DESIGN AND ANALYSIS OF UNDERACTUATED COMPLIANT MECHANISMS

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

Bachelor of Technology

In

Mechanical Engineering

By

DEEPAK BEHERA (108ME077)

Department of Mechanical Engineering

National Institute Of Technology

Rourkela-769008

2011-12

DESIGN AND ANALYSIS OF UNDERACTUATED COMPLIANT MECHANISMS

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

Bachelor of Technology

In

Mechanical Engineering

By

DEEPAK BEHERA (108ME077)

Under The Guidance of

Prof. J. Srinivas

Department of Mechanical Engineering

National Institute Of Technology

Rourkela-769008

2011-12

National Institute Of Technology

Rourkela

CERTIFICATE

This is to certify that the thesis entitled, **“Design and Analysis of Underactuated compliant mechanisms”** submitted by **Deepak Behera (108ME077)** in partial fulfilment of the requirements for the award of **Bachelor of Technology Degree** in **Mechanical Engineering** at National Institute of Technology, Rourkela is an authentic work carried out by them under my supervision and guidance.

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

Date:

Prof. J.Srinivas
Dept. of Mechanical Engg.
National Institute of Technology
Rourkela-769008

ACKNOWLEDGEMENT

I wish to express deep sense of gratitude and indebtedness to **Prof. J. Srinivas,** Department of Mechanical Engineering, N.I.T Rourkela, for introducing the present topic and for his constant guidance, constructive criticism and valuable suggestion throughout this project work. Also I am thankful to my friends Pravat Kumar Sahoo, Mrutyunjay Lenka, Sukanta kumar Mangal and Mohan Charan Sethi for their support throughout this work.

Last but not the least, I express my profound gratitude to the Almighty and my parents for their blessings and support without which this task could have never been accomplished.

Date:

Deepak Behera
(108ME077)
B. Tech.
Mechanical Engg.
NIT Rourkela

ABSTRACT

Precession and accuracy are important in several mechanisms in practical use. Compliant mechanism provides a solution for such mechanical design problems. It has several other advantages over rigid body mechanism. That is why nowadays a lot of research work is going on in this field. But the main disadvantage of this kind of mechanism is its complexity to analyse and design. So, in order to make the analysis simpler pseudo rigid body model (PRBM) technique is often adopted. A mechanism is called underactuated when it has more degrees of freedom than number of inputs or actuations. For such mechanisms we have to perform the kinematic analysis along with force analysis to obtain the solutions. In this work, two underactuated partially compliant mechanisms have been discussed. In the first case, a partially compliant slider-crank mechanism without any input actuation is taken into account. The kinematic and static force equations are solved numerically to find out equilibrium position of the mechanism. Here, the input is provided by two torsional springs. The second case considered is a two-degree of freedom five bar slider mechanism with one input actuator only. Kinematic analysis along with force analysis using principle of virtual work is illustrated to find the solution. The nonlinear algebraic equations obtained are solved simultaneously based on Newton-Raphson method using a computer program in C and the graphs are plotted between input data and other parameters. In both the cases, input data is taken from references for comparison point-of-view.

Table of Contents

List of figures

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

1.1 Compliant Mechanisms

Compliant mechanisms are the mechanisms which get at least some of their mobility from the flexibility of their members (links) along with the movable pin joints or hinges. But in case of rigid body mechanism the mobility of the whole system is only due to the movable joints. As it has several advantages over rigid body mechanism large numbers of research works are going on in this field. Fig.1 shows a partially compliant slider-crank linkage [11].

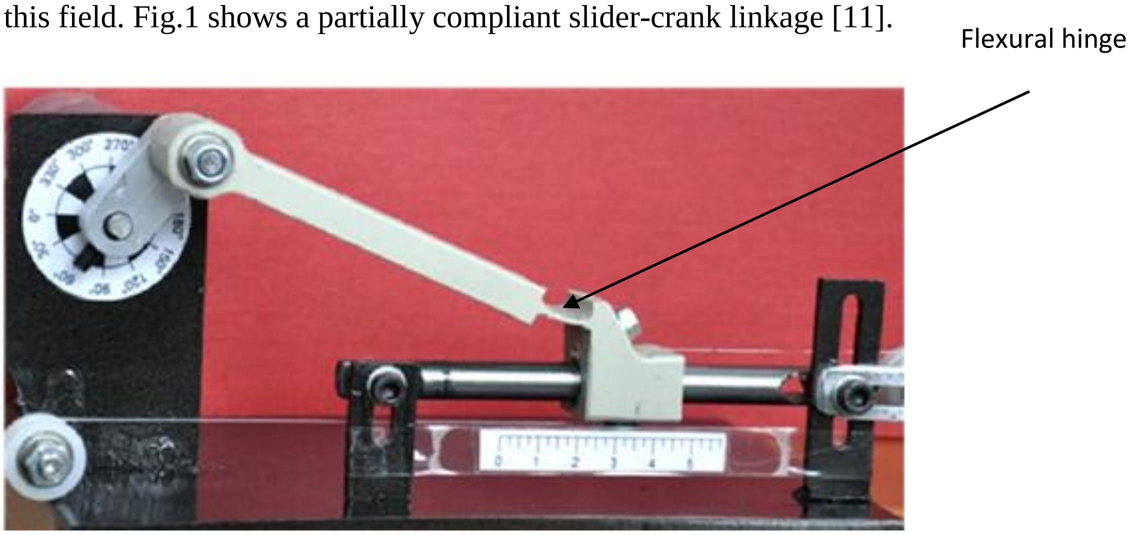

Fig.1.1 Partially compliant eccentric slider crank linkage

The main advantages of compliant mechanism over rigid body mechanism are less expensive, lighter in weight, low maintenance, high performance, and high precession. Compliant mechanisms also have less number of pin joints and sliding joints. This results in reduction in wear and tear and need of lubricants. The reduction in the number of joints also gives the mechanism high precession. Vibration and noise caused by the turning and sliding joints of rigid body mechanisms can also be eliminated up to some extent using compliant mechanism. It is possible to reduce the weight significantly by using a compliant mechanism over their rigid-body counter parts which is very significant in aerospace technology [13]. Compliant mechanisms gain its mobility from the deflection of flexure joints and the strain energy is stored in them.

Along with advantages, a compliant mechanism has several limitations. The most notable one is the complexity and design of such kind of mechanism. To simplify the analysis and design pseudo rigid body model (PRBM) technique has been often adopted. In pseudo rigid body model (PRBM) technique along with pin joints between links torsional springs are used to represent stiffness of flexural hinges.

1.2 Underactuation

A mechanism is called underactuated if the number of input to the mechanism is less than its degrees of freedom. It is also called underconstrained mechanism. This underactuation technique is used in robotics for grasping. Underactuated hands have fewer actuators than number of degrees of freedom, so can easily adapt to the shape of the object. Fig. 2 shows a two degrees of freedom underactuated mechanism [9]. This mechanism has one revolute actuator and one torsional spring.

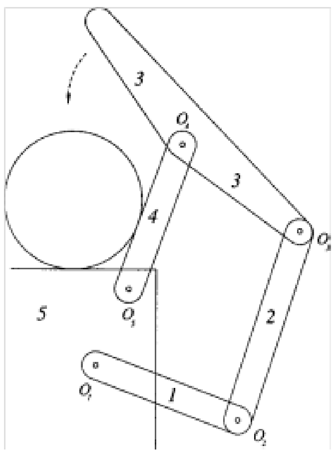

Fig 1. 2. Planar two degree of freedom underactuated mechanism

The simplest underactuated mechanism is with one degree of freedom and zero input. As degrees of freedom (DOF) and inputs increase, the complexity also increases. For such mechanisms we have to perform the kinematic analysis along with static force analysis, to get the analysis result.

1.3 Gripping action using underactuated mechanism

Under actuation technique is used for grasping of objects of various shape and size. For this purpose a finger like structure is manufactured to adapt its shape according to the object to be grasped. The finger and mechanical hand is controlled by reduced number of actuators. Its action is very similar to a human hand. Today microgrippers of micron size and high gripping speed are of prime importance in the field of robotics. They have wide ranges of application. For example they are used for precise positioning of needle and surgical instruments in the field of medical science [12]. Fig.3 shows a gripper design using slider crank mechanism [12].

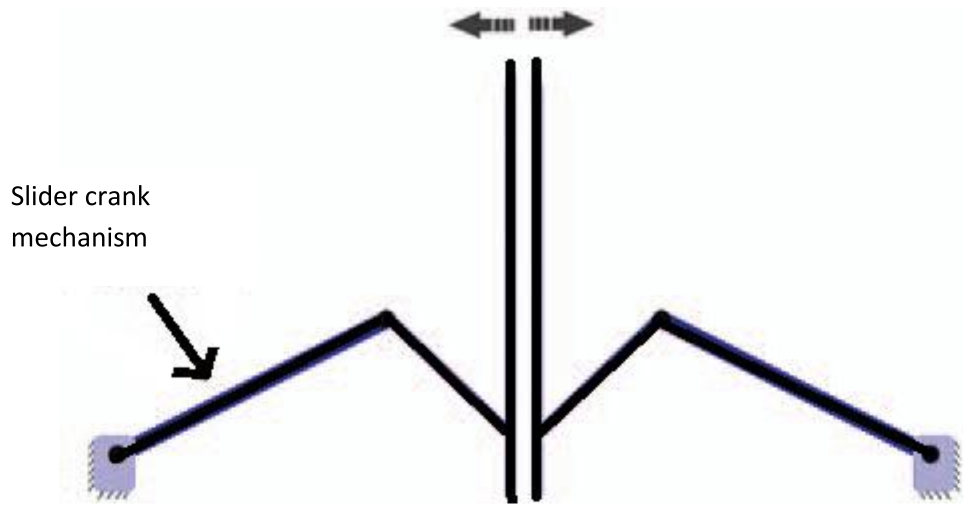

Fig 1.3.The schematic of employment of slider–crank mechanism in a gripper design

1.4 Literature Review

This section presents literature-review on various works related to underactuation, mechanism compliance and gripping applications in chronological order.

Aten *et al*. [1] proposed a numerical method for analysing pseudorigid-body models (PRBM) of underactuated compliant mechanisms. In this paper minimum potential energy method as well as virtual work principle is applied to find out the equilibrium position of the given system. A slider mechanism with one degree of freedom and zero input are considered for this purpose. It has taken the undeflected position of springs as initial condition and static along with kinematic analysis of the given mechanism are performed.

Tanik and Soylemez [2] illustrated analysis and design of an underactuated compliant mechanism using pseudo rigid body model. The paper considers the mechanism for two conditions (i) given output loading and (ii) constant input torque. It explains the capability of such mechanism to provide nearly constant output for wide range of input. To this mechanism, virtual work principle is applied which facilitates the static and kinematic analysis. The variable stroke mechanism described in this paper has one input (crank) and one output (slider) .but there are two degrees of freedom which accounts for its underactuated property. This paper presents design charts for given output loading and constant input torque.

Kragten and Herder [3] illustrated two metrics for measuring the performance of underactuated fingers to pick up objects and move them. These metrics have the capability to achieve stable grasp equilibrium of a wide range of moving objects and the ability to hold the grasped objects while disturbing forces are applied to it. The calculations and measurements of these metrics are shown for cable-pulley driven mechanical hands.

Tian *et al*.[4] proposed mechanical design and dynamics of a three degree of freedom flexure based parallel mechanism. in this paper it is explained how flexure hinges are used as the revolute joints to provide smooth motion with high accuracy . Finite element analysis method is used to verify the performance of the three degree of freedom flexural mechanism.

Zhang *et al*.[5] presented the solutions to deal with grasping force of underactuated hands which is normally difficult to control. This paper has proposed a novel underactuated hand based on linkages. The force output of the finger can be varied by using a motor. In this paper analysis of the finger has been analysed and designed. According to this a two degree of freedom and a three degree of freedom were designed which are included in a hand with comprises of five fingers and six motors with 15 DOFs.

Kragten and Herder [6] proposed the design and evaluation of the performance of compliant underactuated mechanical hands. The position of the objects relative to the hand can be varied to make the grasping operation successful. This paper considers planar symmetrical grasping operation, where objects can move along line of symmetry. Also an innovative approach to emulate frictionless grasping operation for circular objects is presented in this paper.

Cheng *et al*. [7] proposed an underactuated mechanism with one degree of freedom for finger operation. Design of this underactuated mechanism is based on spring elements of the structure. The finger mechanism is capable of human like grasping (gripping) operations. The fingers which perform gripping operation are controlled by reduced number of actuators. The performance and efficiency of the mechanism can be verified by kinematic as well as dynamic analysis.

Erkaya *et al*.[8] presented the kinematic and dynamic analysis of a modified slider–crank mechanism with an additional eccentric link between connecting rod and crank pin, as distinct from a conventional mechanism. The extra link is called eccentric connector, which transmit gas force to the crank. A planetary gear is used to transmit driving forces to the output. The planetary gear is also driven by the eccentric connector with the help of a pinion. As a result the driving force is transmitted to the crankshaft by two different ways. The dynamic analysis result of the proposed slider crank mechanism is examined and compared with conventional slider crank mechanism. it is observed that the proposed slider crank mechanism has greater output torque as compared to the conventional one, although both work with same gas pressure and same stroke.

Montambault and Gosselin [9] proposed the concept of underactuation applied to gripper mechanism. The main objective of this work is kinematic analysis for design of the underactuated mechanism. So, a kinetostatic model is proposed which involves both kinematic as well as static force analysis. In this paper an underactuated gripper based on a five bar mechanism withtwo degrees of freedom is studied to verify its feasibility.

Daoud *et al*. [10] presented a global strategy for object manipulation with the fingertips with an anthropomorphic dexterous (for a two year old humanoid) hand. Fine manipulation with the fingertips requires computing on one hand, finger motions able to produce the desired object motion and on the other hand, it is necessary to ensure object stability with a real time scheme for the fingertip force computation.

1.5 Objective of present work

In recent days underactuated compliant mechanism has attracted attention of many researchers due to its growing application in various fields. In this work two examples of underactuated compliant mechanism have been discussed. For the design and analysis of such kind of mechanism, the pseudo rigid body model (PRBM) for each case is considered. This makes the kinematic and static force analysis much easier. After that virtual work principle is applied to it to obtain a set of nonlinear equations. In this work the nonlinear equations are solved numerically using Netwon-Raphson method in both the examples. In the first example (slider-crank mechanism) Newton Raphson method is applied directly. But in the second case it has to be solved by considering Taylor series expansion method. To obtain desired results for both the examples computer programs in C are used. The obtained data are plotted to to show the variation (increasing / decreasing trend) between these parameters.

Following are the main objectives of the present work:

1. Define compliant mechanism under underactuation.

2. Analysis of an underactactuated mechanism using virtual work method to obtain the nonlinear equations.
3. Solve the equations obtained to get the solutions and find out the relationship among them.

CHAPTER 2

MATHEMATICAL MODELLING

2. MATHEMATICAL MODELING

This chapter presents the mathematical equations used in obtaining the solutions for the two underactuated linkages considered.

2.1 Single Slider Underactuated Mechanism

Here an underactuated four bar compliant mechanism is taken into account. This mechanism has one degree of freedom and no input is provided. Two sprigs provide initial actuation in the system. On applying virtual work method a nonlinear equation can be obtained which describes equilibrium position of the mechanism. From kinematic analysis other two equations are obtained. Newton Raphson method is used to find out accurate values of equilibrium position for different initial conditions. Some other parameters like link size, spring deflection per unit revolution (in rad) are taken same as that of published data [1]. This is implemented with the help of a computer program in C .Values for equilibrium position are obtained from given set of assumed value and for different initial conditions.

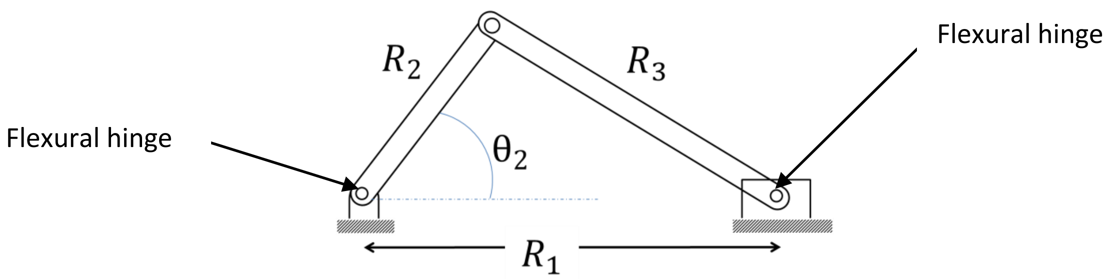

Fig. 2.1An underactuated mechanism with single degree of freedom

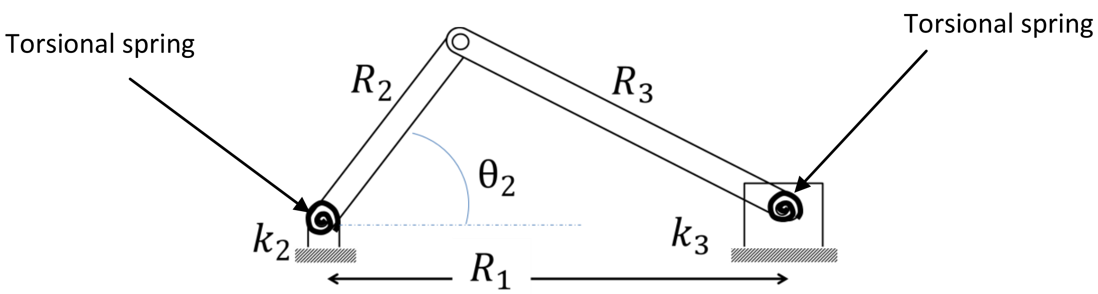

Fig. 2.2 Pseudorigid body model for the underactuated partial compliant mechanism

Fig. 2.1 shows a single DOF, slider crank mechanism with two flexure hinges. Corresponding PRBM is shown in Fig. 2.2.

From static equilibrium or principle of virtual work, we can write

$$\frac{R_2 cos\Theta_2}{R_3 cos\Theta_3} k_3(\Theta_3 - \Theta_{30}) - k_2(\Theta_{2-}\Theta_{20}) = 0 \tag{1}$$

Also from geometry,

$$R_2 cos\Theta_2 + R_3 cos\Theta_3 = r_1 \tag{2}$$

$$R_2 sin\Theta_2 + R_3 sin\Theta_3 = 0 \tag{3}$$

From equation (3) we obtain, $\Theta_3 = \sin^{-1}(-\frac{R_2 sin\Theta_2}{R_3})$ (4)

$$Now\ let\ \sin^{-1}(-\frac{R_2 sin\Theta_2}{R_3}) = \Theta_2{}' \tag{5}$$

Putting the values of $\Theta_3\ in$ (1),

$$\frac{R_2 cos\Theta_2}{R_3 cos(\Theta_2{}')} k_3(\Theta_2{}' - \Theta_{30}) - k_2(\Theta_{2-}\Theta_{20}) = 0 \tag{6}$$

We have used Newton-Raphson method for solving this equation. It uses:

$$x_1 = x_0 - \frac{f(x_0)}{f\prime(x_0)} \tag{7}$$

Where

$x_0 = assumed\ value,\ \ x_1 = value\ obtained\ after\ one\ iteration$

$$f(x_0) = \frac{Rcos\Theta_2}{R_3 \cos(\Theta_2')} k_3(\Theta_2{}' - \Theta_{30}) - k_2(\Theta_{2-}\Theta_{20}) \tag{8}$$

$$f'^{(x_0)} = first\ derivative\ of\ f(x_0)\ w.r.t.\Theta_2$$

2.2 Five bar underactuated slider mechanism

Here a five bar underactuated compliant mechanism has been analysed and discussed. It has two degrees of freedom with only one input. This particular condition makes it underactuated. PRBM is employed in this case, which enable us to perform the kinematic analysis as well as force analysis. In this mechanism input is provided across crank and output is obtained across the slider. Certain assumptions have taken into account like mass of the links are negligible and mechanism is operating at very slow speed. The virtual work principle is taken into account which is nothing but to giving some virtual angular displacement and equating the net work done equal to zero. In this method the springs between links are removed and torques which are equal in magnitude and opposite in direction are employed at those points. From kinematic analysis we have obtained two basic equations. Other equations are obtained by force analysis. A five bar underactuated mechanism and its pseudo-rigid body model are presented in Fig. 2.3 and Fig.2.4 respectively.

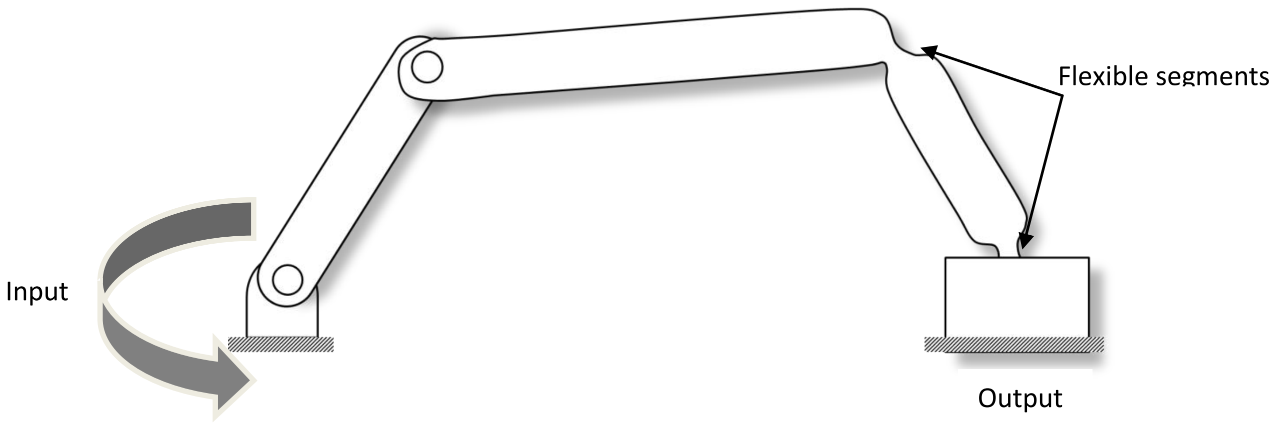

Fig. 2.3 a five bar underactuated compliant mechanism

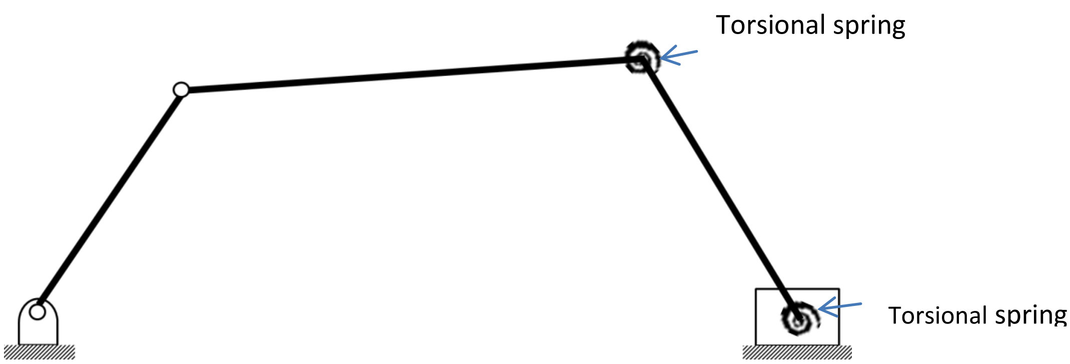

Fig. 2.4 Pseudo-rigid body model applied to an underactuated compliant mechanism

This is a case of variable stroke compliant mechanism with two inputs. There are two equations from kinematics and two equations from force equilibrium. From 6 unknowns two can be assumed. First these equations are explained with reference to FBD given in Fig 2.5.

.

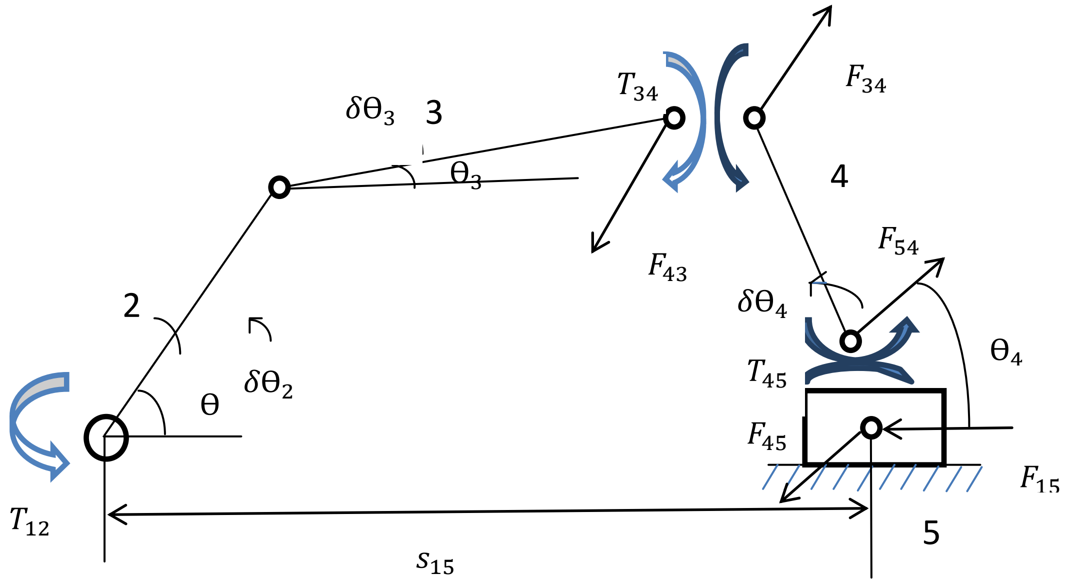

Fig. 2.5 Virtual work principle applied to a Variable stroke compliant mechanism

Now from this free body diagram we have to obtain the required equations by performing kinematic as well as dynamic analysis.

KINEMATICS ANALYSIS:

By resolving in X and y direction we obtain –

$a_2\cos\Theta_2+a_3\cos\Theta_3-a_4\cos\Theta_4=s_{15}$ (9)

$a_2\sin\Theta_2+a_3\sin\Theta_3-a_4\sin\Theta_4=0$ (10)

DYNAMICS ANALYSIS:-

Applying virtual work principle to active forces we can write:

$\partial W= \partial W1+\partial W2+\partial W3+\partial W4$ (11)

$\partial W=0;$ (12)

So,$-T_{12}\partial\Theta_2-F_{15}\partial s_{15}+T_{34}(\partial\Theta_4-\partial\Theta_3)+T_{45}\partial\Theta_4=0$ (13)

Replacing $\partial\Theta_3$ and $\partial\Theta_4$with the following pairs of equations [2],

$$\partial\Theta_3 = \frac{a_2 \sin(\Theta2-\Theta4)\partial\Theta2+\cos(\Theta4)\partial s15}{a_3\sin(\Theta4-\Theta3)} \tag{14}$$

$$\partial\Theta_4 - = \frac{a_2 \sin(\Theta2-\Theta3)\partial\Theta2+\cos(\Theta3)\partial s15}{a_4\sin(\Theta4-\Theta3)} \tag{15}$$

Substituting (6) and (7)in equation (5) ,

$$-T_{12}\partial\Theta_2 - F_{15}\partial s_{15} + T_{34}$$

$$\left(\frac{a_2 \sin(\Theta2-\Theta3)\partial\Theta2+\cos(\Theta3)\partial s15}{a_4\sin(\Theta4-\Theta3)} - \frac{a_2 \sin(\Theta2-\Theta4)\partial\Theta2+\cos(\Theta4)\partial s15}{a_3\sin(\Theta4-\Theta3)}\right)+T_{45}\frac{a2 \sin(\Theta2-\Theta3)\partial\Theta2+\cos(\Theta3)\partial s15}{a_4\sin(\Theta4-\Theta3)} = 0$$

(16)

$$-T_{12}\partial\Theta_2 - F_{15}\partial s_{15} + T_{34}\left(\frac{a_2a_3 \sin(\Theta_2-\Theta_3)\partial\Theta_2+a_3 cos\Theta_3\partial s_{15}}{a_3a_4\sin(\Theta_4-\Theta_3)} - \frac{a_2a_4 \sin(\Theta_2-\Theta_4)\partial\Theta_2+a_4 cos\Theta_3\partial s_{15}}{a_3a_4\sin(\Theta_4-\Theta_3)}\right)$$

$$+T_{45}\left(\frac{a2 \sin(\Theta2-\Theta3)\partial\Theta2+\cos(\Theta3)\partial s15}{a_4\sin(\Theta4-\Theta3)}\right) = 0 \tag{17}$$

Considering the terms of $\delta\theta2$ and δs_{15} in (8) and (9)

$$-T_{12} + T_{34}\left(\frac{a_2 a3 \sin(\Theta2-\Theta3) - a_2 a4 \sin(\Theta2-\Theta4)}{a_3a_4\sin(\Theta4-\Theta3)}\right)+T_{45}\left(\frac{a_2 \sin(\Theta2-\Theta3)}{a_4\sin(\Theta4-\Theta3)}\right) = 0 \tag{18}$$

Equation (11) can be written as,

$$T_{34}\left(\frac{a_2a_3 \sin(\Theta2-\Theta3) - a_2a_4 \sin(\Theta2-\Theta4)}{a_3a_4\sin(\Theta4-\Theta3)}\right)+T_{45}\left(\frac{a_2 \sin(\Theta2-\Theta3)}{a_4\sin(\Theta4-\Theta3)}\right)=T_{12} \tag{19}$$

$$-F_{15} + T_{34}\frac{a_3 \cos(\Theta3) - a4\cos(\Theta4)}{a_3a_4\sin(\Theta4-\Theta3)}+T_{45}\left(\frac{\cos(\Theta3)}{a_4\sin(\Theta4-\Theta3)}\right)=0 \tag{20}$$

Where $T_{34}= k_{34}$ (Θ_3 -Θ_4+C_{34}) and T_{45}=k_{45} (-Θ_4+C_{45})

k_{ij}andC_{ij} represent the spring stiffness spring initial position constant between ith and jth link respectively

Now equations (9),(10),(19) and(20) are to be solved simultaneously by Newton Raphson method.

As these equations are nonlinear we have to apply Taylor series expansion to solve these equations. Now let us consider the equation obtained earlier to which Taylor series expansion method has to be applied.

$$f=a_2\cos\Theta_2+a_3\cos\Theta_3-a_4\cos\Theta_4 - s_{15} \tag{21}$$

$g = a_2\sin\Theta_2 + a_3\sin\Theta_3 - a_4\sin\Theta_4$ (22)

$m = T_{34}\ \left(\frac{a_2 a3\sin(\Theta 2-\Theta 3) - a_2 a_4\sin(\Theta 2-\Theta 4)}{a_3 a_4 \sin(\Theta 4-\Theta 3)}\right) + T_{45}\ \left(\frac{a_2\sin(\Theta 2-\Theta 3)}{a_4\sin(\Theta 4-\Theta 3)}\right) - T_{12}$ (23)

$n = T_{34}\frac{a_3\cos(\Theta 3) - a_4\cos(\Theta 4)}{a_3 a a_4 \sin(\Theta 4-\Theta 3)} + T_{45}\left(\frac{\cos(\Theta 3)}{a_4\sin(\Theta 4-\Theta 3)}\right) - F_{15}$ (24)

As f= 0, (25)

$f_0 + \frac{\partial f}{\partial \Theta_3}\Delta\Theta_3 + \frac{\partial f}{\partial \Theta_4}\Delta\Theta_4 + \frac{\partial f}{\partial s_{15}}\Delta s_{15} + \frac{\partial f}{\partial T_{12}}\Delta T_{12} = 0$ (26)

g=0 so, $g_0 + \frac{\partial g}{\partial \Theta_3}\Delta\Theta_3 + \frac{\partial f}{\partial \Theta_4}\Delta\Theta_4 + \frac{\partial g}{\partial s_{15}}\Delta s_{15} + \frac{\partial g}{\partial T_{12}}\Delta T_{12} = 0$ (27)

m=0 so, $m_0 + \frac{\partial m}{\partial \Theta_3}\Delta\Theta_3 + \frac{\partial m}{\partial \Theta_4}\Delta\Theta_4 + \frac{\partial m}{\partial s_{15}}\Delta s_{15} + \frac{\partial m}{\partial T_{12}}\Delta T_{12} = 0$ (28)

n=0 so, $n_0 + \frac{\partial n}{\partial \Theta_3}\Delta\Theta_3 + \frac{\partial n}{\partial \Theta_4}\Delta\Theta_4 + \frac{\partial n}{\partial s_{15}}\Delta s_{15} + \frac{\partial n}{\partial T_{12}}\Delta T_{12} = 0$ (29)

The matrix is in the form, B + AΔX=0 (30)

$$\Delta X = -A^{-1}B \quad (31)$$

And, $X = X_0 + \Delta X$ (32)

X_0 will be incremented by ΔX for every iteration.

Matrix [A](4x4) will be calculate by using first order differential for f, g, m and n for every single iteration. f, g, m and n and the first order differential for f, g, m and n can be calculated by taking the input set of values.

Components of the matrix [A](4x4) are given below.

$A_{11} = \frac{\partial f}{\partial \Theta_3} = -a_3\sin\Theta_3$ (33)

$A_{12} = \frac{\partial f}{\partial \Theta_4} = a_4\sin\Theta_4$ (34)

$A_{13} = \frac{\partial f}{\partial T_{12}} = -1$ (35)

$A_{14} = \frac{\partial f}{\partial s_{15}} = 0$ (36)

$A_{21} = \frac{\partial g}{\partial \Theta_3} = a_3\cos\Theta_3$ (37)

$$A_{22}=\frac{\partial g}{\partial \Theta_4}=-a_4\sin\Theta_4 \quad (38)$$

$$A_{23}=\frac{\partial f}{\partial T_{12}}=0 \quad (39)$$

$$A_{24}=\frac{\partial f}{\partial s_{15}}=0 \quad (40)$$

$$A_{31}=\frac{\partial m}{\partial \Theta 3}=\frac{T34a_2(\sin(\Theta_2-\Theta_3)\cos(\Theta_4-\Theta_3)-\sin(\Theta_4-\Theta_3)\cos(\Theta_2-\Theta_3)+\sin(\Theta_2-\Theta_4)\cos(\Theta_4-\Theta_3))}{a_3\sin^2(\Theta_4-\Theta_3)}+$$

$$\{\frac{T_{45}a2(\cos(\Theta_4-\Theta_3)\sin(\Theta_2-\Theta_3)-\sin(\Theta_4-\Theta_3)\cos(\Theta_2-\Theta_3))}{a_4\sin^2(\Theta_4-\Theta_3)}\} \quad (41)$$

$$A_{32}=\frac{\partial m}{\partial \Theta 4}=\{\frac{T_{34}a_2\{(-\sin(\Theta_2-\Theta_3)\cos(\Theta_4-\Theta_3))+\sin(\Theta_4-\Theta_3)\cos(\Theta_2-\Theta_4)+\sin(\Theta_2-\Theta_4)\cos(\Theta_4-\Theta_3)\}}{a_3\sin^2(\Theta_4-\Theta_3)}\}+$$

$$\{\frac{T_{45}a_2(\cos(\Theta_4-\Theta_3)\sin(\Theta_2-\Theta_3))}{a_4\sin^2(\Theta_4-\Theta_3)}\} \quad (42)$$

$$A_{33}=\frac{\partial m}{\partial s_{15}}=0 \quad (43)$$

$$A_{34}=\frac{\partial m}{\partial T_{12}}=-1 \quad (44)$$

$$A_{41}=\frac{\partial n}{\partial \Theta 3}=\frac{T34\cos\Theta_4(-\sin\Theta_3\sin(\Theta_4-\Theta_3)+\cos\Theta_3\cos(\Theta_4-\Theta_3))}{a_3\sin^2(\Theta_4-\Theta_3)}+$$

$$\{\frac{T_{45}(-\sin\Theta_3\sin(\Theta_4-\Theta_3)+\cos\Theta_3\cos(\Theta_4-\Theta_3))}{a_4\sin^2(\Theta_4-\Theta_3)}\} \quad (45)$$

$$A_{42}=\frac{\partial n}{\partial \Theta 4}=\{\frac{(T34\cos\Theta_3(-\sin\Theta_4\sin(\Theta_4-\Theta_3))-\cos\Theta_4\cos(\Theta_4-\Theta_3))}{a_3\sin^2(\Theta_4-\Theta_3)}\}-\{\frac{T_{45}(\cos\Theta_3\cos(\Theta_4-\Theta_3))}{a_4\sin^2(\Theta_4-\Theta_3)}\} \quad (46)$$

$$A_{43}=\frac{\partial n}{\partial T_{12}}=0 \quad (47)$$

$$A_{44}=\frac{\partial n}{\partial s_{15}}=0 \quad (48)$$

Taking the above values the matrix [A] is formed. With the help of computer program in the inverse of matrix [A] is calculated for each iteration. The initial values for T_{12}, s_{15} are taken same as that of published data. Values for F_{15} for various values of Θ_2 are taken in a file from the output load function [2]. It will take random values for Θ_3 and Θ_4 from the given values of $\Theta_2$in the C program.Here we have assumed the values forΘ_3,Θ_4,T_{12}, s_{15}.These values will be used to calculate functional values for f,g, m and n. subsequently we will get actual values of the parameters Θ_3 ,Θ_4,T_{12}, s_{15} for each set of values of Θ_2 , F_{15} after certain number of iterations.

As discussed earlier the equations are obtained by kinematic as well as force analysis. These equations can be solved by numerical methods. Here Newton-Raphson method is used to solve all those equations. But these equations are nonlinear in nature. Now this can be solved with the help of a computer program in C. In the mentioned program the various values for the parameters used in the equations are taken as that of published data. Using Newton-Raphson method we are getting more accurate value after some iteration.

CHAPTER 3

RESULT AND DISCUSSION

3. RESULT AND DESCUSSION

Two case studies of under actuated mechanisms having sliders are illustrated in this chapter.

3.1 Slider crank Underactuated Mechanism with no input

For the single slider underactuated compliant mechanism the following published data are taken for various initial positions.

Link lengths

R_2 =16.6, R_3 = 8.3

Spring constants

k_2=16.46, k_3=49.39

Andθ_{20}, θ_{30} are initial angular positions of the two springs, which are taken as shown in Table-1 for different cases. Various conditions for initial position of the mechanism are taken into account to find out errors associated with different values of Θ_2. Table-1 compares the values of present data with published data for one given condition. Along with that data obtained for other cases are also tabled.

Table-1 Output results of equilibrium solution for various initial conditions

S No.	Initial angular position	θ_2		θ_3		R_1		No. of iterations
		Present data	Published data[1]	Present data	Published data[1]	Present data	Published data[1]	
1	$\theta_{20} = \pi/2$ $\theta_{30} = 0$	0.124380	0.1177	-0.250738	-0.2371	24.512218	24.5531	13
2	$\theta_{20} = \pi/3$ $\theta_{30} = 0$	0.081542	-	-0.163632	-	24.733973	-	9
3	$\theta_{20} = \pi/3$ $\theta_{30} = \pi/4$	-0.272437	-	0.568252	-	22.983356	-	19
4	$\theta_{20} = \pi/4$ $\theta_{30} = \pi/4$	-0.354070	-	0.766247	-	21.550594	-	34
5	$\theta_{\theta 20} = \pi/4$ $\theta_{30} = \pi/3$	-0.253908	-	0.526346	-	23.244353	-	17
6	$\theta_{20} = \pi/3$ $\theta_{30} = \pi/4$	-0.331378	-	0.708495	-	21.999415	-	32

Fig 3.1 represents the error convergence for one of the cases using Newton-Raphson solution.

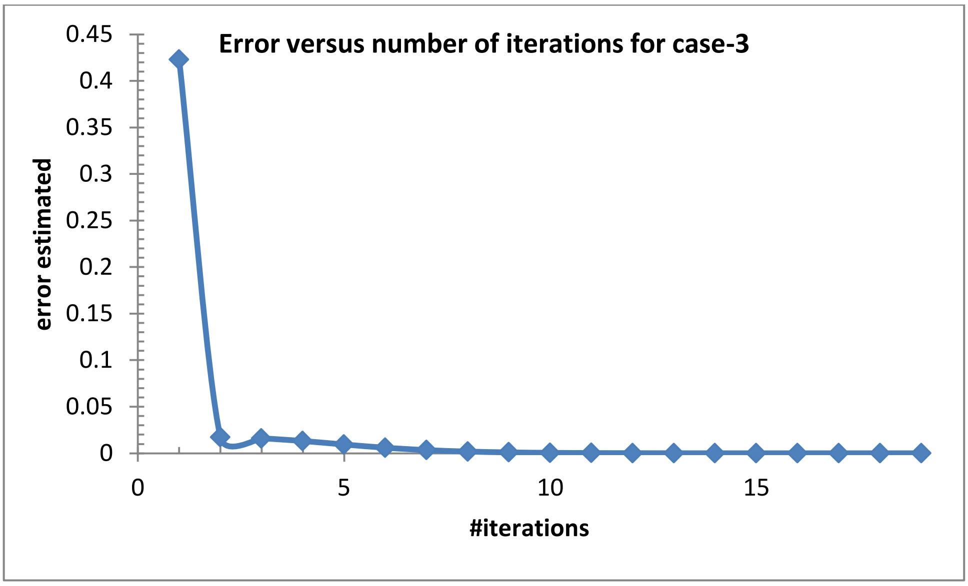

Fig.3.1 Variation between no. of iteration and error

3.2 Five bar underactuated partially compliant mechanism

In this work data from the output load function is taken according to the table below.

Table 2 Discrete data considered for analysis

Input crank angle(θ_2) (deg)	0	40	80	120	160	200	240	280	320	360
Slider force ($F15$)	80	-40	-40	-40	-40	200	200	200	200	80

Above table is constructed based on the following force-angle relation. The maximum value of the output load (F_{15}=200 N) during the work stroke (180°<θ_2<360°) is assumed to be five times that of the return stroke's maximum value. This is also shown in Fig.3.2.

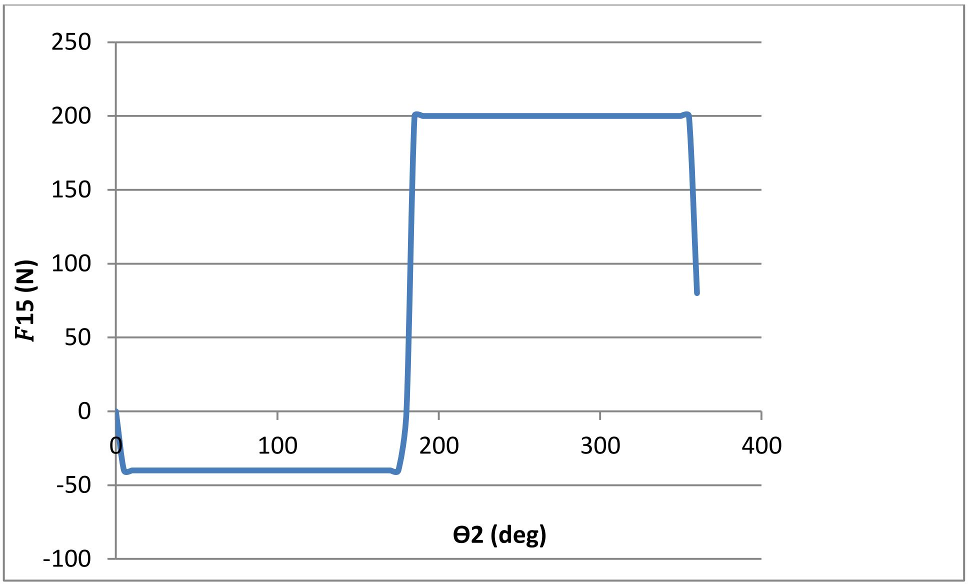

Fig. 3.2 Output load function

Other values taken as input parameters are:

Link lengths

a_2=1

a_3=3

a_4=1

Spring initial position constant for two nearby links

C_{34}= 2.618

C_{45}= 2.618

Linear spring stiffness

k_{34}= 100

k_{45}= 100

The nonlinear equations are numerically by using compute program in ’C’. As a result a set of values forθ_3 , θ_4,s_{15} $and T_{12}$ are obtained against set of θ_2 and F_{15}.The values obtained are plotted against the corresponding values of θ_2. Four graphs are obtained as below (Fig.3.3 to Fig.3.6).

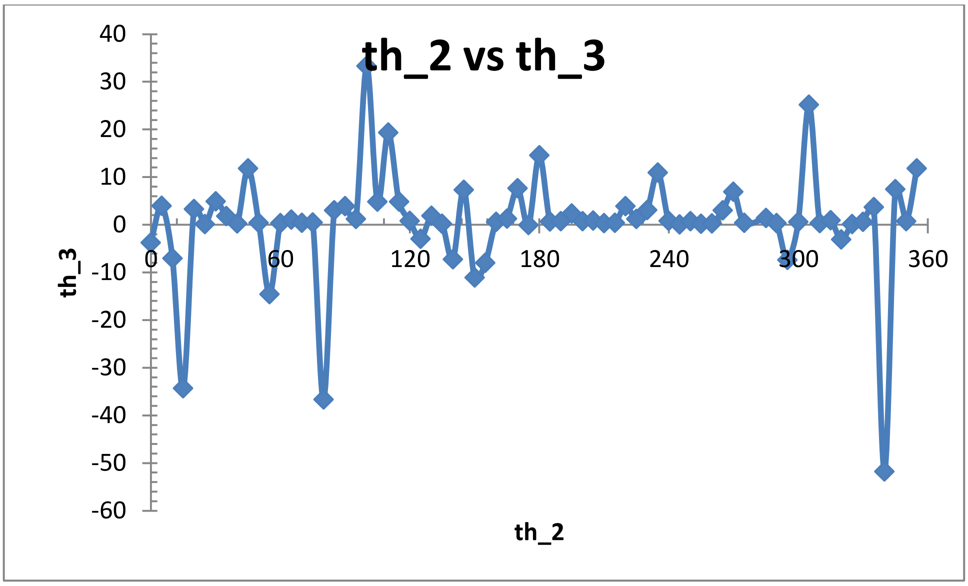

Fig. 3.3 ($Variation\ between\ \Theta_2 and \Theta_3$)

As shown in Fig. 3.3 for the forward stroke ($0 < \theta_2 < 180$) value of Θ_3 is decreasing initially then some increase can be observed. For the return stroke ($180 < \theta_2 < 360$) the exactly reverse can be noticed. In this stroke only the peak value in the negative side can be observed.

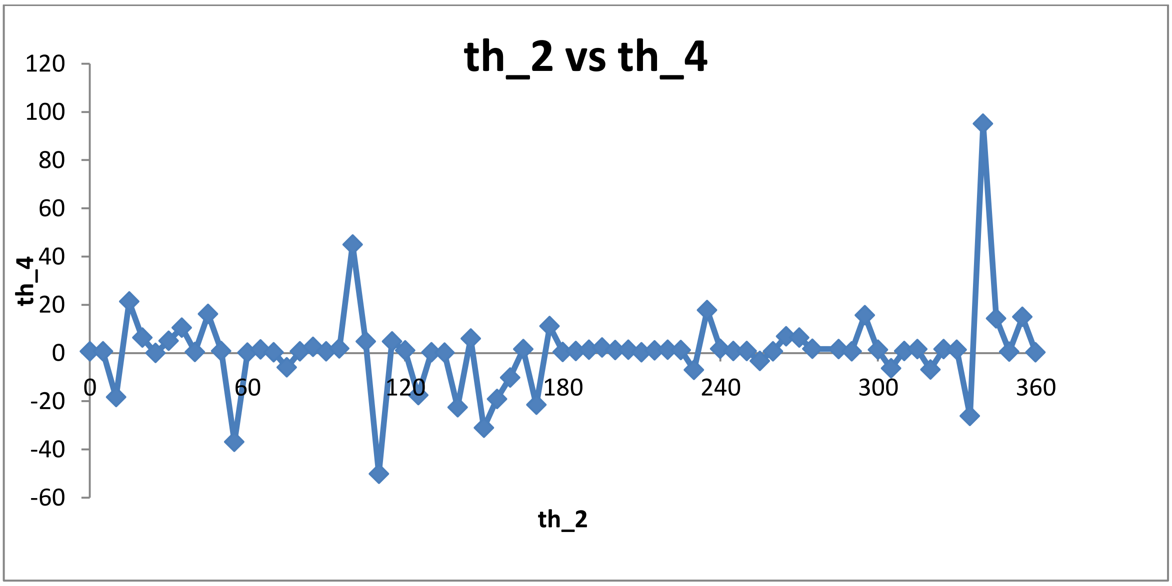

Fig. 3.4 $Variation\ between\ \Theta_2 and \Theta_4$

In this case the peak value for Θ_4 in the positive direction occurs at Θ_2=340, in the return stroke which is shown in Fig. 3.4. in the forward stroke relatively lower peaks are obtained.

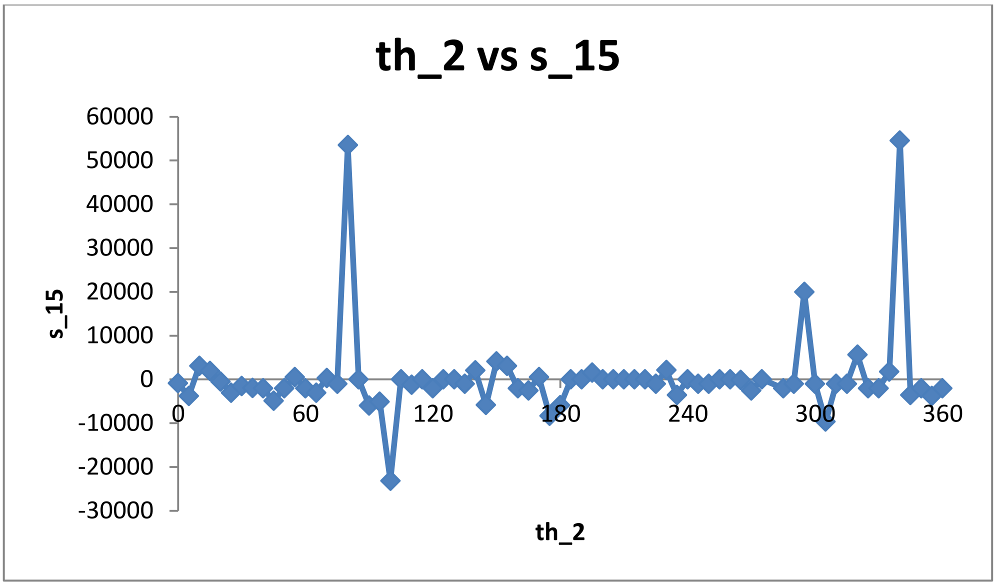

Fig. 3.5 $Variation\ between\ \Theta_2\ and\ \ s_{15}$

Fig. 3.5 explains the variation of s_{15}withΘ_2. Two peaks are observed one in the forward stroke and one in return stroke for Θ_2=80 and 340 respectively.

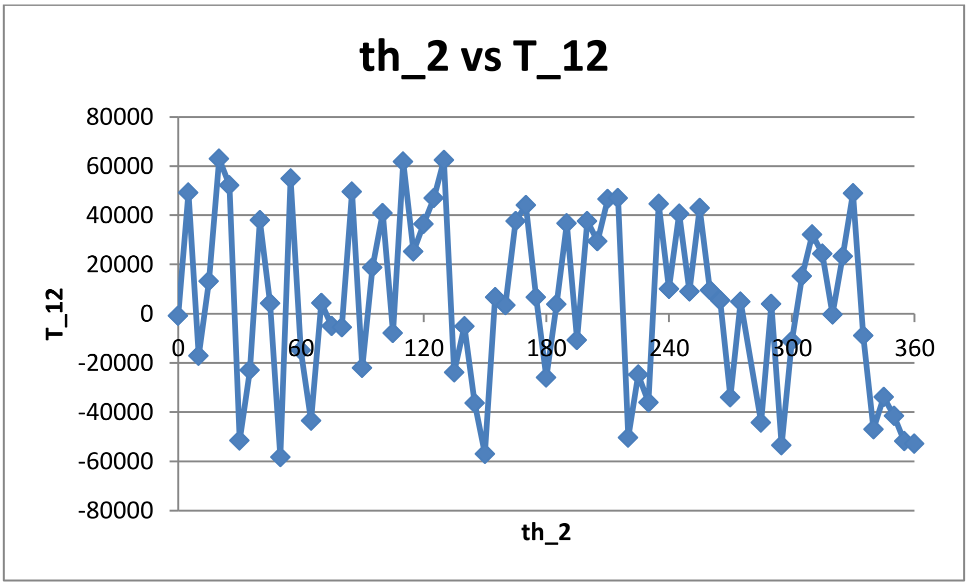

Fig. 3.6 $Variation\ between\ \Theta_2 and T_{12}$

From fig 3.6 it is found out that torque (T_{12}) is fluctuating constantly both for forward stroke ($0 < \theta_2 < 180$) as well as return stroke ($180 < \theta_2 < 360$). But it is observed that the fluctuation is less in return stroke.

CHAPTER 4

CONCLUSIONS

4. CONCLUSIONS

4.1. Summary of the work

In the present work we have considered two examples of partial underactuated mechanism. We tried to analyse these two examples and solve it numerically. Kinematic and static force analyses are presented for both the discussed examples. The nonlinear equations obtained are solved numerically using Newton Raphson method. For the slider crank mechanism values of unknown variables are obtained for different initial conditions. Also the effectiveness of Newton-Raphson method is shown with the help of a graph. In the five bar compliant mechanism for each set of assumed values, the values for unknown variables are obtained.

4.2Future scope

An underactuated compliant mechanism can be analysed and solved more accurately by employing any other numerical method. More complex underactuated mechanism with lot more degrees of freedom can also be analysed and solved. Fabrication of the 2-DOF underactuated linkage discussed in this work is going-on and its performance with flexure hinges to grasp the objects has to be tested.

REFERENCES

1. Q. Aten ,S. A. Zirbel , B.D Jensen and L .L. Howell, ‘A Numerical Method for Position Analysis of Compliant Mechanisms With More Degrees of Freedom Than Inputs’,Journal of Mechanical Design, Trans.ASME, 133 (2011), 061009-1
2. E. Tanik and E. Soylemez, ‘Analysis and design of a compliant variable stroke mechanism’, Mechanism and Machine Theory, 45 (2010), 1385-1394.
3. G.A.Kragten and J.L.Herder, ‘The ability of underactuated hands to grasp and hold objects’, Mechanism and Machine Theory 45 (2010), 408–425.
4. Y. Tian, B. Shirinzadeh and D. Zhang,’ Design and dynamics of a 3-DOF flexure-based parallel mechanism for micro / nano manipulation’Microelectronic Engineering 87 (2010), 230–241.
5. W. Zhang, D. Zhao, Q. Chen and D. Du,’ Linkage under-actuated humanoid robotic hand with control of grasping force’,2nd International Asia Conference on Informatics in Control, Automation and Robotics, IEEE, 417-420, 2010.
6. G. A. Kragten and J. L. Herder,‘A Platform for Grasp Performance Assessment in Compliant or Underactuated Hands’ Trans.’ ASME, 132 (2010),024502-1
7. W.LiCheng,G. Carbone and M.Ceccarelli, ‘Designing an underactuated mechanism for a 1 active DOF finger operation’, Mechanism and Machine Theory 44 (2009), 336–348.
8. S. Erkaya, S. Su and I. Uzmay, ‘Dynamic analysis of a slider–crank mechanism with eccentric connector and planetary gears’, Mechanism and Machine Theory 42 (2007), 393–408.
9. S.Montambault and C. M.Gosselin, ‘Analysis of Underactuated Mechanical Grippers’, Journal of Mechanical Design, ASME Trans. 123 (2001), 367-374.
10. N. Daoud, J.P. Gazeau, S. Zeghloul and M. Arsicault, ‘A real-time strategy for dexterous manipulation: Fingertips motion planning, force sensing and grasp stability’, Robotics and Autonomous Systems, 60 (2012), 377–386.
11. E. Tanik, 'Transmission angle in compliant slider-crank mechanism’, Mechanism and Machine Theory,46 (2011),1623-1632.
12. M.S. Hajhashemi, F. Barazandeh, S. N. Nejad and R Nadafi DB, ’Design and micro fabrication of a constant-force microgripper’, Part C: J. Mechanical Engineering Science, 225 (2011), 2739-2748.
13. L.L. Howell, Compliant Mechanisms, John Wiley and Sons, New York, 2001.

APPENDIX

COMPUTER PROGRAMS(C-language) FOR SOLVING NONLINEAR COUPLED ALGEBRAIC EQUATIONS

(a) Single Slider Crank Mechanism

```
// THIS PROGRAM IS FOR SINGLE SLIDER CRANK MECHANISM WITH ZERO INPUT ACTUATION
// INPUT IS PROVIDED BY SPRINGS AT TWO JOINTS
#include<stdio.h>
#include<math.h>
#include<conio.h>
#define pi 3.1415926
#define limit 1e-10
main()
{
	double c,n_th2,x,f,df_th2,diff,b,R1,R2,R3,th3,th2,th20,th30;
	R2=16.6;R3=8.3;
	double k2=16.46,k3=49.39;
	FILE *fp,*fp1;
	fp=fopen("input.dat","r");
	while(!feof(fp))
	{
		fscanf(fp,"%lf %lf",&th20,&th30);
		inti=0;th2=0;
		fp1=fopen("result.txt","a+");
		fprintf(fp1,"\nFor th20=%lf & th30=%lf ::",th20,th30);
		do
		{
			th3=asin((-R2/R3)*sin(th2));
			printf("\n At %d Iteration theta-3 is:: %lf",i+1,th3);
			c= 1-(((R2*R2)/(R3*R3))*sin(th2)*sin(th2));
			b=pow(c,0.5);
			printf("\nb= %lf",b);
			R1=(R2*cos(th2))+(R3*cos(th3));
			f=((R2*cos(th2)/R3*cos(th3))*k3*(th3-th30))-(k2*(th2-th20));
			printf("\nf= %lf",f);
			x=((R2*R3*b*cos(th3)*sin(th2))+(R2*R2*cos(th2)*cos(th2)*sin(th3))+(k3*R2*R2*
									cos(th2)*cos(th2)));
			df_th2= (-x/(b*R3*R3*cos(th3)*cos(th3)))-k2;
			n_th2=th2-(f/df_th2);
			printf("\n At %d Iteration theta-2 is:: %lf",i+1,n_th2);
			if(n_th2>th2)
				diff=n_th2-th2;
			else
				diff=th2-n_th2;
			i++;
			fprintf(fp1,"\n%lf",diff);
			th2=n_th2;
		}while(diff>limit);
		printf("\n\n\n The Value of th3= %lf & r1= %lf ",th3,R1);
		fclose(fp1);
	}
	fclose(fp);
	return 0;}
```

Fig.A1 shows the flowchart of this program:

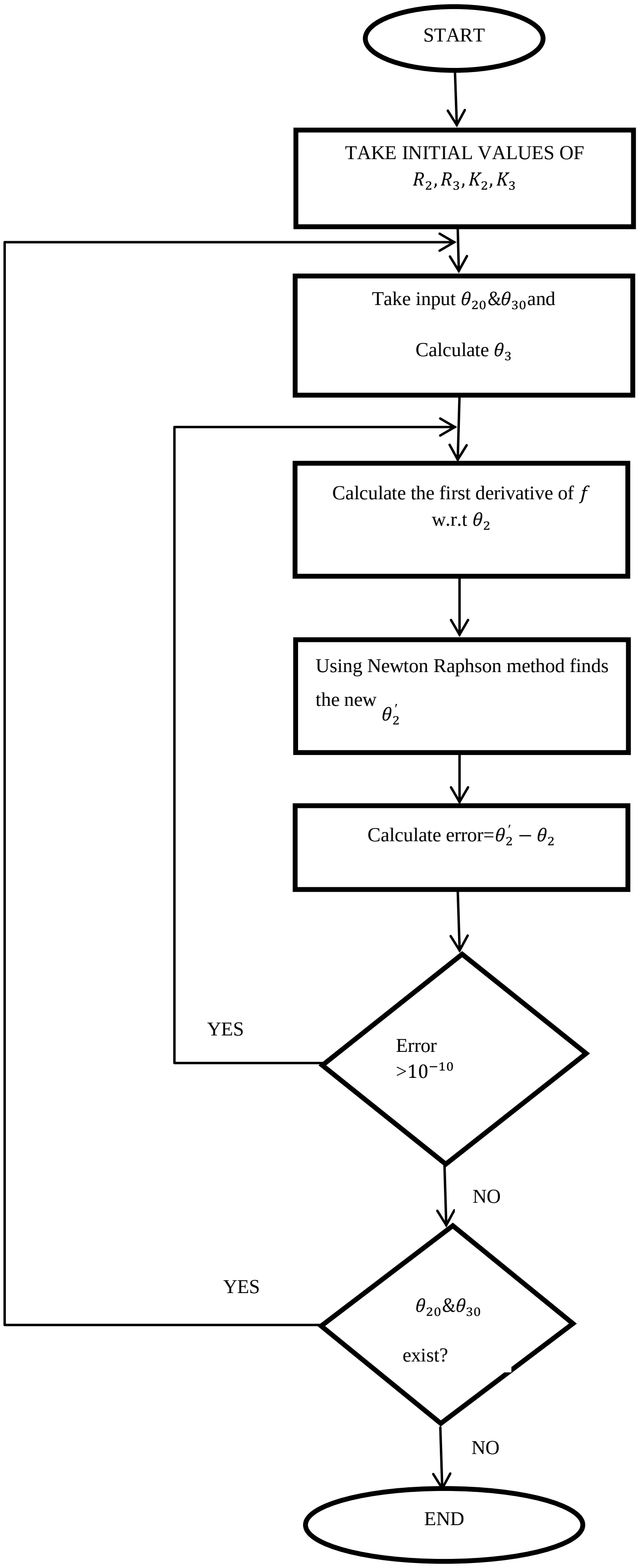

Fig.A1 Single equation solution by N-R method

(b) Two-Degree of Freedom Slider crank linkage

```
// SECOND PROGRAM IS FOR 2-DOF SLIDER CRANK LINKAGE, WHERE THERE ARE FOUR
// EQUATIONS AND SIX UNKNOWNS. COUPLED EQUATIONS ARE TO BE SOLVED BY EXTENDED
//  NEWTON RAPSON METHOD WHOSE FLOWCHART IS GIVEN IN FIG.A1.

#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<math.h>
#include<alloc.h>
#define pi 22/7
#define max 10
int m1,n1;
// calculate the cofactor of element (row,col)
intGetMinor(float **src, float **dest, int row, int col, int order)
{
	// indicate which col and row is being copied to dest
	intcolCount=0,rowCount=0;

	for(inti = 0; i< order; i++ )
	{
		if(i != row )
 {
	colCount = 0;
				for(int j = 0; j < order; j++ )
				{
					// when j is not the element
					if( j != col )
					{
						dest[rowCount][colCount] = src[i][j];
						colCount++;
					}
				}
				rowCount++;
		}
	}
	return 1;
}

// Calculate the determinant recursively.
doubleCalcDeterminant( float **mat, int order)
{
	// order must be >= 0
	// stop the recursion when matrix is a single element
	if( order == 1 )
		return mat[0][0];

	// the determinant value
	floatdet = 0;

	// allocate the cofactor matrix
	float **minor;
	minor = new float*[order-1];
	for(inti=0;i<order-1;i++)
		minor[i] = new float[order-1];
```

```
        for(i = 0; i< order; i++ )
         {
                // get minor of element (0,i)
                GetMinor( mat, minor, 0, i , order);
                // the recusion is here!

                det += (i%2==1?-1.0:1.0) * mat[0][i] * CalcDeterminant(minor,order-1);
                //det += pow( -1.0, i ) * mat[0][i] * CalcDeterminant( minor,order-1 );
         }

         // release memory
        for(i=0;i<order-1;i++)
                delete [] minor[i];
        delete [] minor;

        returndet;
}

// matrix inversioon
// the result is put in Y
voidMatrixInversion(float **A, float **Y)
{
int order=4;
         // get the determinant of a
        doubledet = 1.0/CalcDeterminant(A,order);

         // memory allocation
        float *temp = new float[(order-1)*(order-1)];
        float **minor = new float*[order-1];
        for(inti=0;i<order-1;i++)
                minor[i] = temp+(i*(order-1));

        for(int j=0;j<order;j++)
         {
                for(inti=0;i<order;i++)
                 {
                                // get the co-factor (matrix) of A(j,i)
                                GetMinor(A,minor,j,i,order);
                                Y[i][j] = det*CalcDeterminant(minor,order-1);
                                if( (i+j)%2 == 1)
                                        Y[i][j] = -Y[i][j];
                 }
         }

         // release memory
         //delete [] minor[0];
        delete [] temp;
        delete [] minor;
}
//input A,B output is Z(globally defined) row1 & col1 are dimension of A Matrix whereas row2 & col2 are of B
voidmatrixMultiplication(float **A,float **B,float **Z)
{
        inti,c,d,k,sum=0;

        for ( c = 0 ; c < 4 ; c++ )
                {
                        for ( d = 0 ; d < 1 ; d++ )
                        {
                                for ( k = 0 ; k < 4 ; k++ )
```

```
                {
                        sum = sum + A[c][k]*B[k][d];
                }

                Z[c][d] = sum;
                //printf("\n%f",Z[c][d]);
                sum = 0;
            }
        }
}
main()
{
        intdim,j,i;
        float **A;
        float **Y;
        float **B,**Z;
        dim=4;
        A=new float *[dim];
        Y=new float *[dim];
        for(i=0;i<dim;i++)
        {
            A[i] = new float[dim];
            Y[i] = new float[dim];
        }
        B=new float *[dim];
        for(i=0;i<dim;i++)
            B[i]=new float[1];

        Z=new float *[4];
        for(i=0;i<4;i++)
        {
            Z[i] = new float[1];
        }
        float f,g,m,n,th2d,F15,th2,th3,th4,k34,k45,T34,T45,c34,c45,s15,T12,a2,a3,a4;
        c34=2.618;
        k34=100;
        k45=100;
        c45=2.618;
        a2=1;
        a4=1;
        a3=3;
        FILE *fp,*fp1;
        fp1=fopen("output.txt","w");
        fprintf(fp1,"f\t\tg\t\tm\t\tn\t\tth3\t\tth4\t\ts15\t\tT12");
        fp=fopen("deep.dat","r");
        while (!feof(fp))
        {
            fscanf(fp,"%f %f\n",&th2d,&F15);
            //printf("%f %f\n",th2d,F15);
            th2=th2d*pi/180;
            int random=rand()%100+1;
            th3=pi*(double)random/180;
            int random1;
            do
            {
                random1=rand()%100+1;
                th4=pi*(double)random1/180;   //checking if theta 3 & theta 4 are having
                                                   same values!!
            }while(th3==th4);
            s15=0.2;
```

```
T12=0;
T34=k34*(th3-th4+c34);
T45=k45*(-th4+c45);
for(i=0;i<10;i++)
{
	f=a2*cos(th2)+a3*cos(th3)-a4*cos(th4)-s15;
	B[0][0]=f;
	printf("\n\nf= %f",f);
	g=a2*sin(th2)+a3*sin(th3)-a4*sin(th4);
	B[1][0]=g;
	printf("\ng= %f",g);
	m=(T34*a2*((sin(th2-th3))-(sin(th2-th4)))/(a3*sin(th4-
			th3)))+((T45*sin(th2-th3)*a2)/(a4*sin(th4-th3)))-T12;
	B[2][0]=m;
	printf("\nm= %f",m);
	n=((T34*cos(th3)*cos(th4))/(a3*sin(th4-th3)))+(T45*cos(th3)
						/(a4*sin(th4-th3)))-F15;
	B[3][0]=n;
	printf("\nn= %f",n);
	float df_th3= -a3 * sin(th3);
	A[0][0]=df_th3;
	float df_th4= a4 * sin(th4);
	A[0][1]=df_th4;
	float df_s15=-1;
	A[0][2]=-1;
	float df_T12=0;
	A[0][3]=0;
	float dg_th3=a3*cos(th3);
	A[1][0]=dg_th3;
	float dg_th4=-a4*cos(th4);
	A[1][1]=dg_th4;
	float dg_s15=0;
	A[1][2]=0;
	float dg_T12=0;
	A[1][3]=0;
	float x=T34*a2*((sin(th2-th3)*cos(th4-th3))-(sin(th4-th3)*cos(th2-
th3))+(cos(th4-th3)*sin(th2-th4)))/(a3*sin(th4-th3)*sin(th4-th3));
	float x1=T45*a2*((cos(th4-th3)*sin(th2-th3))-(cos(th2-th3)*sin(th4-
th3)))/(a4*sin(th4-th3)*sin(th4-th3));
	float dm_th3=x+x1;
	A[2][0]=dm_th3;
	x=(T34*((cos(th2-th4)*sin(th4-th3))+(sin(th2-th4)*cos(th4-th3))-(cos(th4-
th3)*sin(th2-th3))))/(a3*sin(th4-th3)*sin(th4-th3));
	x1=(T45*a2*(cos(th4-th3)*sin(th2-th3)))/(a4*sin(th4-th3)*sin(th4-th3));
	float dm_th4=x-x1;
	A[2][1]=dm_th4;
	float dm_s15=0;
	A[2][2]=0;
	float dm_T12=-1;
	A[2][3]=-1;
	x=(T34*cos(th4))*((cos(th3)*cos(th4-th3))-(sin(th3)*sin(th4-
th3)))/(a3*sin(th4-th3)*sin(th4-th3));
	x1=(T45)*((cos(th3)*cos(th4-th3))-(sin(th3)*sin(th4-th3)))/(a4*sin(th4-
th3)*sin(th4-th3));
	float dn_th3=x+x1;
	A[3][0]=dn_th3;
	x=(-T34)*cos(th3)*((sin(th4)*sin(th4-th3))+(cos(th4)*cos(th4-
th3)))/(a3*sin(th4-th3)*sin(th4-th3));
	x1=(T45*cos(th3)*cos(th4-th3))/(a4*sin(th4-th3)*sin(th4-th3));
	float dn_th4=x-x1;
```

```
                A[3][1]=dn_th4;
                float dn_s15=0;
                A[3][2]=0;
                float dn_T12=0;
                A[3][3]=0;

                int loop=i;
                /*
                printf("\nMatix A::\n");
                for(i=0;i<4;i++)
                {
                        for(j=0;j<4;j++)
                        {
                                printf("%f\t",A[i][j]);
                        }
                        printf("\n");
                }

                printf("\nMatix B::\n");
                for(i=0;i<4;i++)
                {
                        for(j=0;j<1;j++)
                        {
                                printf("%f\t",B[i][j]);
                        }
                        printf("\n");
                } */

                MatrixInversion(A,Y);
                /*
                //Matrix Inverse of A
                printf("\nInverseMatix of A::\n");
                for(i=0;i<4;i++)
                {
                        for(j=0;j<4;j++)
                        {
                                printf("%f\t",Y[i][j]);
                        }
                        printf("\n");
                }
                */
                matrixMultiplication(Y,B,Z);
                //revisited values of theta-3, theta-4, s15,T12
                th3=th3+Z[0][0];
                th4=th4+Z[1][0];
                s15=s15+Z[2][0];
                T12=T12+Z[3][0];
                i=loop;
                //getch();
            }
            float th3d=(th3*180)/pi;
            float th4d=(th4*180)/pi;
            fprintf(fp1,"\n%f\t%f\t%f\t%f\t%f\t%f\t%f\t%f",f,g,m,n,th3d,th4d,s15,T12);
        }
        fclose(fp1);
        getch();
        return 0;
}
```

Fig.A2 shows the flowchart for solving 4 nonlinear simultaneous equations:

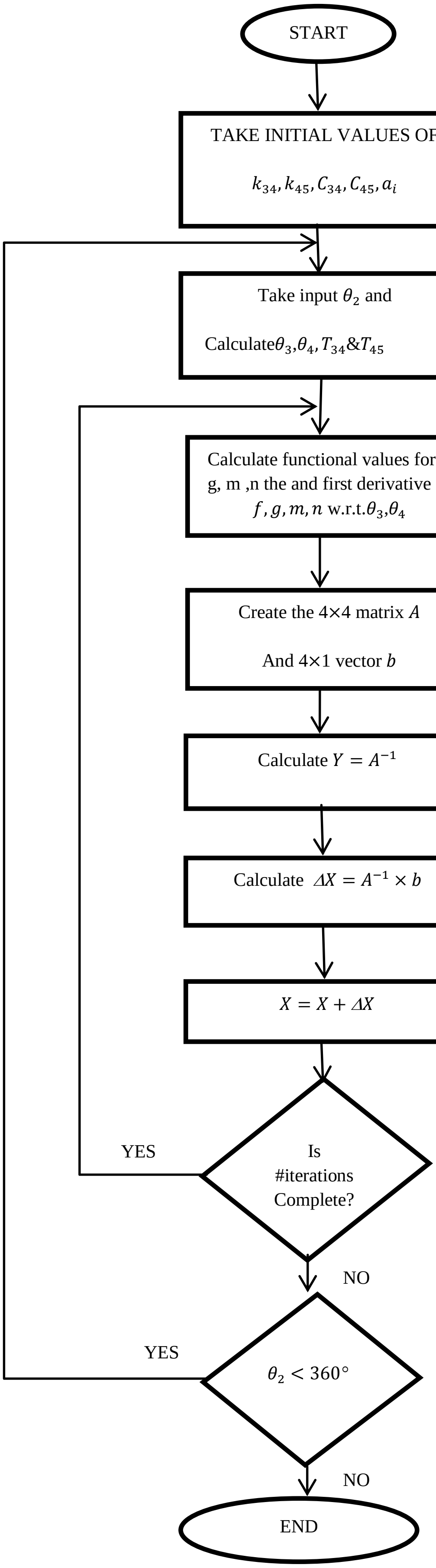

Fig.A2 Flowchart of second program