SYNTHESIS OF PLANAR PARALLEL MECHANISM

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR DEGREE OF

Bachelor of Technology
in
Mechanical Engineering

By

KUSHAGRA NIGAM
Roll No: 1O8ME044

Department of Mechanical Engineering
National Institute of Technology
Rourkela
2011-2012

SYNTHESIS OF PLANAR PARALLEL MECHANISM

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR DEGREE OF

Bachelor of Technology
in
Mechanical Engineering

By

KUSHAGRA NIGAM
Roll No: 1O8ME044

Under the Guidance of
Prof. J. Srinivas

Department of Mechanical Engineering
National Institute of Technology
Rourkela

2011-2012

National Institute of Technology

Rourkela

CERTIFICATE

This is to certify that the thesis entitled, **"Synthesis of Planar Parallel Mechanism"** submitted by Sri **Kushagra Nigam** in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Mechanical Engineering at the **National Institute of Technology, Rourkela** (Deemed University) is an authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in this thesis has not been submitted to any other University/Institute for the award of any Degree or Diploma.

Date:

Prof. J. Srinivas

Department of Mechanical Engineering

National Institute of Technology

Rourkela-769008

ACKNOWLEDGEMENT

It is with a feeling of great pleasure that I would like to express my most sincere heartfelt gratitude to Prof. J.Srinivas, Dept. of Mechanical Engineering, NIT Rourkela, for suggesting the topic for my thesis report and for his ready and able guidance throughout the course of my preparing the report. I am greatly indebted to him for his constructive suggestions and criticism from time to time during the course of progress of my work.

I express my sincere thanks to Prof. K.P. Maity, Head of the Department of Mechanical Engineering, NIT Rourkela, for providing me the necessary facilities in the department.

I feel pleased and privileged to fulfill our parents' ambition and I am greatly indebted to them for bearing the inconvenience during my M.E. course.

Date:

KUSHAGRA NIGAM
(108ME044)

ABSTRACT

Parallel mechanisms are found as positioning platforms in several applications in robotics and production engineering. Today there are various types of these mechanisms based on the strcture, type of joints and degree of freedom. An important and basic planar mechanism providing three degree of freedom at the end-effector (movable platform) is a 3-R$\underline{P}$R linkage. Here the underscore below P indicates the presence of actuated prismatic joints and 3 indicates the number of legs used to carry the mobile platform. A lot of work has been done on this mechanism since 1988. In the present work, the kinematics of 3-R$\underline{P}$R linkage is specifically studied to understand the synthesis procedure. The forward kinematics in parallel mechanisms is a multi-solution problem and involves cumbersome calculations compared to inverse kinematics. In inverse kinematics, we design the actuator input kinematic parameters for a known table center coordinates. In other words it is a transformation of platform pose vector to the actuator degrees of freedom. In 3-R$\underline{P}$R mechanism considered in present task, the actuators are sliders and hence the slider displacements reflect the input degrees of freedom. On the other hand, for a known posture (available slider displacement status), the table center coordinates are predicted in forward kinematics. In present work, forward kinematics solutions are obtained by defining error function and optimizing it using genetic algorithms programs. Also, the workspace and Jacobian matrices are computed at corresponding solution and singularity analysis is briefly highlighted. Main objective is to fabricate a scaled model of this planar manipulation mechanism with calculated dimensions and observe the practical workspace obtained. An attempt is made in this line to some extent.

CONTENTS

CHAPTER-1

INTRODUCTION

Mechanism is defined as Rigid bodies connected by joints in order to accomplish a desired force and motion . Parallel mechanism are closed-loop mechanisms where all of the links are connected to the ground and the moving platform at the same time. They have high rigidity, load capacity, precision and especially structural stiffness, since the end- effector is linked to the movable plate at several points. The spatial parallel mechanism that have three to six degrees of freedom (DOFs) can translate and rotate in the three dimensional space. One of the most popular spatial manipulators is the Gough–Stewart platform which is extensively used in flight simulation. The second group is planar parallel mechanism (PPMs) which translate along the x- and y-axes, and rotate around the z-axis, only. Synthesis of mechanism refers to design a linkage for a prescribed motion or path or velocity of tracing joint or link.

1.1 Types of planar parallel mechanism

Planar parallel mechanism are classified on the basis of following

1. Structure .
2. Degree of freedom.

1.1.1 3-RPR Mechanism

Figure 1 shows a 3-RPR mechanism designates that the end effector is connected to the base by three serial kinematic chains consisting of a passive revolute (R) joint connected to the base, followed by an actuated and (thus underlined prismatic) P joint, and followed by a passive revolute (R) joint connected to the end-effector.

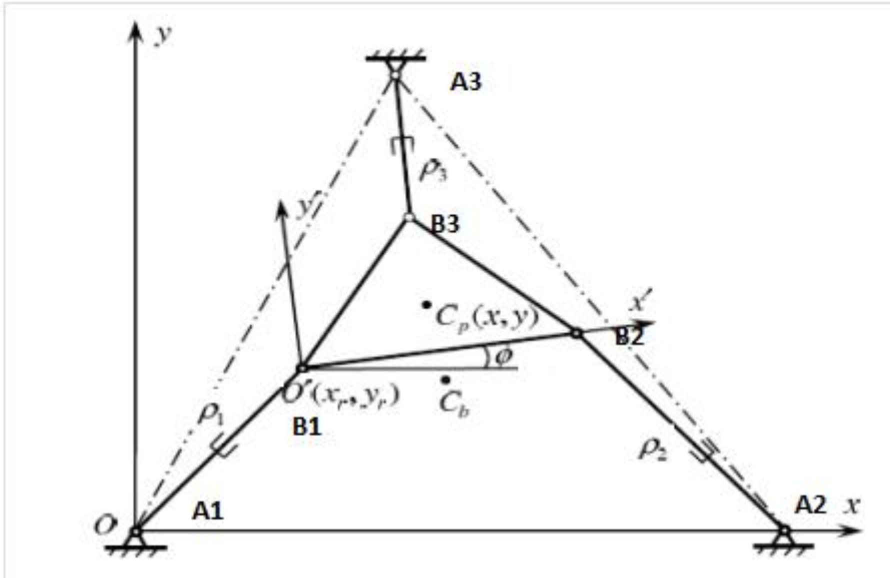


Figure 1 :Planar 3-RPR parallel mechanism

1.1.2. 2-RPR Mechanism.

A general two-degrees-of-freedom planar parallel mechanism actuated with prismatic joints (2-RPR) mechanism is shown in Fig. 2.

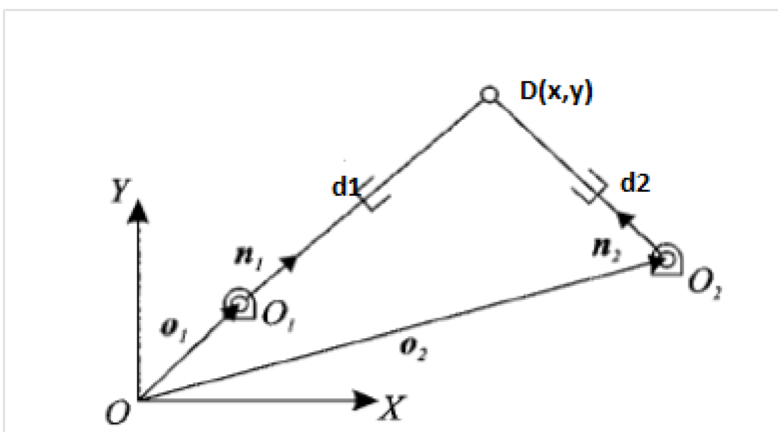


Figure 2. Planar 2-RPR parallel mechanism.

The lengths of the segments *OiD* in this section(*i*=1, 2) are denoted by ρ*i*. The end effector (*C*) can be positioned in a plane by modifying the lengths of these segments within the permissible ranges of the prismatic actuators ($\rho_{i,min} \leq \rho_i \leq \rho_{i,max}$). A fixed reference frame (*OXY*) is defined. The output of the mechanism, corresponding to the position of the end effector, can thus be expressed as

$$\boldsymbol{x} = [x, y]^T \quad (1)$$

The mechanism's input corresponds to the lengths of the prismatic actuator

$$\varphi = [\rho_1 , \rho_2]^T \quad (2)$$

1.1.3. 3-RRR Mechanism.

A general three-degrees-of-freedom planar parallel mechanism actuated with revolute joints (3-RRR mechanism) is illustrated in Fig. 3. The length of all proximal links (O_iB_i)) is l_1, while that of all distal links (B_iC_i) is l_2. A fixed reference frame (*OXY*) is defined as being attached to the mechanism's base. Furthermore, a mobile reference frame (*Cx'y'*) is defined as being attached to the end effector. As was the case for the 3-RPR mechanism, quantities expressed in the mobile frame will henceforth accompanied by the ' symbol. Vectors along lines OB_i, O_iB_i , O_iC_i , and CC_i are represented by o_i , u_i , r_i, and s_i respectively. Unit vectors n_i are also defined as being directed from points B_i to C_i. The output mechanism is

$$\boldsymbol{x} = [x, y, \varnothing]^T \quad (3)$$

while its input is expressed as the angular positions of the revolute actuators measured from the X axis to each of the serial kinematic chain's proximal links

$$\varphi = [\Theta_1 , \Theta_2, \Theta_3]^T$$

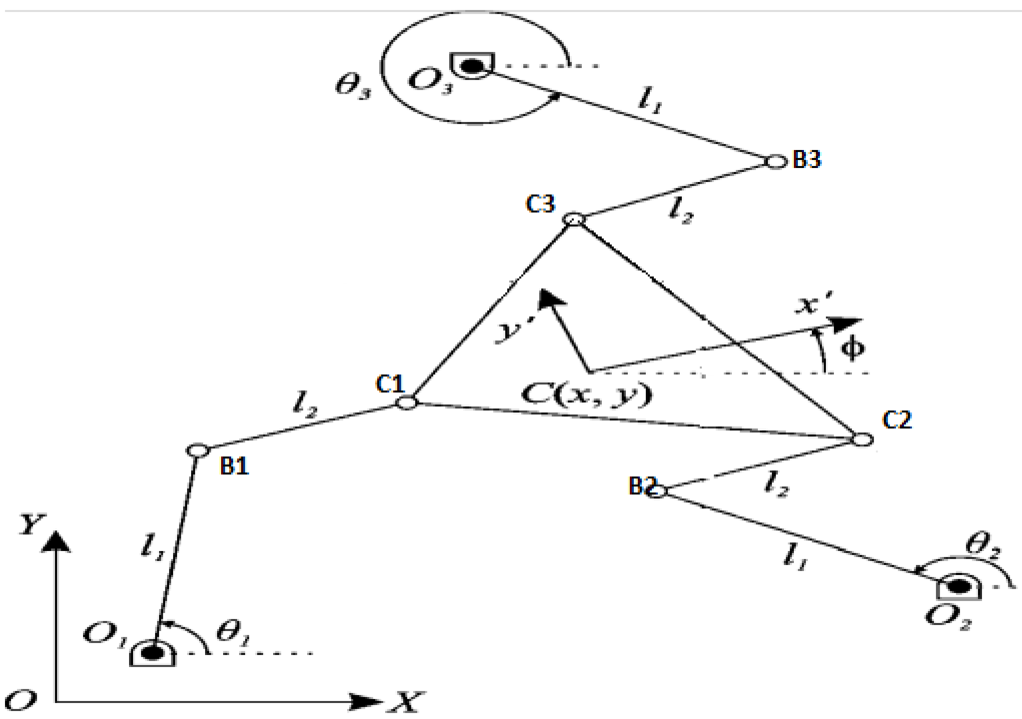


Figure 3. Planar 3-RRR parallel mechanism

1.2 Workspace

Workspace is defined as regions which can be reached by a reference point C located on the mobile platform.There are several definitions for the workspace of parallel mechanisms . Constant orientation workspace corresponds to the set of positions reachable by the end effectors as it translates at a fixed orientation. Maximal workspace is defined as the region which can be reached by point C with at least one orientations . The maximal workspace is also referred to as a reachable workspace . Inclusive workspace is defined as the region which can be obtained by point C with at least one orientation in a given range . The total orientation workspace corresponds to the region which can be reached by point C with every orientation of a platform in a given range. Dextrous workspace is given as a region which can be reached by point C with any orientation of the platform. For all three mechanisms (3-RPR, 2-RPR, 3RRR), it can be shown that the locus of points attainable by the end-effector *C* will have concentric circles centered at (a_i, b_i) as boundaries.

$$(x - a_i)^2 + (y - b_i)^2 = \rho_i^2 \qquad (4)$$

where *i*=1, 2 for the 2-RPR mechanism, and *i*=1, 2, 3 for the 3-RPR and the 3-RRR mechanisms. Note that for the 3-RPR and the 3-RRR mechanisms, Eq. (9) holds only for a constant orientation of their platforms. For the mechanisms with prismatic joints, two circles are obtained when the actuator lengths are set to its boundary values ($\rho_{i,min}$, $\rho_{i,max}$). For the 3-RRR mechanism, the minimum and maximum radii correspond to the configurations where the proximal and distal links are aligned, i.e., $\rho_{i,minmax} = | l_1 \pm l_2 |$

1.3 Singularities

In parallel mechanism, the singularity is an inherent geometric state that corresponds to an uncertainty configuration at which the system exhibits uncontrollable transitory degree of freedom(s). This happens when all the force are coincident at a point.

$$[J]\dot{\theta} = \dot{v} \qquad (5)$$

Condition for singularity

$$\| J \| = 0 \qquad (6)$$

Where , [J] is known as jacobian matrix.

Types of singularities.

(A) Direct kinematic singularity.
(B) Inverse kinematic singularity.

Direct kinematic singularity occurs inside the Cartesian workspace of the parallel manipulator. At this configuration, the end-effector can make infinitesimal motion even if the actuators are locked. Inverse kinematic singularity occur whenever any chain is in a completely stretched out or folded back. The corresponding configurations are located at the boundaries of the manipulator's workspace. At this configuration, infinitesimal rotations of the input links cannot produce motion in the end-effector. At this configuration, infinitesimal rotations of the input links cannot produce motion in the end-effector.

1.4 Jacobain matrix

A Jacobian matrix can be obtained for a parallel manipulator as follows. Let the actuated joint variables and the location of the moving platform be denoted by the vectors q and x, respectively. Then the kinematic relations can be written in the general form as f(x,q)=0 where f is the function of $x = (x, y, \emptyset)^T$ and q=$(q_1, q_2, q_3)^T$ and 0 is an n-dimensional zero vector. The variables x, y and Ø are the coordinates of the end-effector point P with respect to the base and orientation of the platform, respectively. Moreover, q_1, q_2 and q_3 denote actuated joints. Differentiating the f with respect to the time, $Ax^{\cdot} + Bq^{\cdot} = 0$ is obtained. Here $x^{\cdot}$ and $q^{\cdot}$ are the time derivatives of x and q, respectively. Here A and B are two separate Jacobian matrices. The overall Jacobian matrix for a parallel manipulator can be obtained as

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

1.5 Literature review

Following works are related to the present thesis.

Arsenault & Boudreau [1] presented a reliable synthesis method capable of optimally selecting the geometrical parameters of planar parallel mechanisms. Three different architectures are considered and a genetic algorithm is used to perform the optimization. The performance of each mechanism is evaluated according to four different criteria: workspace, singular configurations, dexterity, and stiffness. In order to make the synthesis method as realistic as possible, mechanical constraints affecting the angular rotation of the 2-RPR and 3-RPR mechanisms' passive revolute joints are considered. Moreover, since the conventional methods for computing the dexterity and the stiffness index are not valid for the 3-RPR and 3-RRR mechanisms, an alternative computation method is used.

Huang & thebert [2] considered a kinematic structure with three in parallel actuated, R-R-R, and R-P-R serial chain geometries.

Jiang & Gosselin [3] analyzed the effects of the orientation angle, the minimal leg length as well as the base shape on the singularity-free workspace using the Gauss divergence theorem.

Caro et al.[4] found the variations geometric parameters of parallel kinematics machine (PKMs) can be either compensated or amplified.

Sefrioui & Gosselin [5] obtained a graphical representation of singularity loci of general three degree of freedom in the maniplulator's workspace.

Gallant & Boudreau [6] synthesized the three-degree-of-freedom planar parallel manipulators using a genetic algorithm. The architecture of a manipulator and its position and orientation with respect to a prescribed workspace are determined.

Chandra & Rolland [7] applied hybrid metaheuristics for solving the forward kinematics of the 3RPR parallel manipulator.

Kucuk [8] developed a novel interactive simulation and design tool based on a MATLAB graphical user interface (GUI) for the performance analysis of planar parallel manipulators (PPMs), which are a special group among the other parallel robot manipulators.

Wenger & Chablat [9] analyzed a class of analytic planar 3-RPR manipulators. These manipulators have congruent base and moving platforms and the moving platform is rotated of 180 deg about an axis in the plane. The forward kinematics is reduced to the solution of a 3rd-degree polynomial and a quadratic equation in sequence. The singularities are calculated and plotted in the joint space. The second-order singularities (cups points), which play an important role in non-singular change of assembly-mode motions, are also analyzed.

1.6 Objective of present work

Evolutionary algorithms such as genetic algorithm have obtained solutions of high accuracy in optimization problems, therefore, it is reasonable to apply genetic algorithm in solving problems that involve non-linear equation systems which are still an open problem. Genetic algorithms have been applied for solving the forward kinematics of 3RPR parallel mechanism (FKP). Parallel mechanism problems are mostly associated with solving a system of non-linear equations and are rarely treated as a direct optimization problem. Therefore, the problem of solving a system of non-linear equation has to be converted into an optimization problem where the objective function describes the entire parallel mechanism kinematics. Each optimization approach has as own advantages and disadvantages in term of convergence accuracy, reliability, complexity and speed. Initially the forward kinematic solutions of 3-RPR mechanism are obtained as an optimization solution. The objective here is the distance between the connecting points of the limbs with platform as calculated from the moving coordinate frame and fixed reference frame, which is minimized so as to obtain the required pose of the platform corresponding to a known actuated lengths. When this distance (error) is zero for all the limbs, the manipulators achieve a possible configuration for desired position and orientation of mobile platform. Workspace is determined while the coordinates of base with respect to fixed reference frame, platform with respect to mobile reference frame and minimum and maximum leg lengths of prismatic joints are known for 3-RPR. Jacobian matrix has been found out for given orientation of 3RPR.

The remainder thesis has been organized as follows

Chapter 2 - Explains the forward kinematic equation and objective function to be minimized.

Chapter 3 - Describes the genetic algorithm optimization method adopted in present work.

Chapter 4 – Formulae for calculation Jacobian matrix and workspace.

Chapter 5 – Results and discussion.

Chapter 6 – Conclusion

CHAPTER 2

FORWARD KINEMATIC EQUATION AND OPTIMIZED FUNCTION

2.1 Forward kinematics of 3-RPR mechanism

Figure 4 shows a general 3-RPR manipulator, constructed by connecting a triangular moving platform to a base with three RPR legs. The actuated joint variables are the three link lengths ρ1, ρ2 and ρ3. The output variables are the position coordinates (*x*, *y*) of the operation point *P* chosen as the attachment point of link 1 to the platform, and the orientation φ of the platform. A reference frame is centred at *A*1 with the *x*-axis passing through *A*2. Notation used to define the geometric parameters of the manipulator is shown in Fig 4. The inverse kinematics constraint equations are as follows:

$$\rho_1{}^2 = x^2 + y^2 \tag{7}$$

$$\rho_2{}^2 = (x + l_2\cos(\varphi) - c_2)^2 + (y + l_2\sin(\varphi))^2 \tag{8}$$

$$\rho_3{}^2 = (x + l_3\cos(\varphi + \beta) - c_3)^2 + (y + l_3\sin(\varphi + \beta) - d_3)^2 \tag{9}$$

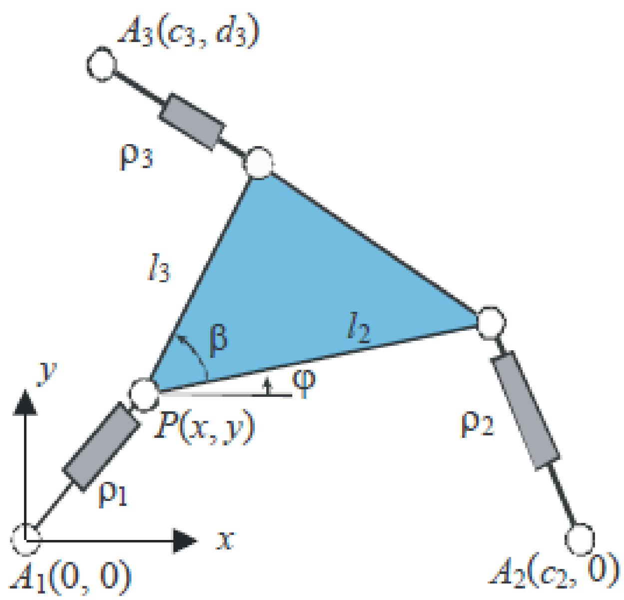


Figure 4 .Planar 3-RPR parallel mechanism

2.2 Distance error as objective function

Optimization is carried by genetic algorithms for minimizing the fitness function . Therefore, we need to effectively convert the problem which solves a system of equations into an optimization problem. The fitness function represents the total error on each leg length as shown in Fig.5. Let L_{gi} be the leg length of kinematics chain i which is given as input of the problem.

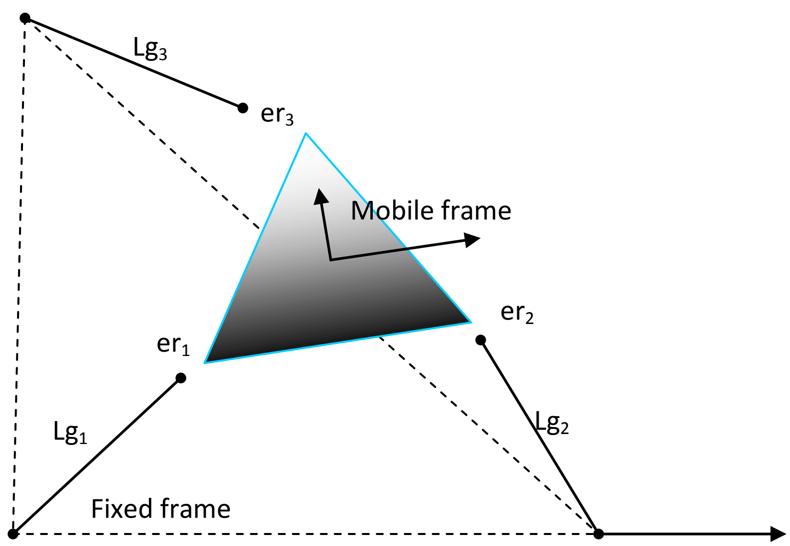


Fig.5 Calculation of errors in 3-RPR mechanism

Therefore, the fitness function is given in Eq. (10)

$$F(x)= \sum_{1}^{3}(L_i - L_{gi})^2 \quad (10)$$

If we set H_i=$L_i{}^2$,the fitness function change to

$$F(x)= \sum_{1}^{3}(sqrt(H_i) - L_{gi})^2 \quad (11)$$

CHAPTER 3

GENETIC ALGORITHMS

3.1 Basic Method

A genetic algorithm (GA) is a search heuristic that mimics the process of natural evolution. This heuristic is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

The basic idea used in GA optimization is given in Table-3.1. Initially, a number of candidate solutions constitute a population. After each generation, the algorithm evaluates each individual according to its fitness and employs genetic operators to produce offspring from selected parents. The fitness function measures solution quality which is problem dependent. The offspring are added into the population while sometimes, least fit individuals are discarded. The process is repeated until the algorithm obtains a sufficiently good solution.

Table-3.1 Basic steps in GA

Algorithm of Genetic Algorithm

Initialize Population (P)
Evaluate fitness
while Not Termination do
for each Individual in P
do
1. Evaluate fitness
2. Select Parents
3. Apply Crossover and produce Offspring
4. Mutate the Offspring
end for
Update P
end while

The choice of the appropriate genetic operator is important as it directly influences the convergence of the genetic algorithm. However, different forms of the main genetic operators are needed according to the type of the genetic algorithm and the nature of the optimization problem. An overview of the main components of a genetic algorithm is discussed below.

1. Initialization: At the initialization stage, candidate solutions or individuals are randomly generated. The number of individuals in the population is determined according to the problem, and in many cases, empirically evaluated in trial experiments. In some cases, the candidate solutions are seeded in the area of search space where the desired solution is likely to be found.
2. Selection : During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a *fitness-based* process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as the latter process may be very time-consuming.
3. Reproduction using crossover: The main reproduction operators are crossover and mutation. The crossover operator exchanges genetic material from selected parents and forms either a single or multiple offspring.
4. Reproduction using mutation: The mutation operator provides random diversity in the population. This is important when the algorithm gets trapped in a local minimum.
5. Termination: This generational process is repeated until a termination condition has been reached. Common terminating conditions are (1). A solution is found that satisfies minimum criteria (2). Fixed number of generations has reached (3).The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results.

Genetic Algorithms are being used in several applications for arriving the optimum solutions. The objective may be either implicit or explicit function of the design variables. As the algorithms works on several individuals at time, there is a guaranteed optimum solution. There are several invariants such as micro GAs and real coded GAs, etc.

3.2 Application to Present Problem

The objective function defined in chapter 2 that is the sum square error is to be minimized. As it is a function of x,y and ϕ, we find a best solution by maximizing the corresponding fitness function which is inverse of the objective function. There are two solutions that give maximum fitness value at the end of all the cycles.

A computer program in MATLAB is utilized with the following objective function:

```
function obj=rpr(yv)
x=yv(1);y=yv(2);phi=yv(3);
Lg1=100;
Lg2=120;
Lg3=150;
L1S= (x^2 + y^2);
L1 =sqrt(L1S);
e1 =(L1-Lg1)^2;
L2S =(x+50*cos(phi)-200)^2+(y+50*sin(phi))^2;
L2 =sqrt(L2S);
e2=(L2-Lg2)^2;
L3S =(x+40*cos(phi)-40*sin(phi))^2+(y+40*sin(phi)+40*cos(phi)-200)^2;
L3= sqrt(L3S);
e3=(L3-Lg3)^2;
obj=e1+e2+e3;
```

CHAPTER 4

WORKSPACE AND JACOBIAN MATRIX

5.1 Workspace

Workspace is defined as regions which can be reached by a reference point C located on the mobile platform.

$$\rho_1^2 = [x - (x_p cos\emptyset - y_p sin\emptyset)]^2 + [y - (x_p sin\emptyset + y_p cos\emptyset)]^2 \qquad (12)$$

$$\rho_2^2 = [x - (x_p cos\emptyset - y_p sin\emptyset + t_4 cos\emptyset + t_1)]^2 + [y - (x_p sin\emptyset + y_p cos\emptyset - t4sin\emptyset]2 \qquad (13)$$

$$\rho_3^2 = \left[x - (x_p cos\emptyset - y_p sin\emptyset - t_5 cos\emptyset + .t_6 sin\emptyset + t_2)\right]^2 + [y - (x_p sin\emptyset + y_p cos\emptyset - t5sin\emptyset - .t6cos\emptyset + t3]2 \qquad (14)$$

These are three equation of circle . By plotting them and finding the area of their intersection gives the workspace of the 3RPR mechanism .

5.2 Jacobian matrix :

Jacobian matrix is used to establish a relation between generalized and actuator velocities as well as between generalized and actuator forces and couples. Formula for finding jacobian matrix is given as..

$$x^{\cdot} = J\theta^{\cdot} \qquad (15)$$

$$J = -B^{-1}A \qquad (16)$$

$$A = \begin{pmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{pmatrix} \qquad (17)$$

$$B = \begin{pmatrix} \rho 1 & 0 & 0 \\ 0 & \rho 2 & 0 \\ 0 & 0 & \rho 3 \end{pmatrix} \quad (18)$$

Where,

$$a_{11} = x \quad (19)$$

$$a_{12} = y \quad (20)$$

$$a_{13} = 0 \quad (21)$$

$$a_{21} = (x - c_2) + l_2 cos\emptyset \quad (22)$$

$$a_{22} = y + l_2 sin\emptyset \quad (23)$$

$$a_{23} = l_2(ycos\emptyset - (x - c_2)sin\emptyset) \quad (24)$$

$$a_{31} = (x - c_3) + l_3 \cos(\emptyset + \gamma) \quad (25)$$

$$a_{32} = (y - d_3) + l_3 \sin(\emptyset + \gamma) \quad (26)$$

$$a_{33} = l_3[(y - d_3)\cos(\emptyset + \gamma) - (x - c_3)\sin(\emptyset + \gamma)] \quad (27)$$

where ,

x,y are coordinates of the corner of platform w.r.t to fixed reference plane

Ø is an orientation angle of platform .

γ is an angle of platform .

l_2 and l_3 are the lengths of side of the mobile platform.

CHATER 5

RESULTS AND DISCUSSION

5.1 Methodology adopted

Fig.6 below shows the methodology adopted in present task.

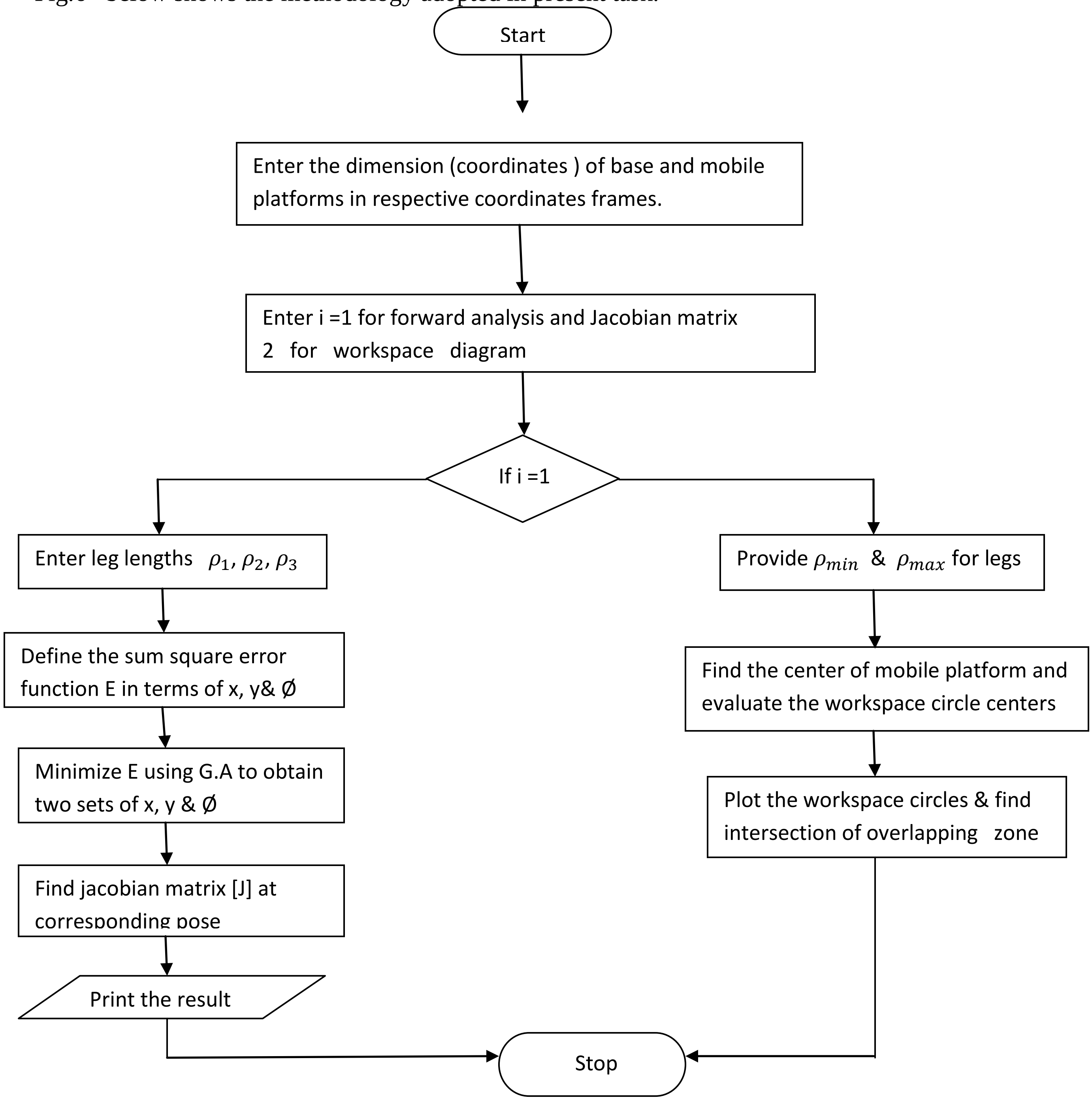


Figure 6 Methodology adopted for the project

5.2 Kinematic analysis of 3-RPR mechanism using genetic algorithm.

We first use the genetic algorithm using uniform crossover and mutation. The crossover rate is given 0.999 and mutation rate is given as 0.001. The high crossover rate ensures that maximum global search is achieved. We use one example for examining our genetic algorithm code. The 3-RPR mechanism base coordinates are given with respect to fixed reference frame and mobile platform coordinates are given with respect to moving reference plane. The leg lengths are taken as L: =[100, 120, 150]. Initialized the initial population size with real no 40. The three selected kinematics variables represent the end-effector position and orientation, being x, y and ϴ . Computation time is given in seconds.

The coordinates are given as

A_1 = (0, 0)

A_2 = (200, 0)

A_3 = (0, 200)

B_1 = (0, 0)

B_2 = (50, 0)

B_3 = (40, 40)

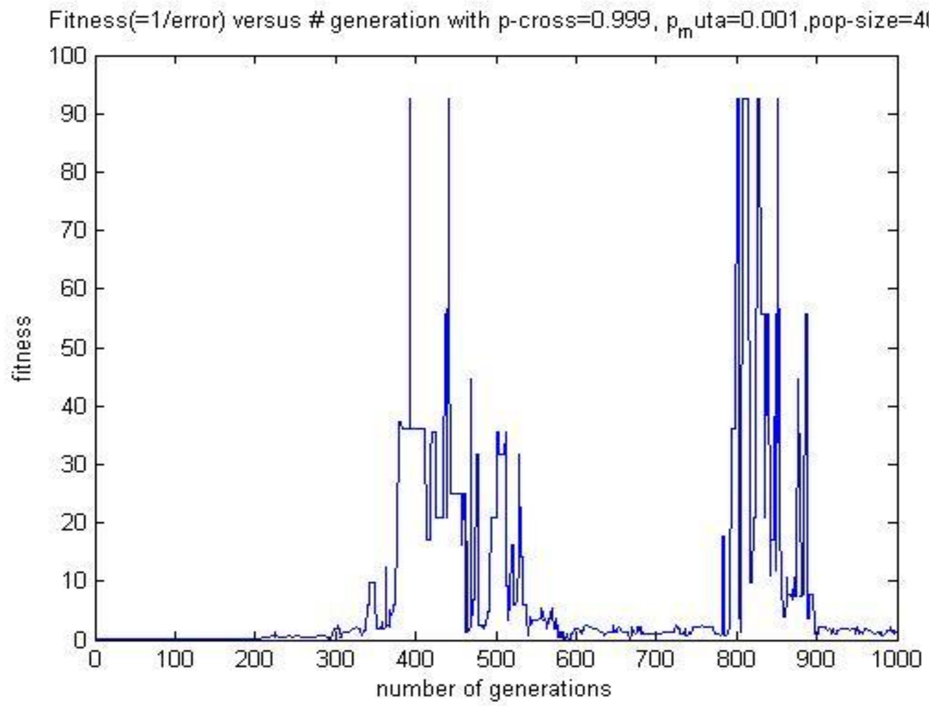
By genetic algorithm we have obtained these results from the function. Fig.7 shows the fitness variation with no of iterations

Generation 500 for G.A

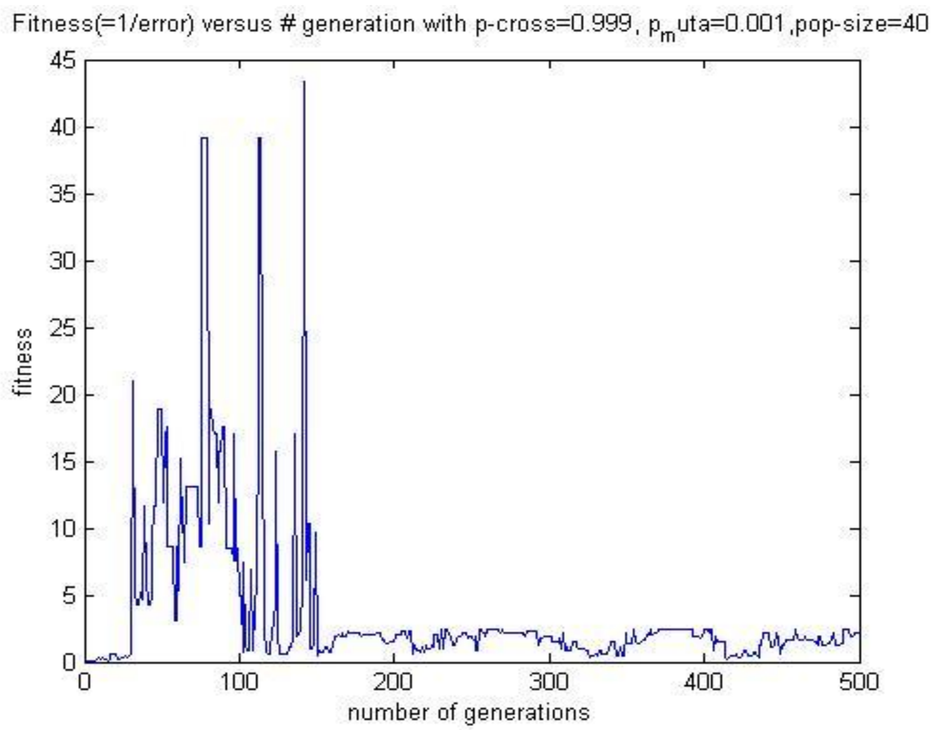
Variable	X	Y	ϴ
G.A.	97.849462	20.039101	84.1045
Algebraic method	97.99	19.91	85.01

Generation 1000 for G.A

Variable	X	Y	ϴ
G.A.	52.981427	85.141740	-33.7865
Algebraic method	52.86	84.95	-33.45



(a)



(b)

Fig 7.Graph of fitness vs no of generation with generation 500 and 1000 respectively

5.3 Jacobain matrix

3-R<u>P</u>R mechanism for which jacobian matrix has to be found out is as given below .The coordinates are

A_1= (0, 0)

A_2 = (200, 0)

A_3 = (0, 200)

B_1 = (0, 0)

B_2 = (50, 0)

B_3 = (40, 40)

coordinate of end effector B1 w.r.t to fixed reference plane (x=97.99 y=19.91) and angle of orientation Ø=85.02 degree and value of leg lengths are [100,120,150] .the jacobian matrix is as follows.

$$[J] = \begin{pmatrix} -0.855658 & 0.190268 & -0.0903346 \\ -0.811356 & -0.93643 & 0.444595 \\ -0.00522636 & -0.0127963 & -0.00770599 \end{pmatrix}$$

5.4 Workspace diagram

The workspace diagram for the above given planar parallel mechanism has been plotted with ρ_{min} = 80, ρ_{max} = 160 and orientation angle Ø= 15 degree. (Fig .8)

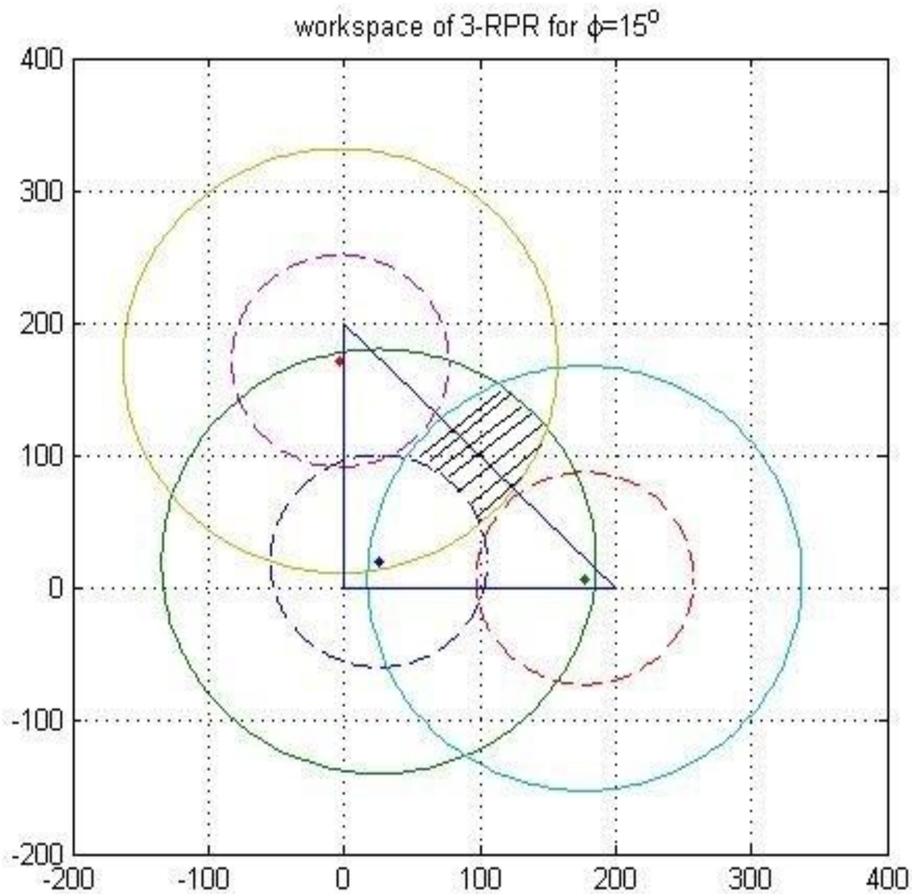


Fig. 8 Workspace for Ø=15 degree

Workspace circle for previous 3-RPR mechanism with orientation angle Ø=30 degree (Fig.9)

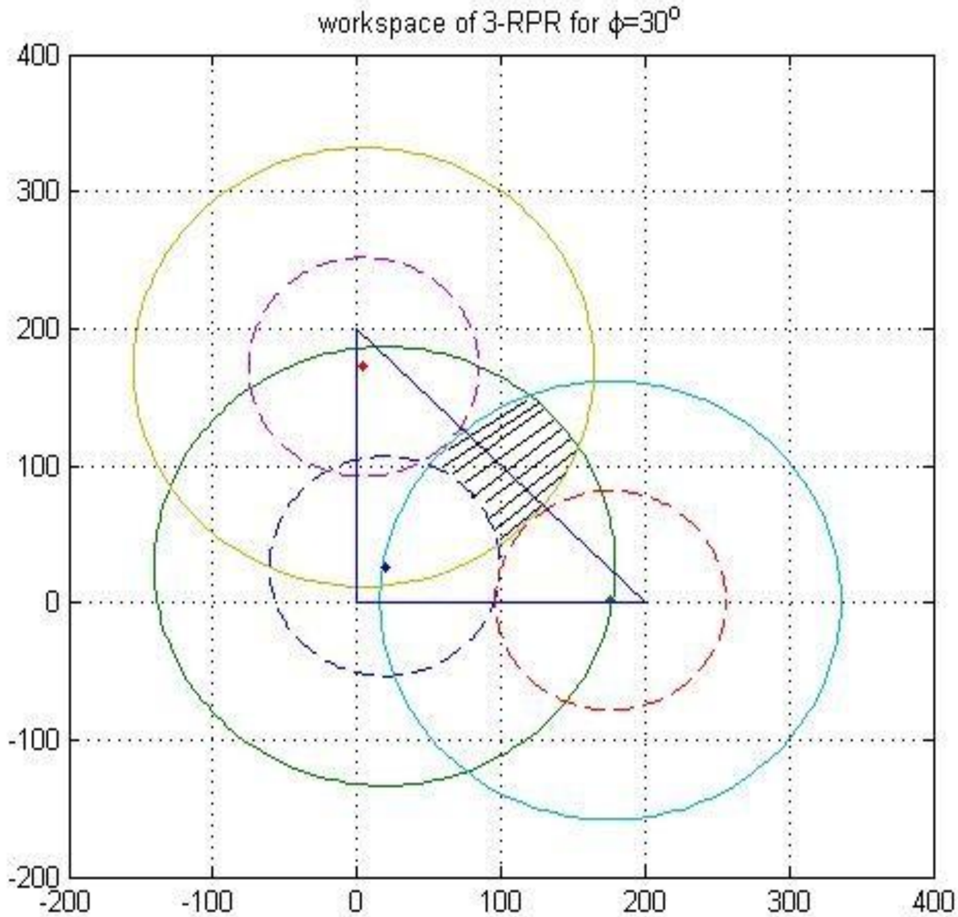


Fig .9 Workspace for Ø=30 degree

Workspace circle for previous 3-R<u>P</u>R mechanism with orientation angle Ø=45 degree. (Fig.10)

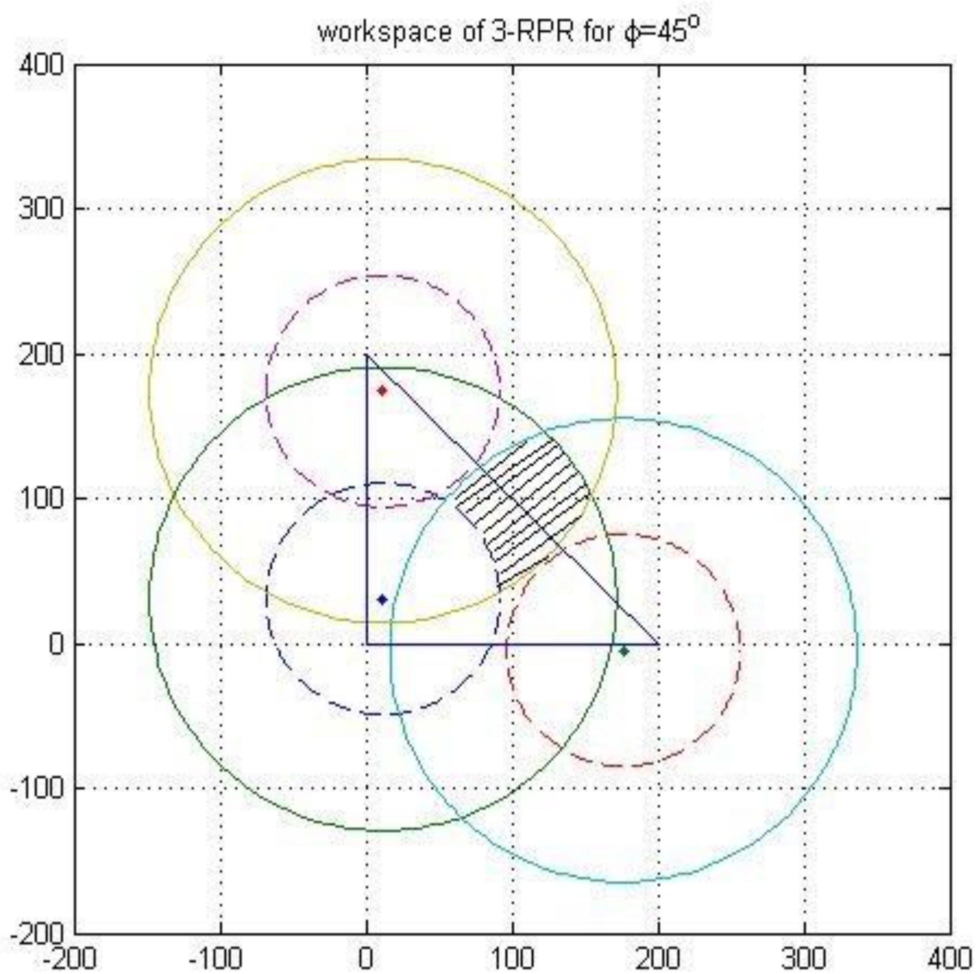


Fig. 10 Workspace for Ø=45 degree

Workspace circle for previous 3-RPR mechanism with orientation angle Ø=60 degree (Fig .11)

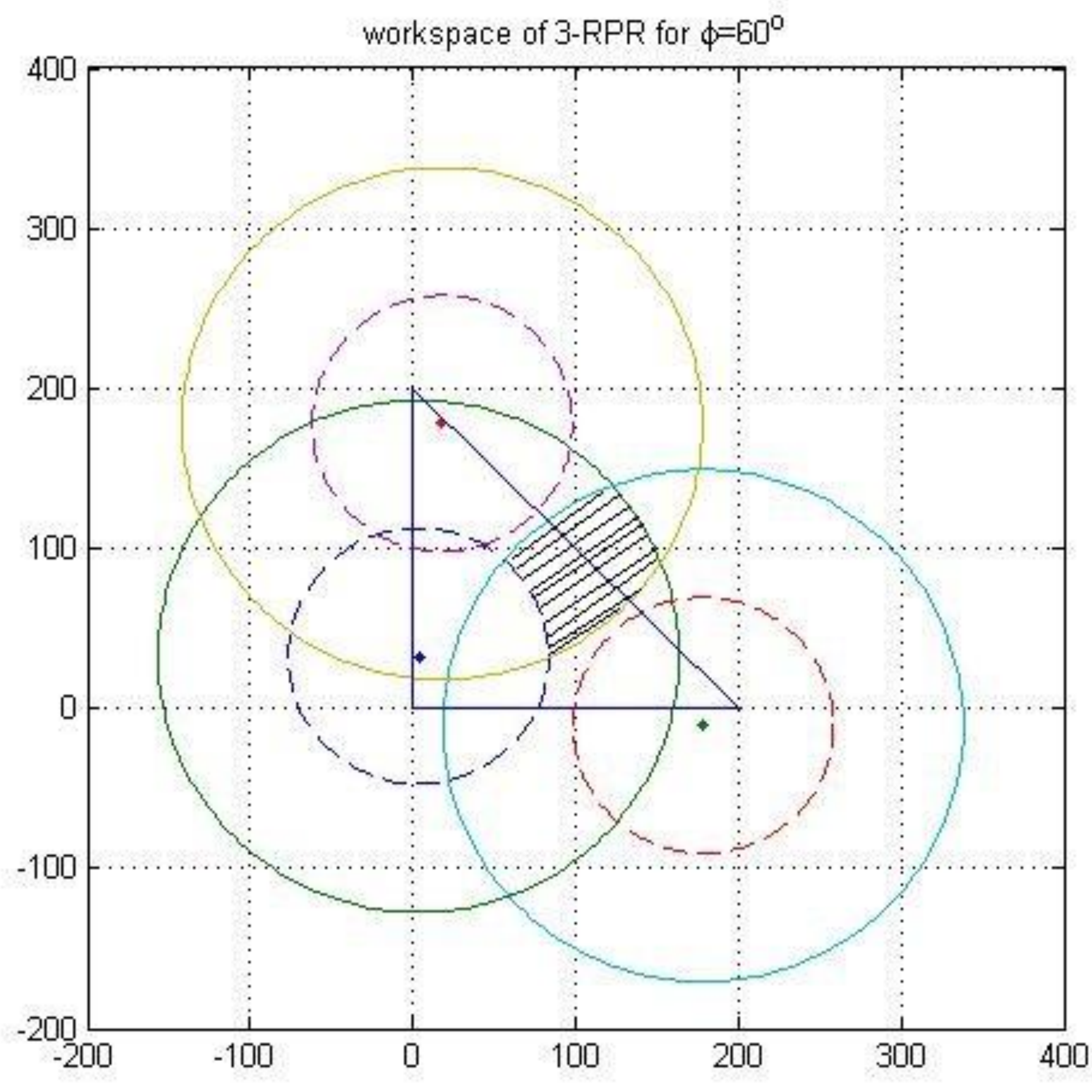


Fig 11. Workspace for Ø= 60 degree

CHAPTER 6

CONCLUSION

6.1 Summary of the work

In this project initially the forward kinematic solutions of 3-RPR mechanism are obtained as an optimization solution. The objective here is the distance between the connecting points of the limbs with platform as calculated from the moving coordinate frame and fixed reference frame, which is minimized so as to obtain the required pose of the platform corresponding to a known actuated lengths. When this distance (error) is zero for all the limbs, the manipulators achieve a possible configuration for desired position and orientation of mobile platform. The project employs genetic algorithms for obtaining multi-objective solution and neural network model for arriving the forward kinematic solution. In addition we have also plotted the workspace circle and jacobian matrix has been found for given 3-RPR mechanism.

REFRENCES

1. Marc Arsenault and Roger Boudreau, 'Synthesis of planar parallel mechanism considering workspace, dexterity stiffness and singularity avoidance.' , Journal of mechanical design , Volume -128, page no-69, 2006.
2. MingZ.Huang and Jean-Luc Thebert, 'A study of workspace and singularity characteristics for design of 3-DOF planar parallel robots.', International journal of advance manufacturing technology , vol-51,page no- 789,2010.
3. Qimi Jiang and Clement M.Gosselin , 'Geometric synthesis of a planar 3 RPR parallel mechanism for singularity free space.', Canadian Society for Mechanical Engineering vol-33, page no-667, 2009.
4. Stephane caro , Nicolas Binaud and Philippe Wenger, 'Sensitivity analysis of 3-RPR planar parallel manipulators', Journal of mechanical design., Vol-131, page no-121005-1 to 121005-13, 2009
5. Jaouad Sefrioui and Clement M.Gosselin, 'On the Quadratic Nature of the Singularity Curves of Planar Three-Degree-Of-Freedom Parallel Manipulators', Mechanism and Machine Theory, Vol -30, No-4, page no -533–551, 1994.
6. Marise Gallant, and Roger Boudreau, 'The Synthesis of Planar Parallel Manipulators with Prismatic Joints for An Optimal, Singularity-Free Workspace', Journal of Robotics Systems, Vol-19, pp. 13–24, 2001.
7. Rohitash Chandra,and Luc Rolland 'On solving the forward kinematics of 3RPR planar parallel manipulator using hybrid metaheuristics.' Applied Mathematics and Computation, vol -217, pp. 8997-9008,2011.
8. Serdar Kucuk, 'Simulation and design tool for performance analysis of planar parallel manipulators.' Transactions of the Society for Modeling and Simulation International ,pp 1-15 2011
9. P.Wenger and D.Chablat, 'Kinematic analysis of a class of analytic planar3-RPR parallel manipulators.' International Workshop on Computational Kinematics, 2009
10. http://en.wikipedia.org/wiki/mechanism
11. http://en.wikipedia.org/wiki/Genetic_algorithm

APPENDIX

COMPUTER PROGRAMS

A C++ program is written for finding the Jacobian matrix of 3-RPR mechanism

```
// program to find the jacobian matrix
# include<iostream.h>
# include<stdio.h>
# include<conio.h>
# include<math.h>
int main()
{
  float a[3][3],b[3][3],c[3][3],d[3][3],e[3][3],h[3][3];
  float x,y,n=0,q;
  int t1,t2,t3,i,j,k;
  float phi,gam,m;
  float d1,d2,d3,l1,l2,l3;
  cout<<"enter the value of coordinate of the mobile platform w.r.t to fixed platform";
  cin>>x>>y;
  cout<<"enter the coordinates of the base";
  cin>>t1>>t2>>t3;
  cout<<"enter the value of leg lenghts at given position";
  cin>>d1>>d2>>d3;
  cout<<"enter the dimension of platform";
  cin>>l1>>l2>>l3;
  cout<<"enter the value of angle of platform for given position";
  cin>>phi;
  m=(pow(l2,2)+pow(l3,2)-pow(l1,2))/(2*l2*l3);
  gam=acos(m);
  a[0][0]=x;
  a[0][1]=y;
  a[0][2]=0;
  a[1][0]=(x-t1)+l2*cos(phi);
  a[1][1]=y+l2*sin(phi);
  a[1][2]=l2*(y*cos(phi)-(x-t1)*sin(phi));
  a[2][0]=(x-t2)+l3*cos(phi+gam);
  a[2][1]=(y-t3)+l3*sin(phi+gam);
  a[2][2]=l3*((y-t3)*cos(phi+gam)-(x-t3)*sin(phi+gam));
  b[0][0]=d1;
  b[0][1]=0;
  b[0][2]=0;
  b[1][0]=0;
  b[1][1]=d2;
  b[1][2]=0;
  b[2][0]=0;
  b[2][1]=0;
  b[2][2]=d3;
   for(i=0;i<3;i++)
     {     cout<<endl;
         for(j=0;j<3;j++)
         {
             c[i][j]=0;
             d[i][j]=0;
```

```
            e[i][j]=0;
            h[i][j]=0;
        }
        }
//calculating the determinant of the matrix
for(i=0,j=0;j<3;j++)
        {
            if(j==2)
            n+=a[i][j]*a[i+1][0]*a[i+2][1];
            else if(j==1)
            n+=a[i][j]*a[i+1][j+1]*a[i+2][0];
            else
            n+=a[i][j]*a[i+1][j+1]*a[i+2][j+2];
        }
        for(i=2,j=0;j<3;j++)
        {
            if(j==2)
            n-=a[i][j]*a[i-1][0]*a[i-2][1];
            else if(j==1)
            n-=a[i][j]*a[i-1][j+1]*a[i-2][0];
            else
            n-=a[i][j]*a[i-1][j+1]*a[i-2][j+2];
        }

  if(n!=0)
  q=1.0/n;
   else
   {
        cout<<"Division by 0, not good!\n";
        cout<<"=========================================\n"<<endl;
        return 0;
   };
   // finding the inverse of matrix
   for(i=0;i<3;i++)
   {
        cout<<endl;
        for(j=0;j<3;j++)
        {

            d[i][j]=a[j][i];

        }
   }
   cout<<endl<<endl;


   c[0][0]=d[1][1]*d[2][2]-(d[2][1]*d[1][2]);
   c[0][1]=(-1)*(d[1][0]*d[2][2]-(d[2][0]*d[1][2]));
   c[0][2]=d[1][0]*d[2][1]-(d[2][0]*d[1][1]);

   c[1][0]=(-1)*(d[0][1]*d[2][2]-d[2][1]*d[0][2]);
   c[1][1]=d[0][0]*d[2][2]-d[2][0]*d[0][2];
   c[1][2]=(-1)*(d[0][0]*d[2][1]-d[2][0]*d[0][1]);

   c[2][0]=d[0][1]*d[1][2]-d[1][1]*d[0][2];
   c[2][1]=(-1)*(d[0][0]*d[1][2]-d[1][0]*d[0][2]);
```

```
    c[2][2]=d[0][0]*d[1][1]-d[1][0]*d[0][1];

    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            e[i][j]=c[i][j]*q;                 //inverse of the matrix

        }
    }
    // multiplication of two matrix
    for( i=0;i<3;i++)
    {
    for( j=0;j<3;j++)
    {
       h[i][j] = 0;
       for( k = 0 ;k < 3 ; k++)
          h[i][j]+= e[i][k]*b[k][j];
    }
    }
     for( i=0;i<3;i++)
    {
    for( j=0;j<3;j++)
    {
        h[i][j]=-1*h[i][j];
        }
    cout<<"\n========== The jacobian matrix is!!! ==========\n";
        for(i=0;i<3;i++)
        {     cout<<endl;
            for(j=0;j<3;j++)
            {
                cout<<" h["<<i<<"]["<<j<<"]= "<<h[i][j];

            }
        }
        cout<<endl<<endl;

   getch();
  return 0;
}
```

A MATLAB program is used for drawing workspace diagram

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
clear all;
t1=200;
t2=0;
t3=200;
t4=50;
t5=40;
t6=40;
     dmin=80;
     dmax=160;
     xb=[0 t1 t2 0];yb=[0 0 t3 0];


      % CENTROID OF THE MOBILE PLATFORM WITH MOVING COORDINATE FRAME
    x=(t4+t5)/3;
    y=t6/3;
    phi=60*pi/180;%.52381;
    x1=x*cos(phi)-y*sin(phi);
    y1=x*sin(phi)+y*cos(phi);
    %cout<<"\ncenter of first workspace circles are"<<x1<<" "<<y1;
    x2=x*cos(phi)-y*sin(phi)-t4*cos(phi)+t1;
    y2=x*sin(phi)+y*cos(phi)-t4*sin(phi);
    %cout<<"\ncenter of  second workspace circles are"<<x2<<" "<<y2;
    x3=x*cos(phi)-y*sin(phi)-t5*cos(phi)+t6*sin(phi)+t2;
    y3=x*sin(phi)+y*cos(phi)-t5*cos(phi)-t6*sin(phi)+t3;
    i=1;
    for t=0:pi/100:2*pi
   Xmin=dmin*cos(t);
   Ymin=dmin*sin(t);
   Xmax=dmax*cos(t);
   Ymax=dmax*sin(t);
   x1min(i)=Xmin+x1;y1min(i)=Ymin+y1;
   x2min(i)=Xmin+x2;y2min(i)=Ymin+y2;
   x3min(i)=Xmin+x3;y3min(i)=Ymin+y3;
   x1max(i)=Xmax+x1;y1max(i)=Ymax+y1;
   x2max(i)=Xmax+x2;y2max(i)=Ymax+y2;
    x3max(i)=Xmax+x3;y3max(i)=Ymax+y3;
   i=i+1;
 end

 plot(x1min,y1min,'--',x1max,y1max,x2min,y2min,'--',x2max,y2max,x3min,y3min,'--',x3max,y3max);
 hold on;
 plot(x1,y1,'.',x2,y2,'.',x3,y3,'.');
 hold on;
 plot(xb,yb,'b');
 grid on;
 title('workspace of 3-RPR for  \phi=60^o');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```