(theta)/(2*sqrt(3))-cos(theta)*sin(phi))];
           q=[(a*b*sin(theta)/3-c*d*sin(theta)*cos(phi)-
2*b*d*cos(phi)*cos(theta)/sqrt(3));(a*b*sin(theta)/12+sin(phi)*cos(theta)/(4*
sqrt(3))+b*d*cos(phi)*cos(theta)/(2*sqrt(3))-
c*d*sin(theta)*cos(phi)+(a*c*cos(theta))/(2*sqrt(3))-
(a*c*sin(theta)*sin(phi)/2));(a*b*sin(theta)/12-
sin(phi)*cos(theta)/(4*sqrt(3))+b*d/(2*sqrt(3))*cos(phi)*cos(theta)-
c*d*sin(theta)*cos(phi)+(a*c*cos(theta))/(2*sqrt(3))+(a*c*sin(theta)*sin(phi)
)/2)];
           r=[(-c*d*sin(phi)*cos(theta)-2*b*d*sin(phi)*sin(theta)/sqrt(3));
(-cos(phi)*sin(theta)/(2*sqrt(3))+sin(phi)/(2*sqrt(3))-
b*d*sin(phi)*sin(theta)/(2*sqrt(3))+b*d*cos(phi)/2-
c*d*sin(phi)*cos(theta)+a*c*cos(theta)*cos(phi)/2); (-
cos(phi)*sin(theta)/(2*sqrt(3))+sin(phi)/(2*sqrt(3))-
b*d*sin(phi)*sin(theta)/(2*sqrt(3))-b*d*cos(phi)/2-c*d*sin(phi)*cos(theta)-
a*c*cos(theta)*cos(phi)/2)];
           j=[p q r];                                %jacobian formulation
           ji=inv(j);
           k=norm(ji)*norm(j);
           k=1/k;                                     %Conditioning Index
```

```
          ki=vertcat(ki,k);
          thetas=vertcat(thetas,theta);
          phis=vertcat(phis,phi);
            end
            end
        end
[thetas]=meshgrid(thetas);
[phis]=meshgrid(phis);
[kif]=meshgrid(ki);
meshc(thetas,phis,kif)
```

Appendix 2

```
                    %% MAtlab code for the workspace volume

clear all
close all
clc
n=input('enter the number of random values between lower and uper limits =
');
a=linspace(200,300,n);          %creating random values for x1 to be put in
(1)
a=transpose(a);
b=linspace(300,500,n);          %creating random values for x1 to be put in
(1)
b=transpose(b);
d=linspace(20,200,n) ;        %creating random values for x1 to be put in (1)
d=transpose(d);
x=[a b d];
cq=[];
F=[];
G=[];
H=[];
qc=[];
s1=[];
s2=[];
s3=[];
r=1
p1=[];
s=[];
s31=[];
s32=[];
s33=[];
s21=[];
s22=[];
s23=[];
s11=[];
s12=[];
s13=[];
c=500;
rslt=[];
volver=[];
aver=[];
bver=[];
dver=[];
for z=0:200:800
    for alpha=0:10:360
p=[r*cosd(alpha) r*sind(alpha) z];
p1=vertcat(p1,p);
plot3(p1(:,1),p1(:,2),p1(:,3))
pause(.001)
grid on
hold on
    end
end
thetas=[];
```

```
phis=[];
    for theta=-1:.1:1
        for phi=-1:.1:1
                            for l=1:1:n
                                a=x(l,1);
  b=x(l,2);
  d=x(l,3);
q1=sqrt((a^2)/3+(b^2)/3+c^2+d^2-2/3*a*b*cos(theta)+2*c*d*cos(theta)*cos(phi)-
2*b*d/sqrt(3)*cos(phi)*sin(theta));
q2=sqrt((a^2)/3+(b^2)/3+c^2+d^2-1/2*a*b*((1/3)*cos(theta)-
1/sqrt(3)*sin(phi)*sin(theta)+cos(phi))+b*d*((cos(phi)*sin(theta))/sqrt(3)+si
n(phi))+2*c*d*cos(theta)*cos(phi)+a*c*(sin(theta)/sqrt(3)+cos(theta)*sin(phi)
));
q3=sqrt((a^2)/3+(b^2)/3+c^2+d^2-
1/2*a*b*((1/3)*cos(theta)+1/sqrt(3)*sin(phi)*sin(theta)+cos(phi))+b*d*((cos(p
hi)*sin(theta))/sqrt(3)-
sin(phi))+2*c*d*cos(theta)*cos(phi)+a*c*(sin(theta)/sqrt(3)-
cos(theta)*sin(phi)));
q=[q1 q2 q3];
qc=vertcat(qc,q);
if 400<=q1 && q1<=750
F=[a b d theta phi q1];
s1=vertcat(s1,F);
end
if 400<=q2 && q2<=750
G=[a b d theta phi q2];
s2=vertcat(s2,G);
end
if 400<=q3 && q3<=750
H=[a b d theta phi q3];
s3=vertcat(s3,H);

     end
        end
    end
    end
sz=[];
sc=[];
sc1=[];
cal=0;
sz=size(s1)
for i=1:1:sz(1,1)
    cal=sqrt(s1(i,4)^2+s1(i,5)^2)
    if cal<=1
        sc=s1(i,:);
        sc1=vertcat(sc1,sc);

    end
end
s11=(sc1(:,1));
s12=(sc1(:,2));
s13=(sc1(:,3));
[s11]=meshgrid(s11);
[s12]=meshgrid(s12);
[s13]=meshgrid(s13);
meshc(s11,s12,s13)
```

```
plot3(s11, s12 ,s13,'*k')
pause(2)
hold on
sz=[];
sc=[];
sc2=[];
cal=0;
sz=size(s2)
    for i=1:1:sz(1,1)
    cal=sqrt(s2(i,4)^2+s2(i,5)^2);
    if cal<=1
        sc=s2(i,:);
        sc2=vertcat(sc2,sc);
    end
    end
s21=(sc2(:,1));
s22=(sc2(:,2));
s23=(sc2(:,3));
[s21]=meshgrid(s21);
[s22]=meshgrid(s22);
[s23]=meshgrid(s23);
meshc(s21,s22,s23)
plot3(s21, s22 ,s23,'*r')
pause(2)
hold on
sz=[];
sc=[];
sc3=[];
cal=0;
sz=size(s3);
    for i=1:1:sz(1,1)
    cal=sqrt(s3(i,4)^2+s3(i,5)^2)
    if cal<=1
        sc=s3(i,:);
        sc3=vertcat(sc3,sc);
    end
    end
    q1min=min(sc1(:,6));
    q2min=min(sc2(:,6));
    q3min=min(sc3(:,6));
    q1max=max(sc1(:,6));
    q2max=max(sc2(:,6));
    q3max=max(sc3(:,6));
    qlow=[q1min;q2min;q3min];
    qhigh=[q1max;q2max;q3max];
    qlow=min(qlow);
    qhigh=max(qhigh);
    vol=3.142*(qhigh-qlow);
s31=(sc3(:,1));
s32=(sc3(:,2));
s33=(sc3(:,3));
[s31]=meshgrid(s31);
[s32]=meshgrid(s32);
[s33]=meshgrid(s33);
meshc(s31,s32,s33)
axis([-2 3 -2 3 0 1000]);
```

Appendix 3

```
        %% Matlab code for Single Objective PSO : conditioning index
clear all
close all
clc
x1=[];
x2=[];
xnewf=[];
v1=[];
v2=[];
v=[];
y=0;
c1=2;
c2=2;
gci=[];
iteration=[];
gold=[];
gbestm=[];
gbestver=[];
p=[];
q=[];
r=[];
m=input('enter the number of iterations for this PSO =');
n=input('enter the number of random values between lower and uper limits =
');
a=random('unif',200,300,1,n);         %creating random values for x1 to be
put in (1)
a=transpose(a);
b=random('unif',300,500,1,n);         %creating random values for x1 to be
put in (1)
b=transpose(b);
d=random('unif',20,200,1,n);         %creating random values for x1 to be put
in (1)
d=transpose(d);
x=[a b d];
kis=[];
j=[];
ji=[];
ki=[];
thetas=[];
phis=[];
c=500;
kif=[];
vg=[];
ps=-1;
tw=1;
ti=1;
tj=1;
rsltver=[];
 thetas=[];
    phis=[];
for iter=1:1:m
x=[a b d];
n=length(x);
```

```
yo=[];
pbestver1=[];
pbestver2=[];
pbestver3=[];
rsltver=[];
ki=[];
aver=[];
bver=[];
dver=[];
for theta=-1:0.2:ti
    for phi=-1:0.2:tj
            for l=1:1:n
                 a=x(l,1);
  b=x(l,2);
  d=x(l,3);
   for tz=0:1:ps
  theta=thetas(l,1);
  phi=phis(l,1);
  end

            k=[];
           p=[(1+d*cos(theta)*cos(phi));
(1+d*cos(theta)*cos(phi)+sin(theta)/(2*sqrt(3))+cos(theta)*sin(phi));
(1+d*cos(theta)*cos(phi)+sin(theta)/(2*sqrt(3))-cos(theta)*sin(phi))];
           q=[(a*b*sin(theta)/3-c*d*sin(theta)*cos(phi)-
2*b*d*cos(phi)*cos(theta)/sqrt(3));(a*b*sin(theta)/12+sin(phi)*cos(theta)/(4*
sqrt(3))+b*d*cos(phi)*cos(theta)/(2*sqrt(3))-
c*d*sin(theta)*cos(phi)+(a*c*cos(theta))/(2*sqrt(3))-
(a*c*sin(theta)*sin(phi)/2));(a*b*sin(theta)/12-
sin(phi)*cos(theta)/(4*sqrt(3))+b*d/(2*sqrt(3))*cos(phi)*cos(theta)-
c*d*sin(theta)*cos(phi)+(a*c*cos(theta))/(2*sqrt(3))+(a*c*sin(theta)*sin(phi)
)/2)];
           r=[(-c*d*sin(phi)*cos(theta)-2*b*d*sin(phi)*sin(theta)/sqrt(3));
(-cos(phi)*sin(theta)/(2*sqrt(3))+sin(phi)/(2*sqrt(3))-
b*d*sin(phi)*sin(theta)/(2*sqrt(3))+b*d*cos(phi)/2-
c*d*sin(phi)*cos(theta)+a*c*cos(theta)*cos(phi)/2); (-
cos(phi)*sin(theta)/(2*sqrt(3))+sin(phi)/(2*sqrt(3))-
b*d*sin(phi)*sin(theta)/(2*sqrt(3))-b*d*cos(phi)/2-c*d*sin(phi)*cos(theta)-
a*c*cos(theta)*cos(phi)/2)];
           j=[p q r];
           ji=inv(j);
           k=norm(ji)*norm(j);
          k=1/k;
           ki=vertcat(ki,k);
           aver=vertcat(aver,a);
           bver=vertcat(bver,b);
           dver=vertcat(dver,d);
           for tz=1:1:tw
           thetas=vertcat(thetas,theta);
           phis=vertcat(phis,phi);
           end
         rslt=[a b d theta phi k];
         rsltver=vertcat(rsltver,rslt);


            end
```

```
    end
end
n=length(ki);
v1=random('unif',0,1,1,n);          %creating random values for v1 to be put
in (2)
v1=transpose(v1);
v2=random('unif',0,1,1,n);          %creating random values for v2 to be put
in (2)
v2=transpose(v2);
v3=random('unif',0,1,1,n);
v3=transpose(v3);
v=[v1 v2 v3];
gbestm=min(ki);
gbestver=vertcat(gbestver,gbestm);
           for pz=0:1:tw
     gold=gbestm;
           end
 if gold>=gbestm
    for i=1:1:n
    if ki(i,1)==gbestm
    i;
        gbest1=aver(i,1);
        gbest2=bver(i,1);
        gbest3=dver(i,1);
        gbest=[gbest1;gbest2;gbest3];
    end
    end
         for i=1:1:n
    if ki(i,1)==gbestm
        vg1=v(i,1);
        vg2=v(i,2);
        vg3=v(i,3);
        vg=[vg1;vg2;vg3];
    end
         end
    vgf=0;
%%     updated velocities
for i=1:1:n
    r1=random('unif',0,1,1,1);
    r2=random('unif',0,1,1,1);
        pbest1=aver(i,1) ;
         pbestver1=vertcat(pbestver1,pbest1);
    vnew=vg(1)+c1*r1*(pbest1-aver(i,1))+c2*r2*(gbest(1)-aver(i,1));
                            %     updated position
xnew=aver(i,1)+vnew;
if xnew>=200 && xnew<=300
if xnew<=pbest1
    xnewf(i,1)=xnew;
    else
    xnewf(i,1)=aver(i,1);
end
else
    xnewf(i,1)=aver(i,1);


end
  pbest2=bver(i,1);
    pbestver2=vertcat(pbestver2,pbest2);
```

```
    vnew=vg(2)+c1*r1*(pbest2-bver(i,1))+c2*r2*(gbest(2)-bver(i,1)) ;
                                %     updated position
    xnew=bver(i,1)+vnew;
if xnew>=300 && xnew<=500
if xnew<=pbest2

    xnewf(i,2)=xnew;
    else
    xnewf(i,2)=bver(i,1);
end
else
    xnewf(i,2)=bver(i,1);

end
  pbest3=dver(i,1);
  pbestver3=vertcat(pbestver3,pbest3);
    vnew=vg(3)+c1*r1*(pbest3-dver(i,1))+c2*r2*(gbest(3)-dver(i,1));
                                %     updated position
    xnew=dver(i,1)+vnew;
if xnew>=20 && xnew<=200
if xnew<=pbest3

    xnewf(i,3)=xnew;
    else
    xnewf(i,3)=dver(i,1);
end
else
    xnewf(i,3)=dver(i,1);

end
end
 end
a=xnewf(:,1);
b=xnewf(:,2);
d=xnewf(:,3);
gold=gbestm;
iteration=vertcat(iteration,iter);
tw=-1;
 ps=0;
 ti=-1;
 tj=-1;
 theta=[];
 phi=[];
 gold=gbestm;

end
     gold
```

REFERENCES

[1] Mark W. Spong, 1995, "Adaptive Control of Flexible Joint Manipulators, Comment on two papers"; Automatica, vol31, no.4, pp585-590.

[2] J.F.He, H.Z.Jiang, D.C.Cong, Z.M.Ye and J.W.Han 2007, "A Survey on Control of Parallel Manipulator"; Key Engineering Materials, vol.339, pp307-313.

[3] C.Gosselin, J.Angeles, 1989 "The Optimum kinematic Design of a Spatial three Degree of Freedom Parallel Manipulator"; Journal of Mechanical Design, vol.111(2),pp.202-207.

[4] Y.D.Patel, P.M.George 2012, "Parallel Manipulator Application A Survey"; Modern Mechanical Engineering, 2, 57-64.

[5] Chao-Chia D.Lu, M.John D.Hayes, "Kinematic Calibration of 6R Serial Manipulators Using Relative Measurements".

[6] Marco Antonio Meggiolaro, Guglielmo Scriffignano and Steven Dubowsky. "Analysis and Optimization of closed Loop Manipulator Calibration with Fixed End Point".

[7] A Campos, R.Guenther, D.Martins, 2005, "Differential kinematics of Serial Manipulators using Virtual Chains"; Journal of Brazilian Society of Mechanical Sciences and Engineering, vol.27 no.4, Brazil.

[8] Denny Oetomo, David Daney and Jean-Pierre Merlet, 2009 "Design Strategy of Serial Manipulators with Certified Constraint Satisfaction"; IEEE Transaction on Robotics, vol.25, no.1.

[9] Lung Wen Tsai; "Robot analysis, The Mechanics of Serial and Parallel Manipulators"; ISBN 0-471-32593-7.

[10] Daniel Glozman and Moshe Shoham, 2008, "Novel 6 DOF Parallel Manipulator with Large Workspace"; Robotica, pp.1-5.

[11] Sergiu-Dan Stan,Vistrian Maties and Radu Balan,2008,"Optimization of a 2 DOF Micro Parallel Robot using Genetic Algorithms"; Frontiers in Evolutionary Robotics, INTECH, ISBN 978-3-902613-19-6.

[12] Dan Zhang, 2009, "Parallel Robotic Machine Tools"; Springer ISBN 978-1-4419-1116-2.

[13] K.A.Arrouk, B.C.Bouzgarrou and G.Gogu, 2010, "Workspace Determination and Representation of Planar Parallel Manipulators in a CAD Environment"; Mechanisms and Machine Science, vol.5, pp.605-612.

[14] Viv Kumar Mehta and Bhaskar Dasgupta, 2011, "A General Approach for Optimal Kinematic Design of 6dof Parallel Manipulators"; Sadhana, Vol.36, Part 6, pp.977-994.

[15] Dan Zhang; "Parallel Robotic Machine Tools"; Ch-1, Springer ISBN 978-1-4419-1116-2.

[16] Y.J.Lou, G.F.Liu and Z.X.Li, 2005, "A General Approach for Optimal Design of Parallel Manipulators"; IEEE Automation science and Engineering, Vol. X no. X, XX.

[17] J-P Marlet, 1997, "Designing a Parallel Manipulator for a Specific Workspace"; the International Journal of Robotic Research, 16(4): 545-556.

[18] M.Guillot, C.M.Gosselin, 1991, "The Synthesis of Manipulator with Prescribed Workspace"; Transaction of the ASME, Journal of Mechanical Design, 113(3):451-455.

[19] Peng Bin Bin, Li Zengming, 2011, "Kinematic Characteristics 3-UPU Parallel Manipulator in Singularity and its Application"; INTECH.

[20] Xu Zhang, 2012, "Optimization of 3-PRS Parallel Manipulator based on Interval Analysis"; Proceedings of the 10th World Congress in Intelligent Control and Automation, China.

[21] Bruno Siciliano, 1999, "The Tricept Robot Inverse Kinematics, Manipulability Analysis and Closed loop Direct kinematic Algorithm"; Robotica, Volume 17, pp.437-445, Cambridge University Press.

[22] Mir Amin Hosseini, Taghirad, 2011, "Dexterous Workspace Optimization of a tricept Parallel Manipulator"; Advanced Robotics 25 1697-1712.

[23] Damien Chablet and Philipe Wenger, "Architecture Optimization of the Orthoglide, a 3Dof Parallel Mechanism for Machining Applications."

[24] Clement Gosselin, 1985, "Kinematic Analysis, Optimization and Programming of Parallel Robotic Manipulators"; Ch-3.

[25] Dietmaier p., 1998, "The Stewart–gough Platform of general Geometry can have 40 real postures"; Advances in Robot Kinematics: Analysis and Control, pp.1-10.

[26] Min-Jie Liu, Cong-Xin Li and Chong-Ni Li, 2000,"Dynamics Analysis of the Gough-Stewart Platform Manipulator"; IEEE Robotics and Automation, Vol.16, no.1,pp-94-98.

[27] Haidong Li, clement M. Gosselin, marc J. Richard and Boris Mayer st-Onge, 2005, "Analytic Form of six Dimensional Singularity Locus of the General Gough-Stewart Platform"; Journal of Mechanical Design, vol 128, no.1,pp-279-287.

[28] J.J Craig, 2003, "Introduction to Robotics, Mechanics and Control"; Prentice Hall, ISBN: 978-0-201-54361-2, New York.

[29] Geoff T.Pond and Juan A.Carretero, 2007, "Quantitative Dexterous Workspace Comparison of Parallel Manipulators"; Mech. Mach. Theory 42, 1388-1400.

[30] J.A.Carretero, R.P.Podhorodeski, M.A.Nahon and C.M.Gosselin, 2000, "Kinematic Analysis and Optimization of a New Degree of Freedom Spatial Parallel Manipulator"; Journal of Mechanical Design, vol 122, No.1,pp.17-24

[31] Kalyanmoy Deb, "Single and Multi objective Optimization using Evolutionary Computation"; Kangal Report number 2004002.

[32] Zar chi su su haling and May Aye khine, 2011, "Solving Traveling Salesman Problem by using Improved Ant Colony Optimization Algorithm"; International Journal of Information and Education and Technology, Vol.1.no.5.

[33] Dorigo M., Maniezzo V., and Colorni A., 1996, "The Ant System, Optimization by a colony of Co-operating Agent"; IEEE Transaction on Systems, Man and Cybernetics, Part B, Vol 26-1, pp 29-41.

[34] Dian Pulupi Rini, Siti Mariyam and Siti Sophiyati, 2011,"Particle Swarm Optimization, Technique System and Challenges"; International Journal of Computer Applications, 0975-8887, vol 14-no.1.

[35] Kennedy J., 1999, "Minds and Cultures, Particle Swarm Implications for beings in Socio cognitive Space"; Adaptive Behavior Journal, 7, 269-288.

[36] Yamille Del Valle, Ganesh Kumar Venayagamoorthy, Salman Mohagheghi, Jean-Carlos Hernandez and Ronald G.Harley, 2008, "Particle Swarm Optimization, Basic Concepts, Variants and Applications in Power System"; IEEE Transactions on Evolutionary Computation, vol.12, no 2.

[37] Rega Rajendra, Dilip K. Pratihar, 2011, "Particle Swarm Optimization Algorithm vs Genetic Algorithm to Develop Integrated Scheme for Obtaining Optimal Mechanical Structure and Adaptive Controller of a Robot"; Intelligent Control and Automation, 2, 430-449.

[38] Tushargoel, Nielen Strander, 2007, “Multiobjective Optimization using LS-OPT”; LS-DYNA Anwenderforum, Frankenthal , Livermore Software Technology Coorporation USA.

[39] Cui Guohua, Wei Bin, Wang Nan and Zhang Yanwei, 2013, “Stiffness, Workspace Analysis and Optimization for 2UPU Parallel Robot Mechanism”; TELKOMNIKA, vol.11, no.9, pp.5253-5261 e-ISSN: 2087-278X.

[40] Shu-Chuan Chu, John F. Roddick and Jeng-Shyang Pan, “A Parallel Particle Swarm Optimization Algorithm with Communication Strategies”.

[41] Antonio M.Lopes, Helio Freire,P.B de Moura Oliveira and E.J Solteiro Pires, 2011, “Multiobjective Optimzation of a Parallel Robot using Particle Swarm Algorithm”; ENOC,pp24-29, Italy.

[42] Rania Hassan, Babak Colhanium and Oliver de Weck, “A Comparison of Particle Swarm Optimization and the Genetic Algorithm”.