THE COMPUTER-AIDED STRUCTURAL DESIGN OF ROBOTIC MANIPULATORS

MAHALINGAM SUBBIAH

THE COMPUTER-AIDED STRUCTURAL DESIGN OF ROBOTIC MANIPULATORS

by

©Mahalingam Subbiah, B.Eng.(Hons.)

A thesis submitted to the School of Graduate Studies
in partial fulfillment of the
requirements for the degree of
Master of Engineering

Faculty of Engineering and Applied Science
Memorial University of Newfoundland

St. John's Newfoundland Canada

ISBN 0-315-33634-X

ABSTRACT

The research carried out in this work involves the kinematic and dynamic analyses of the robotic manipulators. The kinematic analysis includes the displacement analysis of the planar and spatial mechanisms as well as the velocity and acceleration analyses. In the dynamic analysis, the variation of the natural frequencies and the dynamic condensation techniques have been studied.

The displacement analysis of the open and closed-loop systems have been carried out using the optimization principles and the modified Newton-Raphson technique. The velocity and acceleration analyses have been divided into two categories: in the first one, the number of unknowns are either greater or less than three and in the second one they are equal to three. The first type is solved using the modified Newton-Raphson technique whereas the second type solved by simultaneous solution of the equations.

To carry out the dynamic analysis, the equations of motion have been obtained using the finite element analysis. The system matrices for the robotic manipulators are also dependent on the angular velocities and angular acceleration. Several new methods of obtaining the matrix of direction cosines needed to represent the orientation of the local axes

with respect to the global axes, are discussed. The natural frequencies as a result of the variation of several design parameters have been obtained. Finally, a comparative study on the two types of dynamic condensation schemes have been carried out using a machine tool spindle and a robotic manipulator as examples.

ACKNOWLEDGEMENTS

I would like to thank Dr. Anand M. Sharan for his continued guidance and encouragement during the course of this investigation. His contribution of time and technical expertise have helped me immensely. I am very grateful to him for his excellent supervision.

I express my thanks to Dr. F. A. Aldrich, Dean of Graduate Studies and Dr. G. R. Peters, Dean of Engineering for providing me the financial assistance during my studies. I am also deeply indebted to Dr. T. R. Chari, Associate Dean of Engineering, for all his assistance during my graduate program.

Finally, a special word of thanks goes to Mrs. Levinia Vatcher for her great patience and cooperation in quickly typing this thesis.

DEDICATION

This work is dedicated to my everloving parents Meenakshi and Mahalingam for all the sacrifices they made to enlighten my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

a_i	explicit constraint that specifies the lower limits for optimization variables
b_i	explicit constraint that specifies the upper limits for optimization variables
b_{ig}	4x1 position vector for gth grid position on ith link
[C]	global damping matrix of the full system
$[C_i]$, $[C_{ij}]$, $[C_{ijk}]$	4x4 dynamic coefficient matrices
d	slave degrees of freedom
D_i	inner diameter of the forearm
D_o	outer diameter of the forearm
$[D_i]$, $[E_i]$, $[F_i]$	4x4 dynamic coefficient matrices
$(f_{i\alpha})_1$	αth term in the externally applied force vector
$(f_{i\alpha})_2$	αth term in the velocity force vector
$(f_{i\alpha})_3$	αth term in the acceleration force vector
$(f_{i\alpha})_4$	αth term in the gravity force vector
F(t)	global force vector of the full system
$F(\underline{x})$	objective function to be optimized
$[g_i]$	unreduced damping matrix for ith link
g_k	constraint equation
g_z	acceleration due to gravity
[G]	system damping matrix including gyroscopic damping
H_i	distance between the axes x_i and x_{i+1}
i	link index
[I]	4x4 identity matrix
j	link index

$[J_{i\alpha}],[J_{i\alpha\beta}]$	4x4 mass matrices
k_x,k_y,k_z	x, y and z components of the screw axis
$(k_{i\beta\gamma})_1$	stiffness term in the βth row and γth column of the structural stiffness matrix of the ith link
$(k_{i\beta\gamma})_2$	stiffness term in the βth row and γth column of the velocity stiffness matrix of the ith link
$(k_{i\beta\gamma})_3$	stiffness term in the βth row and γth column of the acceleration stiffness matrix of the ith link
$[k_i]$	unreduced stiffness matrix
$[K]$	stiffness matrix of the full robot
$(KE)_i$	kinetic energy of the link i
l_1,m_1,n_1	direction cosines of the local x-axis with respect to the global X_o, Y_o and Z_o axes respectively
l_2,m_2,n_2	direction cosines of the local y-axis with respect to the global X_o, Y_o and Z_o respectively
l_3,m_3,n_3	direction cosines of the local z-axis with respect to the global X_o, Y_o and Z_o respectively
L_i	distance between the axes z_i and z_{i+1}
m	number of free coordinates
$[m_i]$	unreduced mass matrix of the ith link
m_{ig}	mass of the gth grid point in the ith link
[M]	global mass matrix of the total system
$[M_{i\alpha}]$	4x4 gravity matrix for ith link
n	number of interface coordinates
n_x,n_y,n_z	direction cosines of the normal to a plane
N	total number of independent generalized coordinates q
NL	total number of links in the mechanism
$(NG)_i$	number of grid points in the ith link

$(NP)_i$	number of perturbation coordinates in the ith link
$(PE)_i$	potential energy of the ith link
$(PE)_i^E$	potential energy due to elasticity of the ith link
$(PE)_i^G$	potential energy due to gravity of the ith link
P_k	penalty constant
$(P_x)_{calc}$	the calculated x coordinate of the point P
$(P_x)_{sp}$	the specified x coordinate of the point P
q_i	generalized coordinates
[Q]	4x4 matrix derivative operator
r	master degrees of freedom
$\underline{r}_i$	position vector of an arbitrary point with respect to the ith link
$\underline{r}_i^o$	position vector of a point in the ith link with respect to the inertial frame
R_i	random number
s	number of feasible points in the complex
S_i	inner side of the square cross-section upperarm
S_o	outer side of the square cross-section upperarm
t	time
[T]	the transformation matrix relating the local coordinate system to the global coordinate system
$[T_c]$	transformation matrix for component mode synthesis
$[T_{i-1}^i]$	4x4 transformaton matrix from (i-1)th link to ith link
$[U_{ij}], [U_{ijk}]$	4x4 derivative matrices
$\underline{v}_{ig}$	velocity vector of the gth grid point in the ith link
{w(t)}	displacement vector of the global system

$\{\dot{w}(t)\}$	velocity vector of the global system
$\{\ddot{w}(t)\}$	acceleration vector of the total system
$\{\underline{x}\}$	vector of design variables
$\{\underline{x}_c\}$	centroid of the complex
Δx_i	change in the value of the ith design variable
x_m-y_m-z_m	local coordinate system
X_o-Y_o-Z_o	global coordinate system
Z_{ig}	coordinate of the gth grid point in the ith link with respect to the inertial frame
α_i	the twist angle that screws z_i to z_{i+1}
α,β,γ	Euler angles
θ_i	the angle that screws x_i with x_{i+1} along H_i
$\underline{\theta}$	a vector having θ_i, i=1,2,3 as its elements
$[\phi]$	transformation matrix for Guyan's Reduction
$\{\phi_{ig}\}$	4x1 shape vector for gth grid point on ith link
$[\Phi]$	matrix of eigen vectors
$\{\eta\}$	modal coordinate vector
$\wedge$	a symbol used for unit vectors
{ } (_)	column vector
[ ]	matrix
$[\ ]^{-1}$	matrix inverse
$[\ ]^T$	matrix transpose
$(\dot{\ })$, $\frac{d}{dt}$ ( )	time derivative
Tr( )	trace operation
$\odot$	matrix dot product
| |	absolute value
$\frac{\partial}{\partial x}$	partial derivative with respect to x

CHAPTER 1

INTRODUCTION AND LITERATURE SURVEY

1.1 The Automation and Robotics

A revolutionary change in factory production techniques is predicted by the end of the twentieth century. Every operation in the factory of the future, from the product design to manufacturing, assembly, and product inspection would be carried out using the machine tools and robots aided by computers. This new development (the automatic controlled factory) is nothing more than a new phase in the industrial revolution that began in Europe two centuries ago and progressed through the following stages:

1. The construction of simple production machines and mechanizations started in 1770.
2. Fixed automatic mechanisms and transfer lines for mass production came along at the turn of this century.
3. Machine tools with simple automatic controls such as plug-board controllers to perform fixed sequence of operations, and copying machines in which a stylus moves on a master copy.
4. The introduction of numerical control (NC) machine tools, in 1952, opened a new era in automation.

5. Integration of a minicomputer to drive the machine tools (CNC) in 1970.
6. The use of industrial robots for elementary operations such as spot welding, and assembly in 1970.

The new era of automation, which started with the introduction of NC machine tools, was undoubtedly stimulated by the digital computer. The digital technology and computers enable the design of more flexible manufacturing systems (FMS), namely systems which can be adopted by programming to produce or assemble a new product in a short time. Actually, flexibility is the key word which characterizes the new era in industrial automation. Robots and manufacturing systems are becoming more and more flexible with progress in computer technology and programming techniques.

The use of the robots has led to increased productivity and better as well as consistent quality in the manufacturing environment. The robots are also being used in the hazardous environment such as nuclear reactors or under-sea operations where they are controlled remotely. Another example of their usage is in the spray-painting jobs where they replace human being who breath toxic air presently. In summary, the advantages of using robots can be stated as follows:

1. Flexibility
2. Higher productivity
3. Better quality of products
4. Improved quality of human life by performing undesirable and hazardous jobs.

A T3R3 model of a robot manufactured by the Cinncinati Milacron Company is shown in the Figs. 1.1a and 1.1b respectively. This robot can be used by varying three degrees of freedom which are θ_1, θ_2 and θ_3 where θ_1 is the rotation about the base, whereas θ_2 and θ_3 are the rotational angles of the upperarm and forearm respectively. These angles are used to position the gripper or the end effector at a location in the three dimensional space. Once the end effector is positioned at the point, the orientation of the gripper is controlled by three independent angles which are shown as pitch, roll and yaw in Fig. 1.1a.

In the inverse kinematic analysis, the location or the trajectory of the end effector is specified and the problem is to find the angles θ_1, θ_2 and θ_3 corresponding to each location of the end effector. Similarly, in the velocity or the acceleration analysis, the velocity or the acceleration of the end effector is specified in the cartesian space and one has to obtain the respective quantities in the θ space. In the dynamic analysis of these structures, one has to evaluate the system matrices

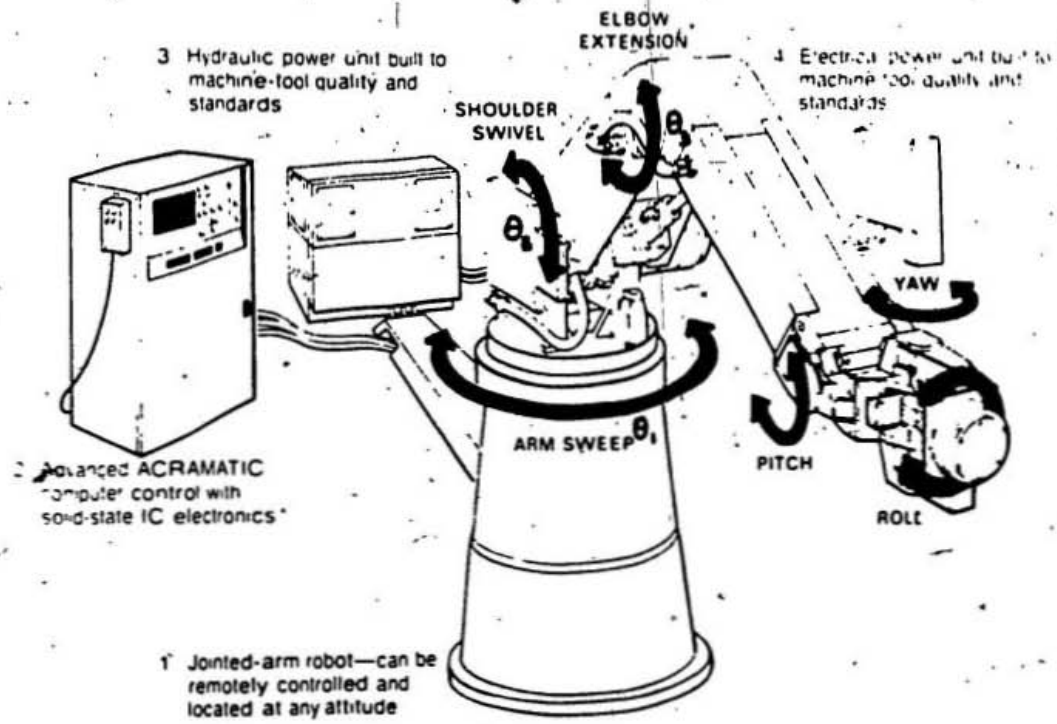

Fig. 1.1a: Industrial Robotic Manipulator [3]

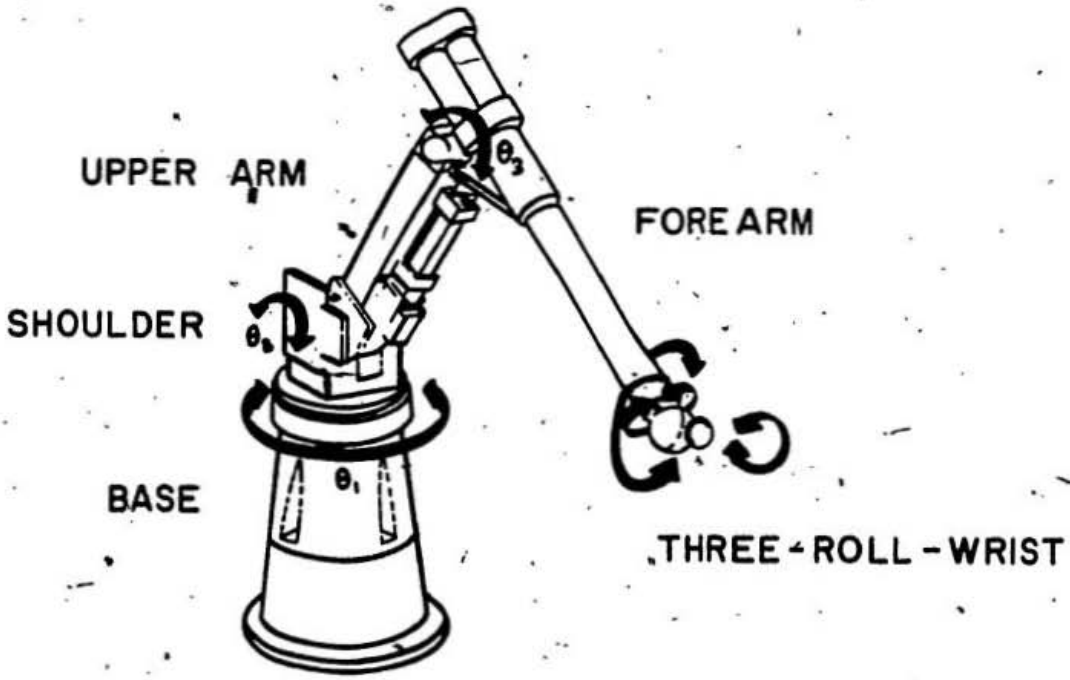

Fig. 1.1b: T3R3 Model Robotic Manipulator [1]

corresponding to each location of the end effector because of the nonlinear variation of these matrices as a function of the location of the end effector. Moreover, it is well known [1] that all these matrices are also a function of joint velocities and joint accelerations. Thus, the dynamic analysis of these structures, which are much more flexible than the conventional machine tools, is fairly involved.

1.2 The Literature Survey

1.2.1 The Kinematic Analysis

The displacement, velocity and acceleration analyses of planar closed loop mechanisms using several techniques such as the graphical method, the complex algebra method etc., are discussed in [2-4]. Some simple spatial closed loop mechanisms are also solved in these references. In reference [5], an iterative technique known as Newton-Raphson technique was used to solve the planar problems. A new technique called the 4 x 4 matrix technique was introduced by Uicker et al [6] to solve spatial closed loop mechanisms. The main advantage of this technique lies in the easier implementation in the computer program. Hall et al [7] solved spatial closed loop mechanisms by using gradient optimization technique. For solving the planar and spatial open-loop mechanisms Turic [8] used the 4 x 4 matrix notation and solved the inverse kinematics problem using

the modified Newton-Raphson technique where the number of unknowns were more than the number of equations. Paul [9] and Craig [10] derived the equations for the inverse displacement analysis of some commonly available robotic manipulations. Some other important research work in this area can be seen in [11-15].

1.2.2 The Dynamic Analysis

Significant progress has been made on the dynamic modeling of rigid spatial mechanisms and manipulators in [16-18]. These investigations include such methods as screw calculus [16], dual vectors [17] and 4 x 4 matrices [18]. The dynamics of the flexible spatial systems was studied in [19].

Finite element methods were first applied to spatial linkages by Winfrey [20]. He investigated the possible advantages of using the Guyan's reduction technique to reduce the size of the dynamic system matrices [21]. Another important reduction technique known as Component Mode Synthesis was used in [22, 23] where the dynamics of linkages was studied. The flexibility and control, in the dynamic analysis of the links in arm-type devices, was reported in [24]. Book and Whitney [25-26] carried out research on the linearly distributed dynamics of planar arms via transfer matrices. Several other researchers have addressed the

flexible-manipulator dynamics problem by analyzing the flexible mechanisms [1, 27-29]. Dubowsky and Gardner [27] have provided bibliographies on this type of work. Sunada [1] developed modelling techniques applicable to both spatial closed loop mechanisms and manipulator arms. Such a technique assumes a known nominal motion over time about which the flexible-arm equation are linearized. This technique is oriented towards finite-element analysis to obtain modal characteristics of the link, which are then combined using a time-varying compatibility matrix. It uses the 4 x 4 matrices to represent the nominal kinematic equations and also in the derivation of the compatibility matrix. The same technique is also used in the present investigation.

Recently, the nonlinear modal analysis technique was used by Book [28] to develop the equations of motion for flexible manipulator arms consisting of rotary joints that connect pairs of flexible links. Currently, work is being carried to analyze flexible manipulators containing prismatic and rotary joints using the finite elements by Naganathan and Soni [29].

1.3 The Objectives of this Investigation

Since the robot manipulators are moving space structures whose system matrices are functions of kinematic

parameters, the present work has been divided into two parts; the first part is the kinematic analysis of these space mechanisms, and the second part is the dynamic analysis. The specific objectives in the kinematic and dynamic analyses are:

1. The displacement analysis of the planar and spatial closed loop mechanisms using the optimization principles.
2. The displacement analysis of the planar and spatial open loop mechanisms using the optimization principles.
3. The velocity analysis of the open loop mechanisms.
4. The acceleration analysis of the open loop mechanisms.
5. The determination of the matrix of direction cosines of the spatial links.
6. The variation of the natural frequencies of the robotic manipulator as a function of design parameters, and
7. The comparative study of the dynamic condensation techniques applied to the machine tools and robotic manipulators.

In Chapter 2, the optimization principles have been used to carry out the inverse displacement analysis of the planar and spatial closed loop mechanisms at first, and then these techniques have been used to solve the inverse problems for the open loop mechanisms. After this, the angular velocities and acceleration of the various joints

have been obtained for the specified velocity and acceleration of the end effector.

The dynamic equations of motions are described in Chapter 3. The system stiffness and damping matrices are functions of angular-velocities and angular accelerations, the force vector is also a function of the angular velocities and accelerations. To carry out the natural frequency computations, the finite element analysis has been used; to specify the orientation of the spatial links, several methods of obtaining the matrix of direction cosines is discussed. Finally, to reduce the degrees of freedom, two dynamic condensation schemes are used.

CHAPTER 2

THE KINEMATIC ANALYSIS OF THE CLOSED AND OPEN LOOP MECHANISMS INCLUDING ROBOTIC MANIPULATORS

2.1 Introduction

The robots are used for welding, assembly, painting jobs, etc. in the industrial environment. While performing a job, the end effector is moved along a specified trajectory with a given velocity and acceleration. This requires a coordinated movement of the various arms of the robot which are driven by independently operated motors or drives. Thus, for every location of the end effector, it is necessary to know the $\underline{\theta}$, $\underline{\dot{\theta}}$ and $\underline{\ddot{\theta}}$ where $\underline{\theta}$ is the vector having as its elements, all the kinematic degrees of freedom of the system. The determination of these vectors is carried out in this chapter. The inverse displacement analysis is carried out first and is presented in the next section.

2.2 The Displacement Analysis of the Planar and Spatial Closed Loop Mechanisms

When the path of the points on a link in a given mechanism lie in a single plane or in parallel planes, it is called a planar mechanism. A vast majority of mechanisms in use today belong to this category. To analyze these

mechanisms, there are many techniques available in the literature [2-5]. Some of these important techniques are:

1. the graphical method,
2. the complex algebra method,
3. the vector algebra method, and
4. the Newton-Raphson method.

In the spatial closed loop mechanisms, the motion of a given point may describe a curve that does not lie in a plane. Even though the conventional method of analyzing planar mechanisms like the vector method can be extended for the spatial mechanism, the calculation becomes enormous and cumbersome, and invariably it leads to the use of the digital computers. Hence, the numerical methods are widely used in the kinematic analysis of spatial mechanisms. In this section, some of the important numerical methods are discussed with suitable examples. Before the numerical methods are discussed, a brief introduction regarding the notations used in setting up the spatial-mechanism displacement-equations is desirable and is discussed next.

2.2.1 The Hartenberg-Denavit Notation

Let us consider the two coordinate systems shown in Fig. 2.1 which describe the relative position and orientation of the (i-1)th and ith links in a spatial mechanism. The two coordinate systems are related by the four coordinate

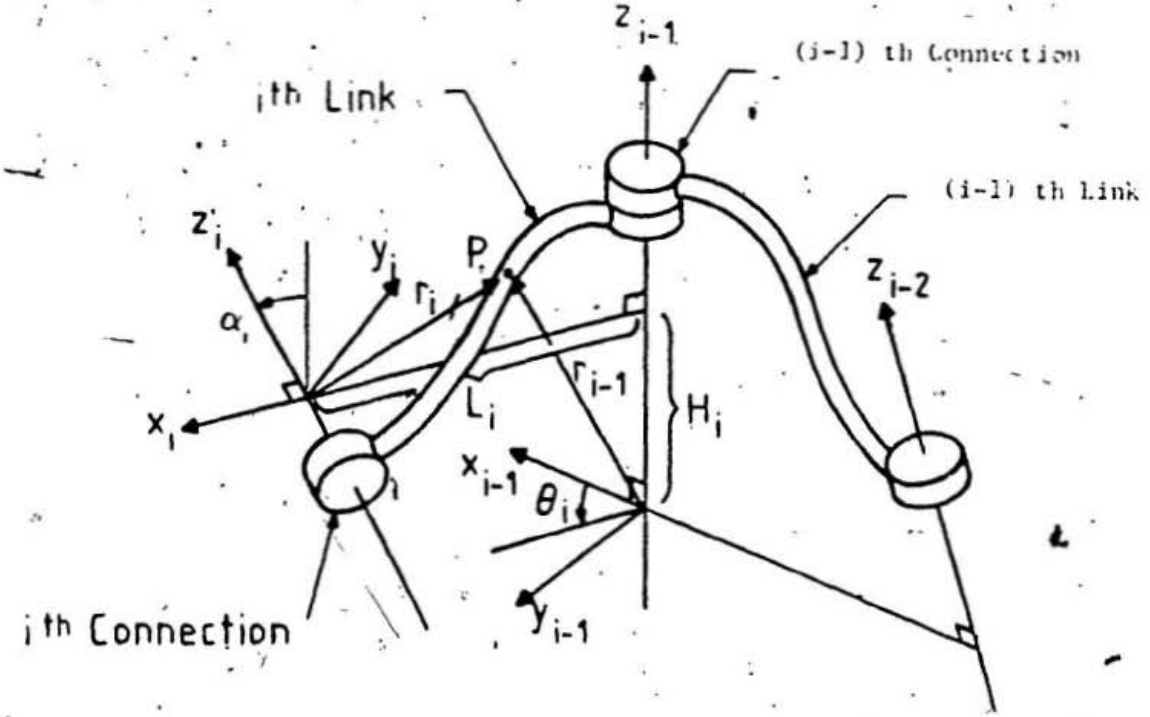

Fig. 2.1: Hartenberg-Denavit Notation

transformation parameters θ_i, α_i, L_i and H_i [6]. The z_i axis represents the rotational axis for each of the individual revolute joints of the mechanism. The x_i axis lies perpendicular to the z_{i-1} and z_i axes and the perpendicular length between these axes is represented as L_i. H_i is the distance along z_i from x_{i-1} to x_i. θ_i is the relative rotation of the x_i axis about the z_{i-1} axis and α_i is the relative rotation of the y_i and z_i axes about the x_i axis. For the links connected only by revolute joints, θ_i is the only time varying parameter. Similarly, for a link with prismatic slider joint, H_i would be the only time varying parameter.

The position vector of any point referenced to the ith link coordinate system can be described as

$$\underline{r}_i = \begin{bmatrix} 1 \\ X_i \\ Y_i \\ Z_i \end{bmatrix} \qquad (2.1)$$

where $\underline{r}_i$, is a 4 x 1 column vector and it represents the position of a point P with respect to the origin of the ith frame. Similarly, the position vector $\underline{r}_{i-1}$ defines a position vector from the origin of the (i-1)th frame. The relation between $\underline{r}_i$ and $\underline{r}_{i-1}$ can be written as

$$\underline{r}_{i-1} = [T^i_{i-1}]\ \underline{r}_i \qquad (2.2)$$

where $[T^{i}_{i-1}]$ is a 4 x 4 link transformation matrix, and it is given as

$$[T^{i}_{i-1}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ L_i\cos\theta_i & \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i \\ L_i\sin\theta_i & \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i \\ H_i & 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix} \quad (2.3)$$

The position vectors in other coordinate systems can be obtained by multiplying the appropriate link transformation matrices. If the reference or the inertial system is X_0-Y_0-Z_0, then the inertial position vector can be defined as

$$\underline{r}_i^{\,0} = [T_0^{\,1}][T_1^{\,2}][T_2^{\,3}] \ldots [T^{i}_{i-1}]\,\underline{r}_i = [T_0^{\,i}]\,\underline{r}_i \quad (2.4)$$

where

$$[T_0^{\,i}] = [T_0^{\,1}][T_1^{\,2}][T_2^{\,3}]\ldots[T^{i}_{i-1}] \quad (2.5)$$

Before proceeding further, it is essential to note some of the important properties of this matrix. This matrix is only a time function of θ_i and H_i, since L_i and α_i are always fixed for a given geometry. If a spatial mechanism contains only revolute joints then H_i will also be a

constant, and so the transformations matrix in this case can be written as $[T^{i}_{i-1}(\theta_i)]$. If this matrix is written for a small increment in θ_i then we get

$$[T^{i}_{i-1}(\theta_i+d\theta_i)] =$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ L_i\cos(\theta_i+\delta\theta_i) & \cos(\theta_i+\delta\theta_i) & -\sin(\theta_i+\delta\theta_i)\cos\alpha_i & \sin(\theta_i+\delta\theta_i)\sin\alpha_i \\ L_i\sin(\theta_i+\delta\theta_i) & \sin(\theta_i+\delta\theta_i) & \cos(\theta_i+\delta\theta_i)\cos\alpha_i & -\cos(\theta_i+\delta\theta_i)\sin\alpha_i \\ H_i & 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix} \tag{2.6}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ L_i\cos\theta_i & \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i \\ L_i\sin\theta_i & \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i \\ H_i & 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 0 \\ -L_i\sin\theta_i & -\sin\theta_i & -\cos\theta_i\cos\alpha_i & \cos\theta_i\sin\alpha_i \\ L_i\cos\theta_i & \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i \\ 0 & 0 & 0 & 0 \end{bmatrix} \delta\theta_i \tag{2.7}$$

The first of the two matrices in the Eqn. (2.7) can be immediately identified as the original transformation matrix evaluated at θ_i and the second matrix can be seen to be the first partial derivative of the transformation matrix with respect to θ_i. Therefore, Eqn. (2.7) can be rewritten as

$$[T^i_{i-1}(\theta_i+d\theta_i)] = [T^i_{i-1}(\theta_i)] + \frac{\partial}{\partial\theta_i}[T^i_{i-1}]d\theta_i \qquad (2.8)$$

The second matrix $\frac{\partial}{\partial\theta_i}[T^i_{i-1}]$ can be obtained using the following simple matrix multiplication.

$$\frac{\partial}{\partial\theta_i}[T^i_{i-1}] = [Q^\theta][T^i_{i-1}] \; . \qquad (2.9a)$$

where,

$$[Q^\theta] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (2.9b)$$

If the time varying parameter was H_i instead of θ_i then we would have

$$\frac{\partial}{\partial H_i}[T^i_{i-1}] = [Q^H][T^i_{i-1}] \qquad (2.10a)$$

where,

$$[Q^H] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \qquad (2.10b)$$

These properties of the 4 x 4 transformation matrix are very useful in reducing the computations and can be very easily implemented in the computer programs used for the kinematic analysis of the planar and spatial mechanisms which is explained next.

2.2.2 The Mathematical Formulation of the Displacement Equation

In carrying out the displacement analysis, several iterative techniques have been used [7, 30, 31]. In these techniques either Newton-type [31] or Quasi-Newton-type [7] approach has been used. In the Newton type procedure an initial estimate of the desired solution vector is made, then the Jacobian matrix is evaluated corresponding to the initial starting vector and the improved estimate vector is calculated. If the improved estimate vector meets the convergence criteria then the iteration is stopped, otherwise it is continued. Unfortunately, the disadvantage of this method is that if the initial guess of the starting vector is not very close to the final vector then the method fails to

converge. On the other hand, if one uses quasi-Newton method such as the Davidon-Fletcher-Powell method the solution or the multiple-solutions are almost always found. However, both these methods are gradient methods, where the gradients are evaluated either numerically or analytically. The gradient methods do not always proceed smoothly to the optima. Thus, a designer involved in the kinematic analysis of complex mechanisms should also have some alternate methods for the analysis. One of such methods would be using the direct search methods which if combined with the penalty functions can be a very powerful tool in analyzing mechanisms with nonlinear constraints [32]. In this thesis, the planar four-bar mechanism and two space-mechanisms are solved using one of these types of methods called the Hooke and Jeeves method [33] which is a direct search method. It does not involve the evaluation of the gradients. Finally, the design of a planar crank-rocker type mechanism is illustrated which can be used for flame-cutting of holes in the plates.

Using the transformation matrix, when applied to a four-bar closed-loop mechanism one can write

$$[T_1{}^2][T_2{}^3][T_3{}^4][T_4{}^1] = [I] \qquad (2.11)$$

where [I] is the identity matrix. Another way of expressing the equations for a n link closed loop mechanism is

$$R_1 \begin{bmatrix} L_1 \\ \alpha_1 \\ \theta_1 \\ H_1 \end{bmatrix} \quad R_2 \begin{bmatrix} L_2 \\ \alpha_2 \\ \theta_2 \\ H_2 \end{bmatrix} \quad \cdots\cdots \quad P_i \begin{bmatrix} L_i \\ \alpha_i \\ \theta_i \\ H_i \end{bmatrix} \quad \cdots\cdots \quad R_n \begin{bmatrix} L_n \\ \alpha_n \\ \theta_n \\ H_n \end{bmatrix} = [I] \tag{2.12}$$

where R and P represent revolute and prismatic joints respectively. When dealing with space mechanisms, the cylindrical pairs are replaced by revolute and prismatic pair combinations. Transposing [I] to the left-hand side in Eqn. (2.11), one can write

$$[T^*] - [I] = [0] \tag{2.13}$$

where $[T^*]$ is the matrix product of all the transformation matrices. To obtain the unknown angles, one can define the objective function as the sum of the square of the difference of each of the elements of the matrices $[T^*]$ and [I], which can be mathematically expressed as

$$F(\underline{x}) = \sum_{i=1}^{n} \sum_{j=1}^{n} (T^{*}_{ij} - I_{ij})^2 \qquad (2.14)$$

where $\{\underline{x}\}$ is the design vector having unknown parameters as its elements.

In the design of the mechanisms, if there are k constraints represented as g_k then the objective function can be modified to include the constraints. The modified objective function can be mathematically written as

$$U(\underline{x}) = F(\underline{x}) + \sum_{k=1}^{m} P_k g_k^2 H(g_k) \qquad (2.15)$$

where $H(g_k)$ is the Heavyside unit step function defined so that

$$H(g_k) = \begin{cases} 1 \text{ for } g_k \geq 0 \\ 0 \text{ for } g_k < 0 \end{cases} \qquad (2.16)$$

In Eqn. (2.15), P_k are large penalty constants which are positive because the present problem is a minimization problem. Next, one needs to solve for the minimum of $U(\underline{x})$ using the Hooke and Jeeves method which is

given in detail in [33,35] and a step-by-step procedure for a design vector $\{\underline{x}\}$ having n components is mentioned below:

1. Start with an initial estimate of the design vector

$$\{\underline{x}\} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \qquad (2.17)$$

and choose Δx_i, i = 1, 2, ... n as step lengths in each of the coordinate directions $\underline{u}_i$, i = 1, 2, ... n.

2. Set temporary base point

$$\underline{y}_{k,0} = \underline{x}_k \qquad (2.18)$$

3. Start the exploratory move by perturbing one design variable at a time in order to find the improved value of the objective function.

Set:

$$\underline{y}_{k,i} = \underline{y}_{k,i-1} + (\Delta x_i)\,\underline{u}_i \text{ if } U^+ = U(y_{k,i-1} + (\Delta x_i)\,\underline{u}_i) < U = U(y_{k,i-1})$$

$$\underline{y}_{k,i} = \underline{y}_{k,i-1} - (\Delta x_i)\,u_i \text{ if } \bar{U} = U(y_{k,i-1} - \Delta x_i \underline{u}_i) < U^- = U(y_{k,i-1}) \quad (2.19)$$

$$< U^+ = U(y_{k,i-1} + \Delta x_i \underline{u}_i)$$

$$\underline{y}_{k,i} = \underline{y}_{k,i-1} \text{ if } U = U(y_{k,i-1}) < \min\,(U^+, U^-)$$

In this way, all the design variables x_n are perturbed and the improved position $\underline{y}_{k,n}$ found.

4. If the point $\underline{y}_{k,n}$ is not different from $\underline{x}_k$, reduce the step lengths Δx_i; set i=1 and go to step 3. If $\underline{y}_{k,n}$ is different from $\underline{x}_k$, obtain the new base point as

$$\underline{x}_{k+1} = \underline{y}_{k,n} \quad (2.20)$$

5. Find the pattern direction $\underline{S}$ using

$$\underline{S} = \underline{x}_{k+1} - \underline{x}_K \quad (2.21)$$

Find the point $\underline{y}_{k+1,0}$ as

$$\underline{y}_{k+1,0} = \underline{x}_{k+1} + \lambda\,\underline{S} \quad (2.22)$$

Find λ^*, the optimum step length in the direction $\underline{S}$ and use λ^* in Eqn. (2.22)

6. Set k = k+1, $U_k = U(y_{ko})$ and i=1; repeat step 3. If at the end of step 3, $U(\underline{y}_{k,n}) < U(\underline{x}_k)$, use the new base point as $\underline{X}_{k+1} = \underline{Y}_{k,n}$ and go to step 5. If $U(\underline{y}_{k,n}) > U(\underline{x}_k)$, set $\underline{x}_{k+1} = \underline{x}_k$ and reduce step lengths; set k = k+1 and go to step 2.
7. The process is terminated if the step lengths become less than ε, a very small quantity.

2.2.3 Results and Discussion of the Planar and Spatial Closed Loop Mechanisms

The Hooke and Jeeves method was used to solve for the unknown parameters in three cases: the first one was the planar four-bar mechanism with dimensions shown in Fig. 2.2 and the other two were two types of seven-pair mechanisms, with dimensions shown in Tables 2.1 and 2.2 respectively. All of these mechanisms are shown in Figs. 2.2 to 2.4.

The four-bar mechanism was also solved using the Newton-Raphson technique, the Davidon-Fletcher-Powell method, and an analytical method (the algebraic method). The results obtained are shown in the Table 2.3. The results clearly indicate that one can obtain these results with sufficient accuracy by using any of the methods. The optimization methods generally converge to the optimum even if the starting vector is not very close to the final value. On the

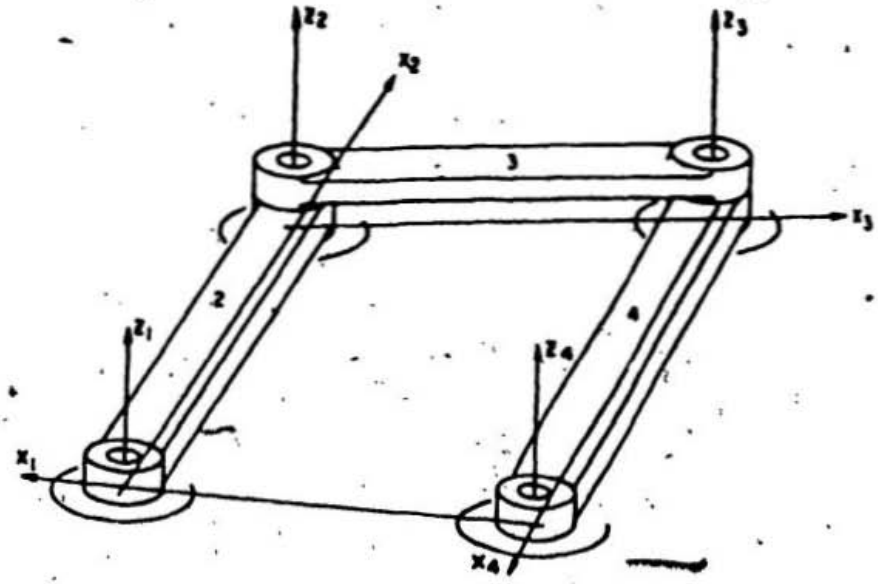

Fig. 2.2: Four Bar Planar Mechanism (a_1 = 0.333, a_2 = 1.5, a_3 = 0.9165, a_4 = 0.833)

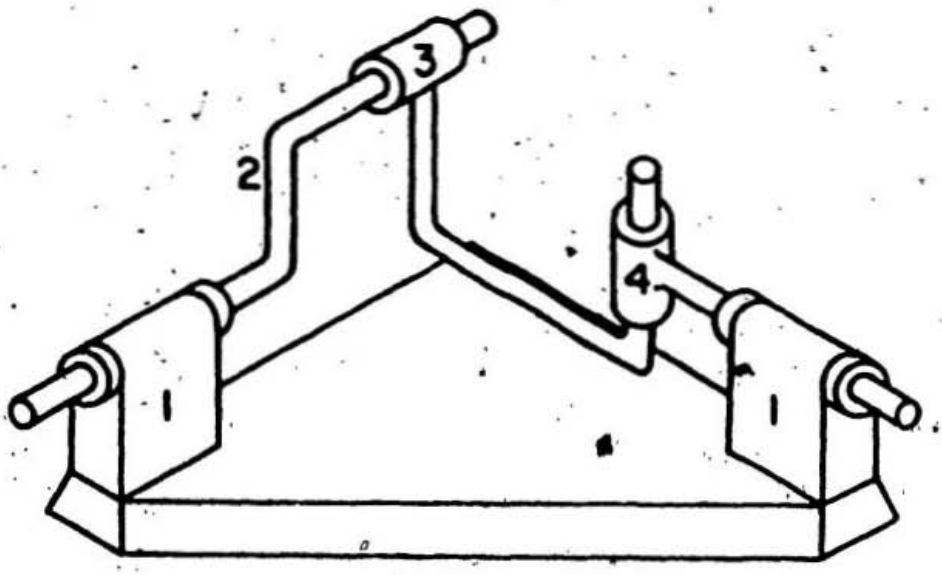

Fig. 2.3: Seven Degree of Freedom Space Mechanism I

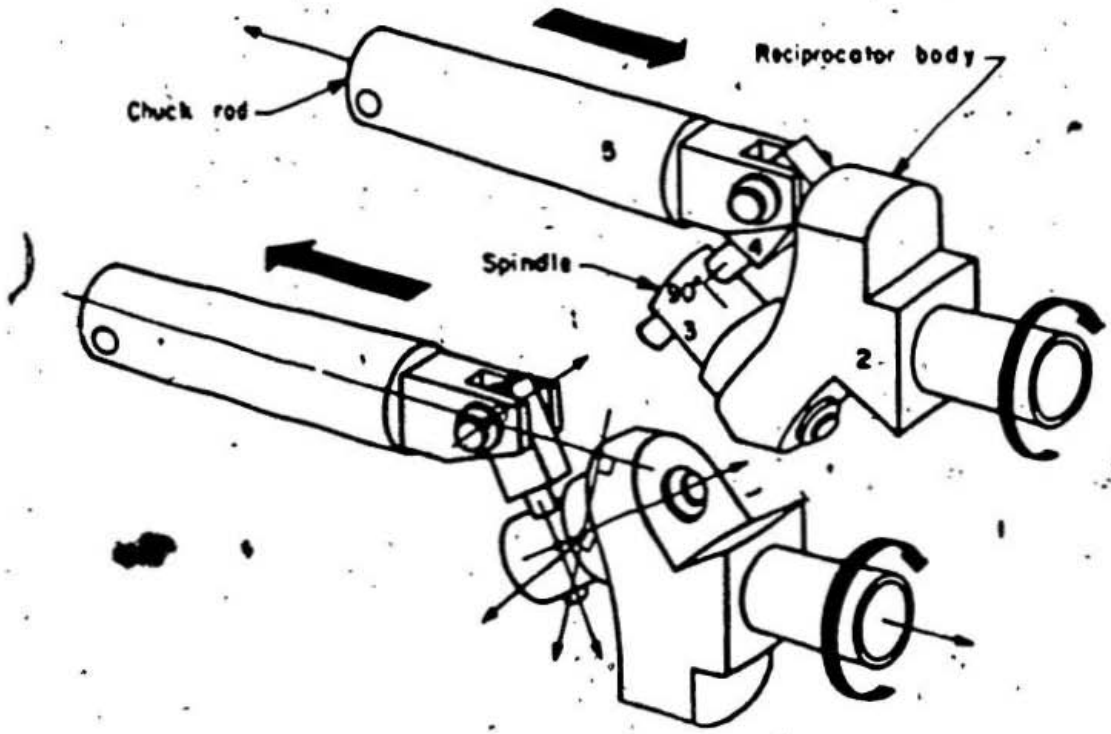

Fig. 2.4: Seven Degree of Freedom Space Mechanism II

Table 2.1

Dimensions of Space Mechanism I

	R_1	R_2	P_3	R_4	P_5	R_6	P_7
L x 0.254m	2.0	0	4.0	0	3.0	0	5.0
α, degrees	30	0	55	0	45	0	60
θ, degrees	θ_1	θ_2	0	θ_4	0	θ_6	0
H x .0254m	0	0	S_3	0	S_5	0	S_7

Table 2.2

Dimensions of Space Mechanism II

	R_1	R_2	P_3	R_4	R_5	P_6	R_7
L x .0254m	0	0	0	0	0	0	0.35
α, degrees	30	90	0	90	90	0	180
θ, degrees	θ_1	θ_2	0	θ_4	θ_5	0	θ_7
H x .0254m	0	0	S_3	0	0	S_6	0

Table 2.3

Results of Planar Mechanism

METHOD	Design Variables			
	θ_1 radian	θ_2 radian	θ_3 radian	θ_4 radian
Analytical Method (Algebraic Method)	5.236	4.5540	3.8940	5.1650
Newton-Raphson Method (Gradient Method)	5.236	4.5540	3.8940	5.1650
Davidon-Fletcher-Powell Method (Gradient Method)	5.236	4.5537	3.8943	5.1655
Hooke and Jeeves Method (Direct-Search Method)	5.236	4.5536	3.8944	5.1653

other hand, Newton-Raphson technique requires that the starting vector be close to the optimum value. The Newton-Raphson technique appears to yield better result in this table because the starting vector was close to the final vector.

The solutions for the two types of seven-pair mechanisms, which are discussed in [7] and are shown in Figs. 2.3 and 2.4, were obtained by the direct search technique and the results are shown in Tables 2.4, and 2.5. The results in both these tables indicate that the final solutions are quite close and that the direct search technique is an alternate method of carrying out the displacement analysis of the space mechanisms.

The dimensions of the planar four-bar mechanism which can be used for flame-cutting a hole on a plate can be obtained using the optimization method. Fig. 2.5 shows the location of 9 points which lie on the surface of a hole. A crank-rocker type of mechanism can be designed such that the coupler point would pass through these 9 points as the crank goes through a complete rotation. These points are located at every 40 degrees of the crank angle rotation In the actual operation, the flame-cutting torch would be mounted at the point P.

In order to flame-cut the hole successfully, the maximum tangential velocity of the point P was specified as 0.0127 m/sec (0.5 inches/sec). Since the equation of the

Table 2.4

Results of Space Mechanism I

METHOD	Design Variables						
	θ_1 radians	θ_2 radians	S_3 (x .0254)m	θ_4 radians	H_5 (x .0254)m	θ_6 radians	S_7 (x .0254)m
Davidon-Fletcher Powell Method [7] (Gradient Method)	0.0	2.618	-0.211	0.785	-2.660	2.520	-0.118
Hooke and Jeeves Method (Direct-Search Method)	0.0	2.610	-0.206	0.799	-2.709	2.514	-0.112

Table 2.5

Results of Space Mechanism II

METHOD	Design Variables						
	θ_1 radians	θ_2 radians	S_3 (x .0254)m	θ_4 radians	θ_5 radian	H_6 (x .0265)m	θ_7 radians
Davidon-Fletcher Powell Method [7] (Gradient Method)	0.0	-1.57	0.35	2.1	4.71	0.0	0.0
Hooke and Jeeves Method (Direct-Search Method)	0.0	-1.57	0.35	2.09	4.71	-0.76 $\times 10^{-6}$	-0.114 $\times 10^{-2}$

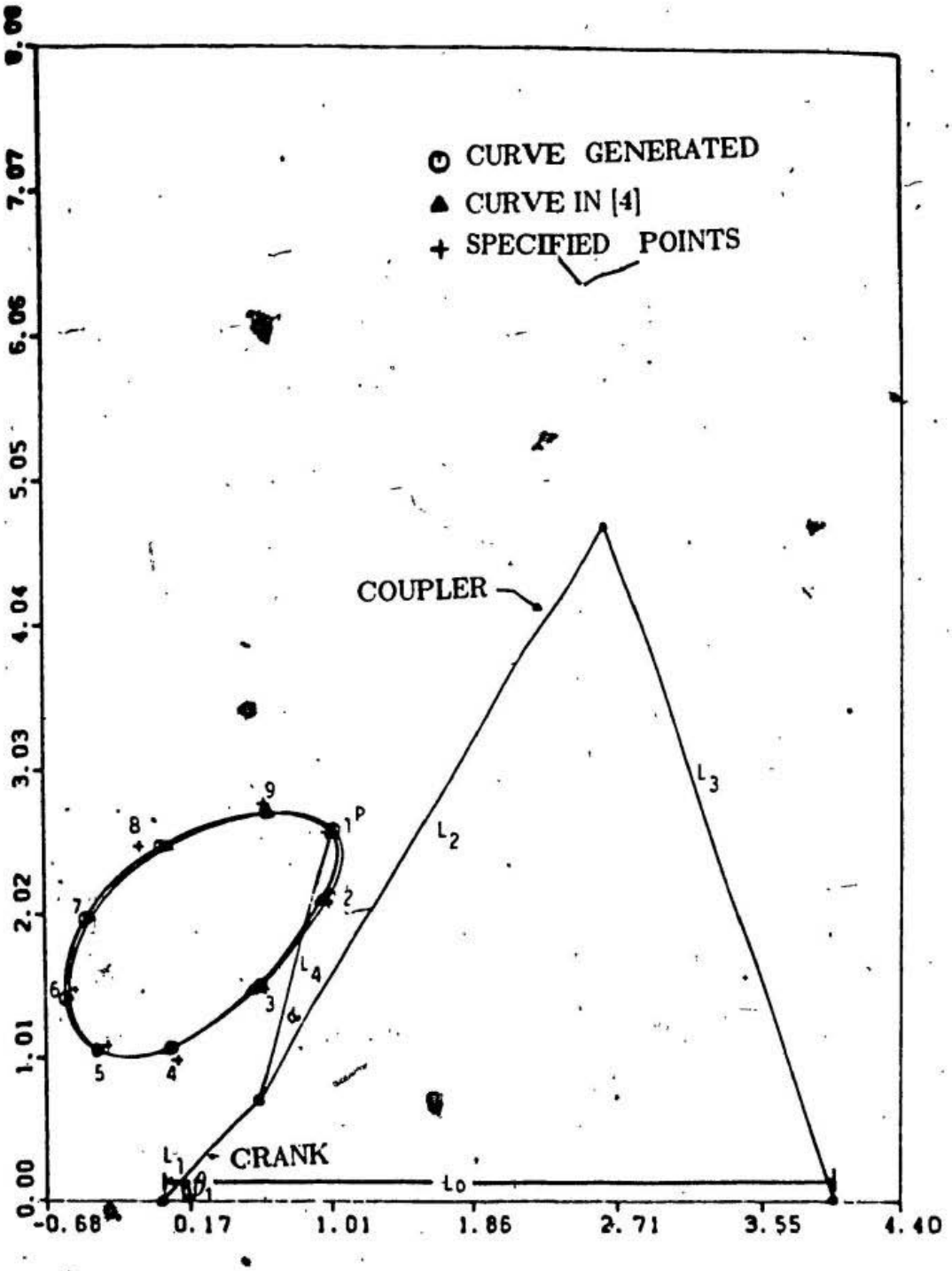

Fig. 2.5: Planar Four Bar Mechanism and the Coupler Curve

closed curve traversed by the point P is not known, the tangential velocity at any point was approximated by using the following steps:

1. A cubic spline curve was fitted through all the 9 points.
2. At the point where the tangential velocity was to be known, the velocity of the point P was resolved along a chord formed by the point under consideration and a close neighbouring point on this curve.

In the optimization study the design vector $\{\underline{x}\}$ was represented by*

$$\{\underline{x}\} = \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ \theta_1 \\ \alpha \end{bmatrix} \qquad (2.23)$$

and the objective function U which was to be minimized was represented as

$$U = \sum_{i=1}^{9} \left\{(P_{xi})_{cal} - (P_{xi})_{sp}\right\}^2 + \left\{(P_{yi})_{cal} - (P_{yi})_{sp}\right\}^2 \qquad (2.24)$$

*The length of the fixed link L_0 is chosen arbitrarily.

The elements of the design vector are shown in the Fig. 2.5. In this figure, the curve obtained by using the optimal design vector are also shown. The curve obtained in the present analysis is very close to those obtained in [4]. Next, the crank optimum angular velocity ω_1 was obtained by a second optimization study where the maximum tangential speed was used as a constraint. The maximum constant crank angular speed was found to be 0.4456 rad/sec. The optimal design vector $\{\underline{x}\}$ for the first optimization obtained as

$$\{\underline{x}\} = \begin{bmatrix} 0.912 \\ 4.723 \\ 4.933 \\ 1.947 \\ 53.430 \\ 15.614 \end{bmatrix}$$

Fig. 2.6 shows the variation of the tangential velocity as a function of the crank angle. This curve has two maxima corresponding to 88 and 290 degrees. It should be added here that the objectives of the study in [4] was to design a four-bar mechanism whose coupler point P passed through the 9 specifiqd points. It did not involve the design of a mechanism for flame-cutting of the hole. Therefore, there was no velocity analysis carried out in that study.

2.3 The Displacement Analysis of Planar and Spatial Open Loop Mechanisms

In this era of high technology, the application of open loop mechanisms has reached almost all areas of engineering; for example, the design of a robot.

The displacement analysis for these types of mechanisms involves finding the joint coordinates of the mechanism for a given position of the end point of the kinematic chain. In other words, the location of the end point is known in both the local as well as the global coordinates and one has to find the possible solutions of this problem. Hence in this section, the displacement analysis of open loop mechanisms is discussed.

For analyzing open loop mechanisms, Turcic [8] used modified Newton-Raphson technique. In this section, at first the modified Newton-Raphson technique is illustrated with examples, and then the complex optimization technique is applied for analyzing the open loop mechanisms. The main advantage of the optimization method is that, among the possible multiple solutions, the best solution is automatically obtained by using a suitable objective function.

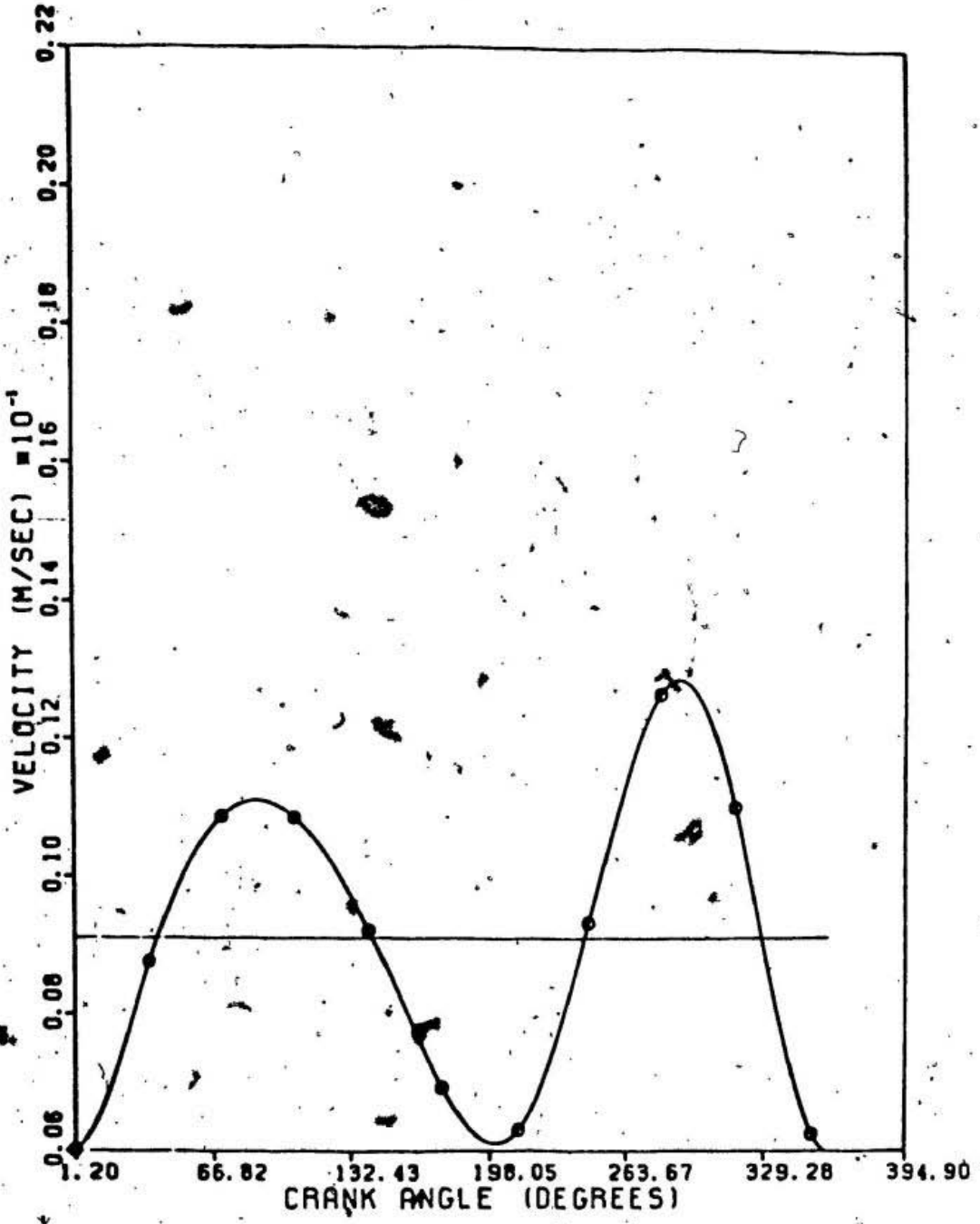

Fig. 2.6: Velocity of the Point P'

2.3.1 The Modified Newton-Raphson Technique

Let us consider a point on the ith link which can be vectorially represented with respect to the ith coordinate system as $\underline{r}_i$ (refer to the Fig. 2.1). The same point if referenced from the inertial coordinates, can be represented as $\underline{r}_i^{\,o}$ and these two quantities can be related as

$$\underline{r}_i^{\,o} = [T_o^{\,i}]\ \underline{r}_i \tag{2.25}$$

where

$$[T_o^{\,i}] = [T_o^{\,1}][T_1^{\,2}][T_2^{\,3}] \ \ldots\ [T_{i-1}^{i}] \ . \tag{2.26}$$

Let

$$\underline{r}_i^{\,o*} = [T_o^{\,i}]^*\ \underline{r}_i \tag{2.27}$$

where

$$[T_o^{\,i}]^* = [T_o^{\,1}(x_1^{\,*})][T_1^{\,2}(x_2^{\,*})][T_2^{\,3}(x_3^{\,*})] \ \ldots\ [T_{i-1}^{i}(x_i^{\,*})] \ . \tag{2.28}$$

Here, $x_1^{\,*}$, $x_2^{\,*}$, $x_3^{\,*}$ ... $x_i^{\,*}$ are the estimated values of the generalized coordinates, and

$$\hat{\underline{r}}_i^{o*} = \begin{bmatrix} 1 \\ x^* \\ y^* \\ z^* \end{bmatrix} \tag{2.29}$$

is the estimated position of the end point, and is also commonly referred to as the control point when the estimated values of the generalized coordinates x_1^*, x_2^*, ... x_i^* are used.

In general, for the initial estimates

$$\hat{\underline{r}}_i^{o*} \neq \underline{r}_i^{o} \tag{2.30}$$

but a first order expansion of this equation yields:

$$\underline{r}_i^{o*} + \sum_{j=1}^{m} \frac{\partial \underline{r}_i^{o*}}{\partial x_j} \Delta x_j = \underline{r}_i^{o} \tag{2.31}$$

where the summation is taken over all the generalized coordinates.

After rearranging, the Eqn. (2.31) can be written as

$$\sum_{j=1}^{m} \frac{\partial\, r_i^{o*}}{\partial x_j} \Delta x_j = \underline{r}_i^{o} - \underline{r}_i^{o*} \tag{2.32}$$

where the right hand side is the error term and represents the difference between the specified position of a control point and the actual position obtained using the estimates for the generalized coordinates.

Let $\Delta\underline{x}$ be defined as a vector representing the change of the generalized coordinates and $\underline{x}$ be a vector containing the generalized coordinates of the system as shown below:

$$\Delta\underline{x} = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \cdot \\ \cdot \\ \cdot \\ \Delta x_m \end{bmatrix}, \qquad (2.33)$$

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_m \end{bmatrix} \qquad (2.34)$$

where m is the number of generalized coordinates of the mechanism or the number of degrees of freedom of the mechanism.

Now Eqn. (2.32) can be rewritten as

$$\left[\frac{\partial \underline{r}_i^{o^*}}{\partial \underline{x}}\right] \Delta \underline{x} = \underline{r}_i^{o} - \underline{r}_i^{o^*} \qquad (2.35)$$

where $\left[\frac{\partial \underline{r}_i^{o^*}}{\partial \underline{x}}\right]$ is a (4 x m) matrix as shown here

$$\left[\frac{\partial \underline{r}_i^{o^*}}{\partial \underline{x}}\right] = \left[\frac{\partial \underline{r}_i^{o^*}}{\partial x_1} \quad \frac{\partial \underline{r}_i^{o^*}}{\partial x_2} \quad \frac{\partial \underline{r}_i^{o^*}}{\partial x_3} \quad \cdots \quad \frac{\partial \underline{r}_i^{o^*}}{\partial x_m}\right] \qquad (2.36)$$

It must be noted that both sides of the above equation have four rows corresponding to 1, x, y, and z coordinates of a point. Eqn. (2.36) can be written as [8]

$$\left[\frac{\partial \underline{r}_i^{o^*}}{\partial \underline{x}}\right] = \left[[D_1]\underline{r}_i^{o^*} \; [D_2]\underline{r}_i^{o^*} \; [D_3]\underline{r}_i^{o^*} \; \cdots \; [D_m]\underline{r}_i^{o^*}\right] \qquad (2.37)$$

where,

$$\begin{aligned} [D_1] &= [T_o^1]\,[Q]\,[T_o^1]^{-1} \\ 4\text{x}4 \quad & 4\text{x}4 \quad 4\text{x}4 \quad 4\text{x}4 \\ [D_2] &= [T_o^2]\,[Q]\,[T_o^2]^{-1} \\ [D_3] &= [T_o^3]\,[Q]\,[T_o^3]^{-1} \\ &\vdots \quad\;\; \vdots \quad\;\; \vdots \\ [D_m] &= [T_o^m]\,[Q]\,[T_o^m]^{-1}. \end{aligned} \qquad (2.38)$$

and $[Q] = [Q^{\theta}]$ is used as in Eqn. (2.8) ff the joint is revolute or $[Q] = [Q^{H}]$, as in Eqn. (2.9) if the joint is prismatic. Let

$$[A] = [\frac{\partial \underline{r}_i^{\,o}}{\partial \underline{x}}] \,, \tag{2.39}$$

Since $\underline{r}_i^{\,o*}$ finally converges to $\underline{r}_i^{o}$, Eqn. (2.35) can be written as

$$[A]\ \Delta\underline{x} = \underline{r}_i^{\,o} - \underline{r}_i^{\,o*} \tag{2.40}$$

where

$$\underset{4xm}{[A]} = [\underset{4x4}{[D_1]}\ \underset{4x1}{\{r_i^{\,o}\}}\ [D_2]\ \{r_i^{\,o}\}\ \ \ldots\ \ [D_m]\ \{r_i^{\,o}\}] \tag{2.41}$$

In general [A] is not a square matrix. By premultiplying both sides of Eqn. (2.40) by $[A]^T$ we get

$$[A]^T\ [A]\ \{\Delta x\} = [A]^T\{\underline{r}_i^{\,o} - \underline{r}_i^{\,o*}\} \tag{2.42}$$

The required increment in the generalized coordinates $\{\Delta x\}$ is given as

$$\{\Delta x\} = [[A]^T[A]]^{-1}\,[A^T]\,\{\underline{r}_i^{\,o} - \underline{r}_i^{\,o*}\} \qquad (2.43)$$

The steps to be followed for the displacement analysis are:

1. Define the mechanism using the appropriate transformation matrices to describe each joint in the mechanism.
2. Define the position of the control point in the local and global coordinate systems.
3. Make an initial estimate of the generalized coordinates and calculate $\underline{r}_i^{\,o*}$ using the transformation matrices.
4. Form the matrix [A] using Eqns. (2.37) and (2.39).
5. Find $\Delta\underline{x}$ using Eqn. (2.43).
6. Update the initial estimates of the generalized coordinates as $\underline{x} = \underline{x}^* + \Delta\underline{x}$.
7. Repeat steps 4 to 6 until the quantity, $\underline{r}_i^{\,o} - \underline{r}_i^{\,o*}$, reduces to the value ε which is an allowable error.

2.3.2 The Optimization Method

In order to solve these types of problems [34], initially an estimate of the generalized coordinates x_i are made, where x_i is θ_i if the connection between (i-1)th and

ith link is revolute and is H_i if the connection is prismatic, and the approximate position vector, $\underline{r}_i^{o*}$ is calculated as

$$\underline{r}_i^{o*} = [T_o^{\,i}]^* \, \underline{r}_i = \begin{bmatrix} 1 \\ x^* \\ y^* \\ z^* \end{bmatrix}_i \qquad (2.44)$$

where $[T_o^{\,i}]$ is calculated by using the estimated generalized coordinates x_i and x^*, y^* and z^* are the approximate position coordinates of the point $\underline{r}_i^{\,o}$. To obtain the unknown values of the generalized coordinates, one can define the objective function as the sum of the squares of the difference between each element of the vector $\underline{r}_i^o$ and $\underline{r}_i^{o*}$, which can be mathematically expressed as

$$F_1(\underline{x}) = (x - x^*)^2 + (y - y^*)^2 + (z - z^*)^2 \qquad (2.45)$$

The minimization of this function $F_1(\underline{x})$ will give the values of the generalized coordinate x_i. This solution will be any one of the multiple solutions that is possible for the open-loop mechanism. In order to obtain the solution that is closest to the previous solution, another objective function is defined as follows

$$F_2(\underline{x}) = \sum_{i=1}^{m} (x_{pi} - x_i)^2 \qquad (2.46)$$

where $\underline{x}_p$ is the previous value of the vector $\underline{x}$ and m is the total number of generalized coordinates in the system. Thus, the final objective function is expressed as

$$\text{minimize} \quad \lim_{a \to \infty} F(a, \underline{x}) = a\, F_1(\underline{x}) + F_2(\underline{x}) \qquad (2.47)$$

where $F_1(\underline{x})$ and $F_2(\underline{x})$ are given by Eqns. (2.45) and (2.46) respectively, and 'a' is a very large number. The reason for using 'a' is that when the solution approaches the final vector then $F_1(\underline{x})$ will be a very small number i.e. it will approach zero whereas $F_2(\underline{x})$ in this case will not approach zero. These types of problems can be solved by the complex optimization method also, and this method is discussed next.

2.3.2.1 The Complex Optimization Method

This method [35] uses a flexible rather than a rigid geometric simplex of points. The basic idea in the sequential simplex method is to start with some arbitrary initial values for the parameters to be optimized.

The steps involved in the optimization are given below:

(i) Minimize $F(\underline{x})$, $\{\underline{x}\} = \{x_1, x_2, \ldots, x_n\}$

where $\{\underline{x}\}$ is a vector of variables $x_1, x_2, \ldots, x_n$ to be optimized subject to

$$a_i \leq x_i \leq b_i; \quad i = 1, 2, \ldots, n \qquad (2.48)$$

$$g_i(\underline{x}) \geq 0;$$

(ii) The method requires the use of $k \geq n + 1$ vertices, each of which must satisfy all the imposed constraints. These vertices may be initially found by starting at a point that satisfied all the constraints. The remaining $k - 1$ points in the first complex are obtained by the use of pseudo-random numbers R_i in the relation

$$x_i = a_i + R_i(b_i - a_i); \quad i = 1, 2, \ldots k \qquad (2.49)$$

where R_i are uniformly distributed over the interval [0,1]. These points satisfy the lower and upper bound constraints. If some implicit constraints are violated, then the trial

point is moved halfway towards the centroid of the already accepted points. The centroid $\underline{x}_c$ is given by the expression

$$\underline{x}_c = \frac{1}{s} \sum_{i=1}^{s} x^i \qquad (2.50)$$

where $\underline{x}^1$, $\underline{x}^2$, ... $\underline{x}^s$ are available feasible vertices.

(iii) The objective function, $F(\underline{x})$ to be minimized is evaluated at each vertex and the vertex $\underline{x}^v$ at which the function $F(\underline{x})$ assumes the largest value is reflected by computing

$$\underline{x}^r = (1 + \alpha)\, \underline{x}^o - \underline{x}^v, \text{ with } \alpha \geq 1 \qquad (2.51)$$

where $\underline{x}^o$ is the centroid of the remaining vertices and is calculated from

$$\underline{x}^o = \frac{1}{k-1} \sum_{\substack{i=1 \\ i \neq v}}^{k} x^i \qquad (2.52)$$

(iv) If the function value $F(\underline{x}^r) < F(\underline{x}^v)$ and $\underline{x}^r$ is feasible, the point $\underline{x}^v$ is replaced by $\underline{x}^r$ and step (ii) is

repeated. If $F(\underline{x}^r) \geq F(\underline{x}^v)$, the overreflection coefficient is reduced to $\alpha/2$, and new $\underline{x}^r$ computed and tried. This is repeated until the difference between two consecutively calculated objective function values is less than β where $\beta = 10^{-5}$ is a satisfactory value. The other details about this method can be seen in [35].

2.3.3 Numerical Examples of Open-Loop Mechanisms

In order to illustrate these techniques two examples are illustrated here. First example is a planar problem taken from [8]. A manipulator with three links was considered in a plane as shown in Fig. 2.7. For the tip of the manipulator to reach a specified position, the problem is to determine the input angles of each link. This problem was solved using the modified Newton-Raphson method, and the complex optimization technique. This problem was solved by the first method in [8]. In the second example, a three-dimensional problem involving a robotic mechanism was solved using these two techniques. An actual industrial manipulator is shown in Fig. 1.1b. It is a T_3R_3 model, a six degrees of freedom robot manufactured by Cincinnati Milacron [1]. The kinematic model of the robot is shown in the Fig. 2.8. The X_o, Y_o, Z_o coordinates represent the fixed inertial base frame. The relative angles between links 0, 1, 2, and 3

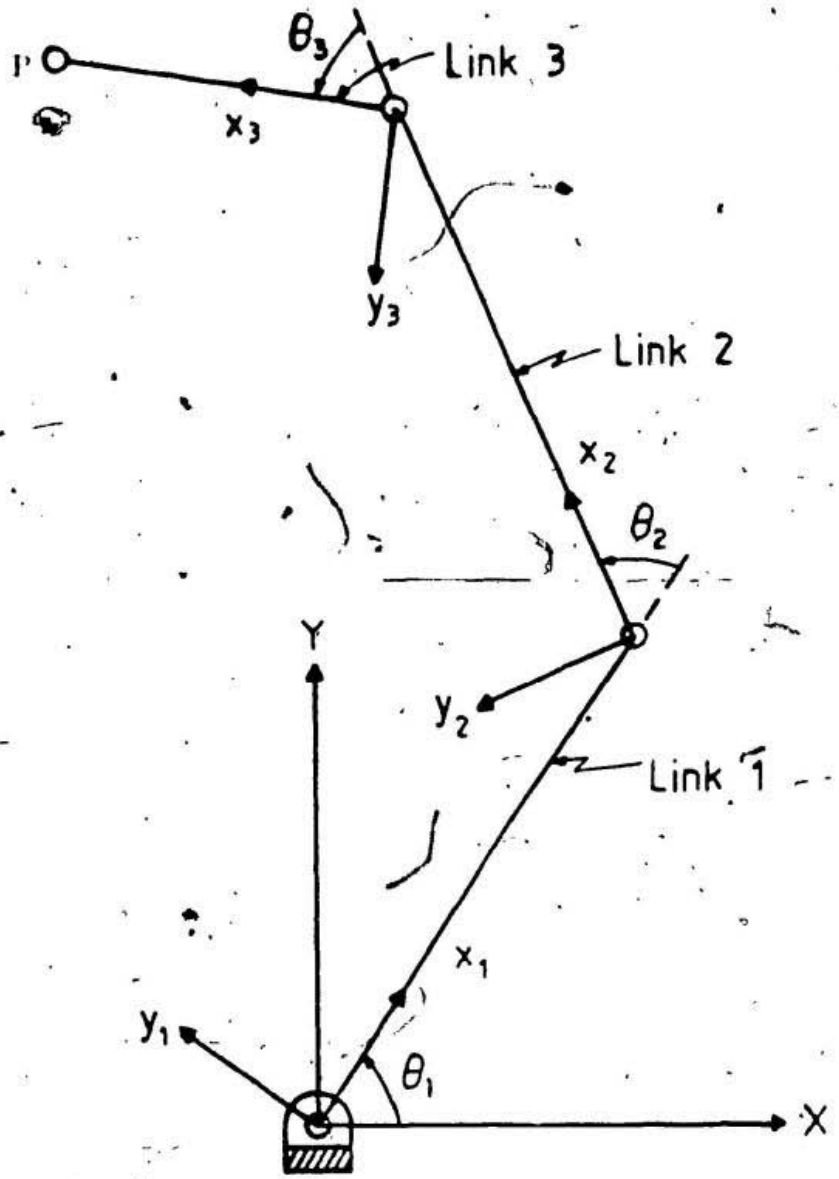

Fig. 2.7: A Planar Manipulator

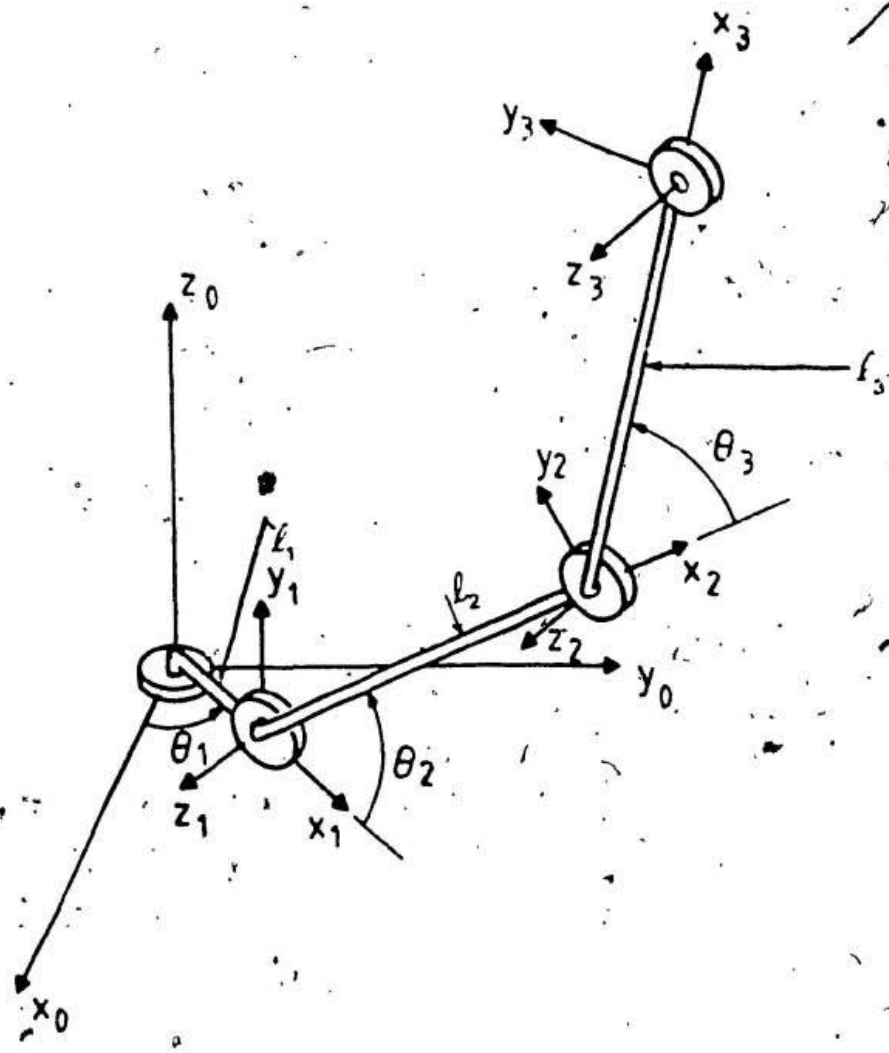

Fig. 2.8: Kinematic Model of the Robot

are θ_1, θ_2 and θ_3 respectively. Table 2.6 lists the 4 x 4 matrix parameters for this system.

This robot performs a task of welding a square plate of dimensions 0.5 m x 0.5 m. The position of this plate with respect to the inertial frame is shown in the Fig. 2.9. The coordinates of some of the important points are shown in the Table 2.7. In order to perform the job, the gripper of the robot must trace the profile of the plate. For this, forty points on the profile of the plate were selected, i.e., the position vectors of all these forty points were known with respect to the inertial frame. For each of these points the nominal joint angles through which the robot must be rotated were to be determined.

2.3.4 The Results and Discussion of the Open Loop Mechanisms

The results obtained by the modified Newton-Raphson technique and the complex method are shown in the Table 2.8 for the first example. These results indicate that the final outcome of both the methods of solution is almost the same. In the second example, the input angles for all the three links were calculated when the robot traced the profile of the plate. The results obtained are shown in Table 2.9 and Figs. 2.10 - 2.12. Refering to Fig. 2.10, it is found that

Table 2.6

Kinematic Parameters of the Robot

	Link		
	1	2	3
θ (degrees)	θ_1	θ_2	θ_3
L (meters)	0	1.016	1.5113
α (degrees)	90	0	0
H (meters)	0	0	0

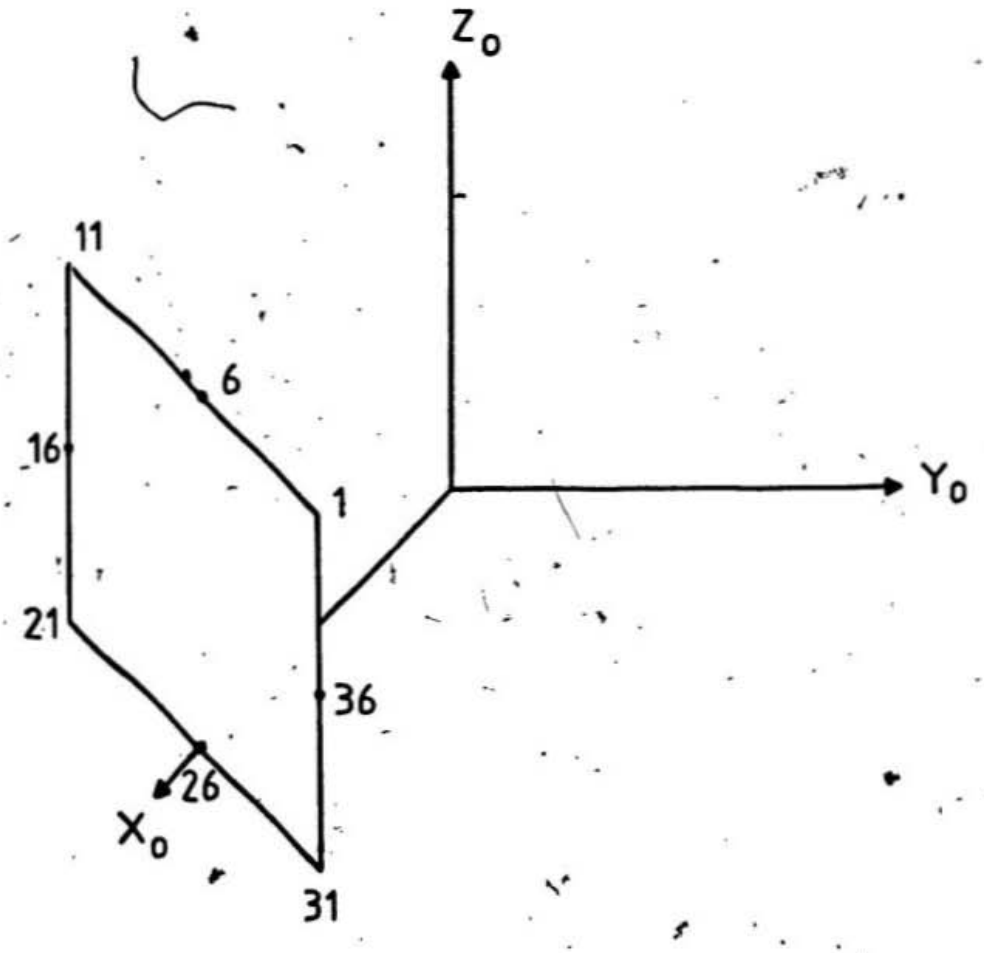

Fig. 2.9: Location of the Plate with Respect to the Inertial Coordinate System

Table 2.7

The Coordinates of Some Important Points on the Plate

Point	Coordinates (meters)		
	X_o	Y_o	Z_o
1	2.00	0.25	0.50
6	2.00	0.00	0.50
11	2.00	-0.25	0.50
16	2.00	-0.25	0.25
21	2.00	-0.25	0.00
26	2.00	0.00	0.00
31	2.00	0.25	0.00
36	2.00	0.25	0.25

Table 2.8

Comparison of Modified Newton-Raphson Technique and Complex Optimization Technique

No.	The coordinates of the Point P (Fig. 2.7). meters		Angle θ_1 degrees		Angle θ_2 degrees		Angle θ_3 degrees	
	X	Y	Newton-Raphson	Complex Method	Newton-Raphson	Complex Method	Newton-Raphson	Complex degress[3]
1	15.403	3.415	57.296	57.296	-57.296	-57.296	-90.000	-90.000
2	26.000	0.000	0.006	0.000	- 0.069	0.000	0.034	0.000
3	18.385	18.385	44.971	45.034	0.057	-0.051	0.029	0.034
4	23.070	7.070	45.000	45.000	-45.000	-45.000	0.000	0.000
5	17.071	1.071	45.000	45.000	-45.000	-45.000	-90.00	-90.000

Table 2.9

The Variation of Input Angles for 10 Points

No.	Position Coordinate (meters)			Angles (Degrees)		
	X	Y	Z	θ_1	θ_2	θ_3
1	2	0.25	0.5	7.125	57.436	288.929
2	2	0.05	0.5	1.432	58.298	287.750
3	2	-0.15	0.5	-4.289	58.010	288.142
4	2	-0.25	0.4	-7.125	55.870	287.163
5	2	-0.25	0.2	-7.125	51.827	284.834
6	2	-0.25	0.0	-7.125	46.662	284.063
7	2	-0.05	0.0	-1.432	47.418	282.912
8	2	0.15	0.0	4.283	47.167	283.295
9	2	0.25	0.1	7.125	49.379	284.256
10	2	0.25	0.3	7.125	53.996	285.801

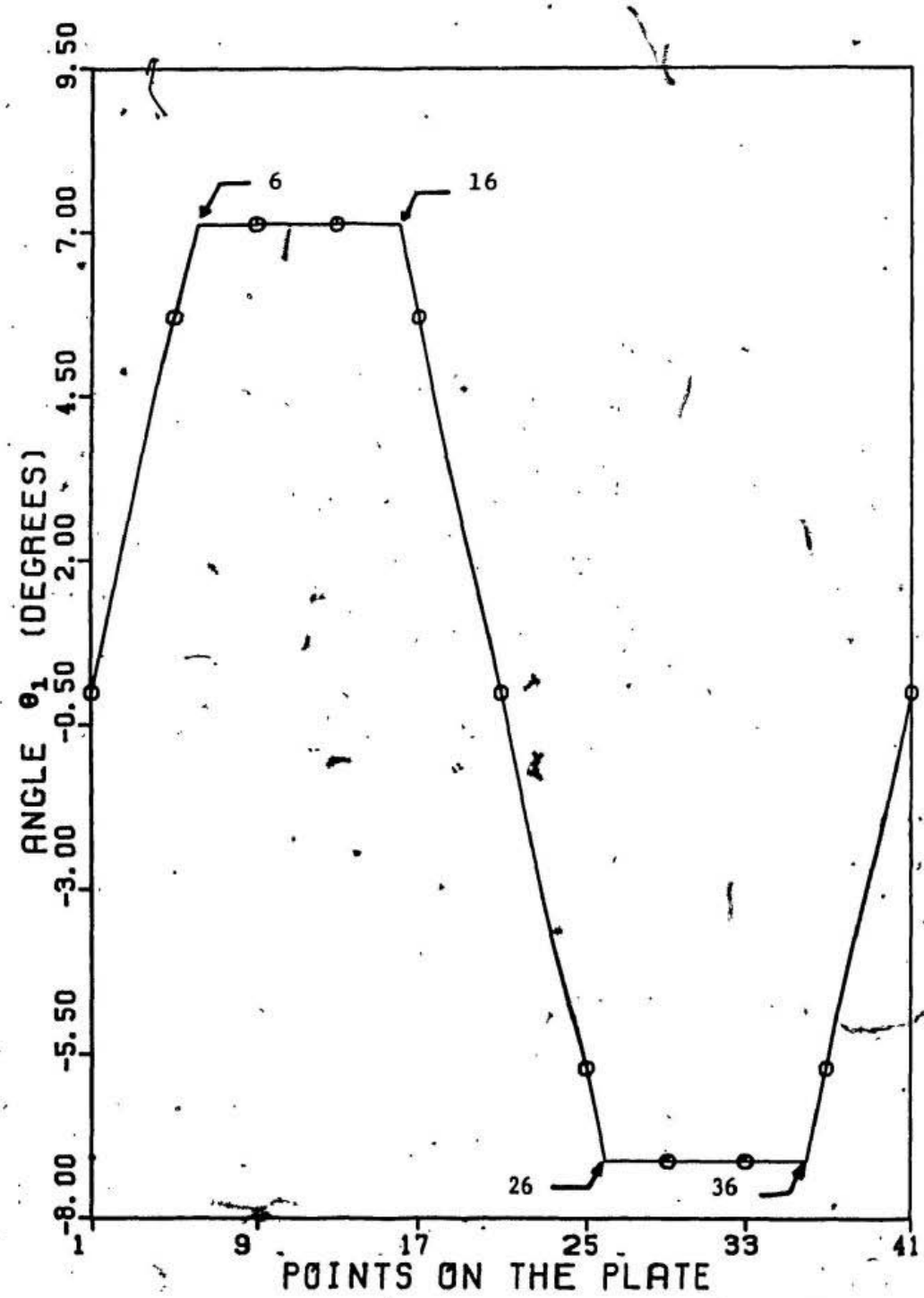

Fig. 2.10: Variation of θ_1 at Various Points on the Plate

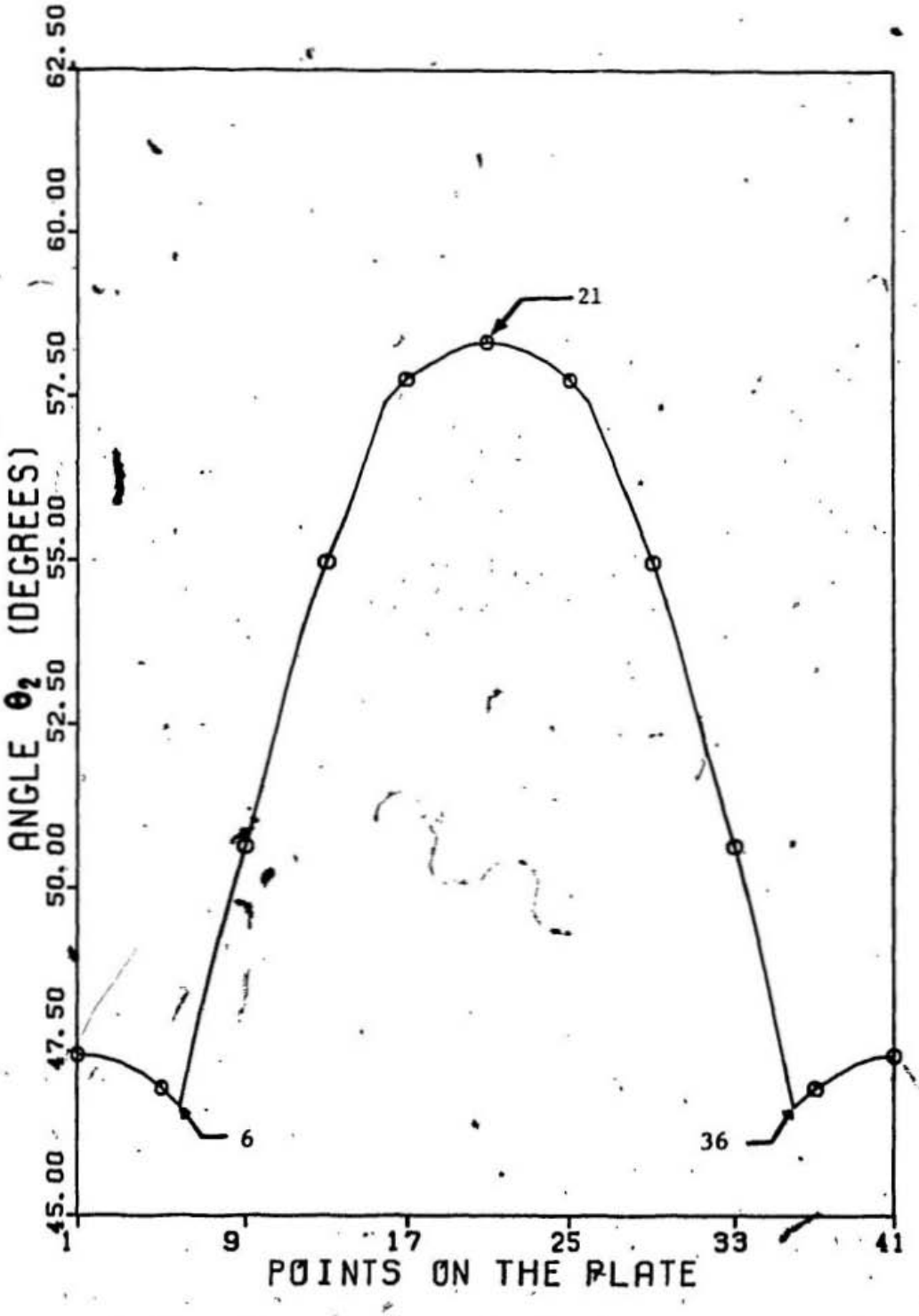

Fig. 2.11: Variation of θ_2 at Various Points on the Plate

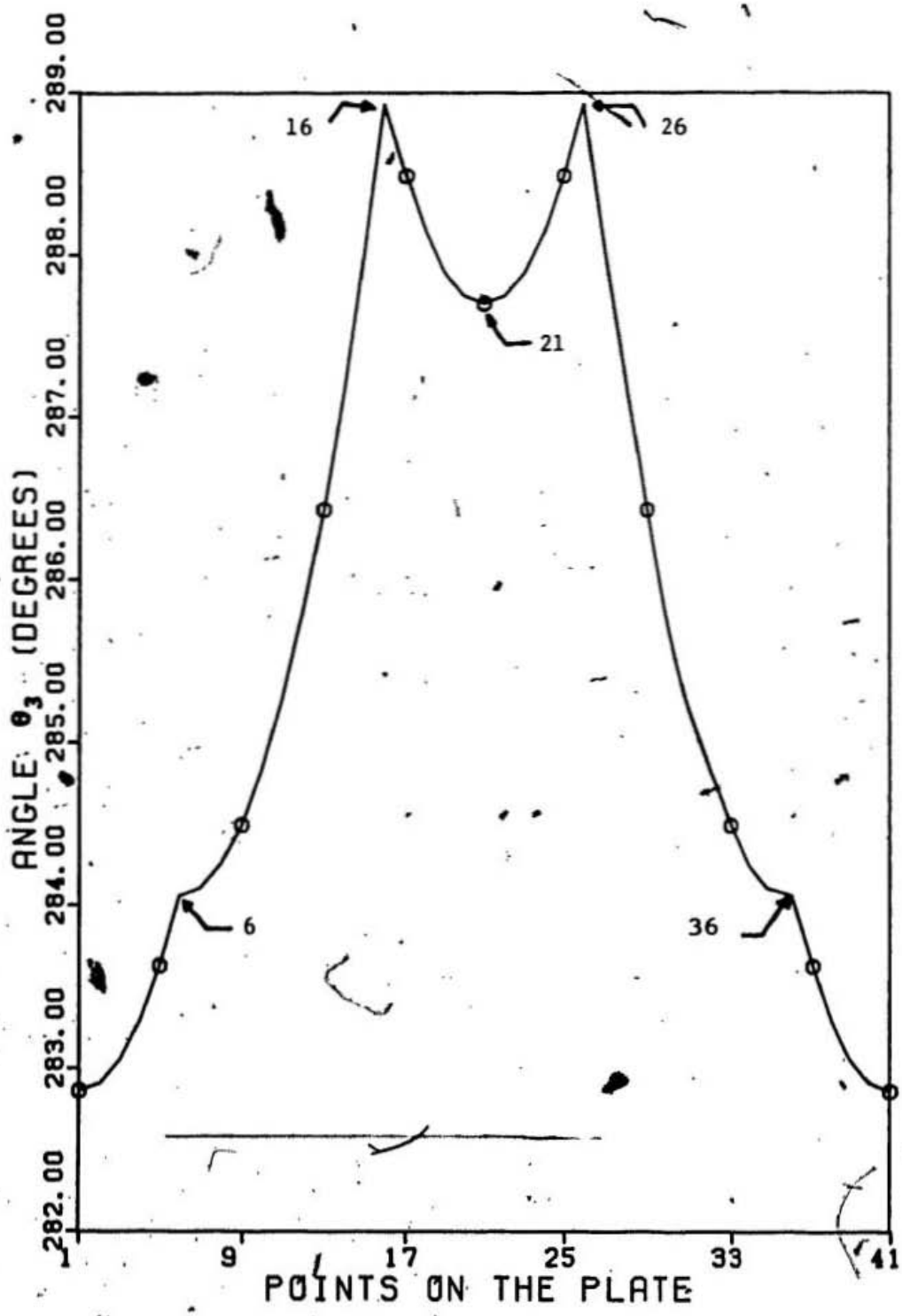

Fig. 2.12: Variation of θ_3 at Various Points on the Plate

the input angle remains constant when the gripper traces the points 6-16 and 26-36. Moreover, anti-symmetry is observed between points 1-21 and 21-40. Also, the variation of the angles are abrupt at points 6, 16, 26 and 36. If we refer to Fig. 2.11, it can be noted that the angle θ_2 reaches a maximum at the point 21 and minimum at points 6 and 36. Here, the variation is symmetrical about the point 21. Refering to Fig. 2.12, it can be seen that the angle θ_3 reaches maximum at points 16 and 26 and has a local minimum at the point 21. Thus, it is observed that the most significant points on the plate are 6, 16, 21, 26 and 36 where the angles either go through a maximum or minimum or change abruptly. All these points are either corners or mid-points of the sides of the square plate.

It is essential to note that it is always possible to get multiple solutions in the displacement analysis of open loop mechanisms. As an example, when the three-dimensional robotic manipulator problem was solved using modified Newton-Raphson technique, three different solutions were obtained depending on different starting vectors. These solutions for one of the points on the plate (point 1 in Fig. 2.9) are shown in Table 2.10. Among these solutions, the second solution is better than the other two because this is closer to the previous configuration of the robot, i.e. when the end effector was at the 40th point.

Table 2.10

Multiple Solutions for the Position (2, 0.25, 0.5) on the Plate

	Angles (Degree)		
	θ_1	θ_2	θ_3
Solution 1	7.125	330.429	71.078
Solution 2	7.125	57.436	288.929
Solution 3	187.125	209.571	288.929

*The previous configuration of the robot was

$\theta_1 = 7.125°$, $\theta_2 = 53.996°$, $\theta_3 = 285.801°$

But, while using complex optimization technique only the best solution was obtained. The other two solutions were eliminated because of the inclusion of the function $F_2(\underline{x})$ in the Eqn. (2.47). Thus, by using complex optimization technique, one can avoid the robot links having any abrupt motion that may give rise to large oscillations, which is an undesirable effect.

It is clear from the discussion above that the multiple solutions pose a very difficult problem when the modified Newton-Raphson technique is used. One has to use several starting vectors to obtain different solutions. For example, in Table 2.10, the solution 1 was obtained using a starting vector $\{5.730°, -57.296°, 57.296°\}^T$. It took 0.8 seconds CPU time on a VAX 11/785 digital computer. The second solution was obtained in 0.5 seconds. The starting vector in this case was $\{7.125°, 53.996°, 285.801°\}^T$. On the other hand, the complex method required 3 seconds but it yielded the final solution vector directly. It required 120 objective function evaluations. The reflection factor used was 1.3. The parameter "a" used in the Eqn. (2.47) was varied from 10^3 to 10^{10} so that $F(\underline{x})$ was minimum. In this particular case the times taken for each of the solutions in the Newton-Raphson technique were smaller than the complex method but it should not lead one to infer that the Newton-Raphson technique is always more efficient than

the complex method. Here, to start with, same starting vector was chosen for both the methods. The first solution was obtained using the Newton-Raphson technique and then another starting vector had to be given. At this stage, the CPU time for the second solution depends upon the relative location of the second starting vector and the second optima. If it is closer to the second optima then the time will be small othewise it may take much longer time or even never converge. The algorithm of the complex method is such that based on the starting vector and using the pseudo-random numbers, it uses several points in the domain in the random fashion. In this way it approaches the global optima. In addition, in the complex method when the region near the optima is found, the final complex collapses into the centroid. Thus, the objective function $F_1(\underline{X})$ in Eqn. (2.45) would have a much smaller value as compared to the ones obtained using the Newton-Raphson technique. In the present case, $F_1(X)$ using the complex method was of the order of 10^{-13} whereas those obtained using the Newton-Raphson technique were 10^{-6}. Another fact which is worth mentioning here is that any implicit or explicit constraints can be easily handled in the optimization methods which are quite difficult in either the modified Newton-Raphson technique or in any analytical techniques. In a manufacturing environment the constraints such as avoiding obstacles do occur, so in

such situations it would be a lot easier to use the present method. In summary, the present technique (techniques based on the optimization principles) has definite advantages over either the modified Newton-Raphson technique or any other analytical techniques due to the possible multiple solutions and the occurence of constraints in practical problems.

2.4 The Velocity Analysis of the Open Loop Mechanisms (Robotic Manipulators)

2.4.1 The Methods of the Velocity Analysis

The velocity analysis of robotic manipulators consists of determining the angular velocities, given the velocities of the gripper or the end effector. In this work the velocity analysis is carried out by two methods and is discussed in Sections 2.4.2 and 2.4.3.

The first method uses the modified Newton-Raphson technique. The reason for using this iterative technique is that for a given velocity of the end effector multiple solutions exist. This is because, in these cases, the number of equations are less than the number of unknowns. If, in the special case, the number of unknowns are 3 then the number of equations will be equal to the number of unknowns and then the exact solution can be found by solving these equations simultaneously. In all cases, the number of

equations are three which are due to the three components of the velocity of the end-effector.

In the next section, Section 2.4.2, the numerical method is discussed and in the Section 2.4.3, the exact solution for the robotic manipulator having three joint angles are obtained by solving the equations simultaneously.

2.4.2 The Modified Newton-Raphson Technique

In the Section 2.3.1 the displacement analysis of a robotic manipulator was carried out using the modified Newton-Raphson Technique. Recalling Eqn. (2.4) which defines the position vector of a point with respect to the inertial frame as

$$\underline{r}_i^{\,o} = [T_o^{\,i}]\,\underline{r}_i \,, \qquad (2.4)$$

the velocity of the same point, $\dot{\underline{r}}_i^{\,o}$, is obtained by differentiating this equation with respect to time. At this point it must be noted that $[T_o^{\,i}]$ is a function of $x_1, x_2 \ldots x_i$. Therefore, by the chain rule for differentiation we get,

$$\dot{\underline{r}}_i^{\,o} = \sum_{j=1}^{m} \frac{\partial [T_o^{\,i}]}{\partial x_j} \dot{x}_j \underline{r}_i = \sum_{j=1}^{m} D_j \dot{x}_j \underline{r}_i^{\,o} = \omega_i \underline{r}_i^{\,o} \qquad (2.53)$$

where $\omega_i = \sum_{j=1}^{m} D_j \dot{x}_j$ is the velocity matrix for link i.

After the displacement analysis is completed, the $\underline{r}_i^{\,o}$ can be defined as $\underline{r}_i^{\,o^*}$ and the above equation is written as

$$\dot{\underline{r}}_i^{\,o} = \sum_{j=1}^{m} D_j \dot{x}_j \underline{r}_i^{\,o^*} \qquad (2.54)$$

But from Eqns. (2.36) and (2.37)

$$D_j \underline{r}_i^{\,o^*} = \frac{\partial \underline{r}_i^{\,o^*}}{\partial x_j} \qquad (2.55)$$

Therefore, Eqn. (2.54) now becomes

$$\dot{\underline{r}}_i^{\,o^*} = \sum_{j=1}^{m} \frac{\partial \underline{r}_i^{\,o^*}}{\partial x_j} \dot{x}_j \qquad (2.56)$$

The Eqn. (2.56) can be written in the matrix form as

$$\dot{\underline{r}}_i^{\,o} = \left[\frac{\partial \underline{r}_i^{\,o*}}{\partial \underline{x}}\right] \dot{\underline{x}} \qquad (2.57)$$

The quantity $\left[\frac{\partial \underline{r}_i^{\,o*}}{\partial \underline{x}}\right]$ was defined in Eqn. (2.37) and is equal to the matrix [A] which was previously used in the displacement analysis. The Eqn. (2.56) now becomes

$$\dot{\underline{r}}_i^{\,o} = [A]\, \dot{\underline{x}} \qquad (2.58)$$

Multiplying by $[A^T]$ we get

$$[A]^T \dot{\underline{r}}_i^{\,o} = [A]^T[A]\, \dot{\underline{x}} \qquad (2.59)$$

So, the angular velocities can be obtained using

$$\dot{\underline{x}} = [[A]^T[A]]^{-1}[A]^T \dot{\underline{r}}_i^{\,o} \qquad (2.60)$$

Referring to the Eqn. (2.60), the vector $\{\dot{x}\}$ is the

velocity of the joint coordinates, i.e. the angular velocities of the links or the linear velocity in case of a prismatic joint. This can be easily calculated because the vector $\dot{r}$ is the specified velocity of the end effector, the other matrices in Eqn. (2.60) have already been calculated while carrying out the displacement analysis and can be stored in the computer memory.

2.4.3 The Velocity Analysis of Mechanisms with Three Degrees of Freedom

Velocity analysis for a T3R3 model robotic manipulator can be easily carried out analytically, since the number of unknown components of the angular velocities and the number of linear equations are equal. The kinematic model of T3R3 robot is shown in the Fig. 2.8

The position of the end effector of this robot can be given as [36],

$$x = l_2 \cos\theta_2 \cos\theta_1 + l_3 \cos(\theta_2 + \theta_3) \cos\theta_1$$
$$y = l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1$$
$$z = l_2 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3) \qquad (2.61)$$

Here l_1 does not occur because it is a fictitious link for this particular robot. Then, the velocity of the end effector can be obtained by differentiating the above

equation with respect to time and it can be written as

$$\dot{x} = -[l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_1$$
$$-[l_2 \sin\theta_2 \cos\theta_1 + l_3 \sin(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_2$$
$$-[l_3 \sin(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_3$$
$$\dot{y} = [l_2 \cos\theta_2 \cos\theta_1 + l_3 \cos(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_1$$
$$-[l_2 \sin\theta_2 \sin\theta_1 + l_3 \sin(\theta_2 + \theta_3) \sin\theta_1)\dot{\theta}_2$$
$$-[l_3 \sin(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_3$$
$$\dot{z} = [l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3)]\dot{\theta}_2$$
$$+[l_3 \cos(\theta_2 + \theta_3)]\dot{\theta}_3 \qquad (2.62)$$

The above form of the velocity equation can be written in the matrix form as

$$\begin{bmatrix} 1 \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & v_{11} & v_{12} & v_{13} \\ 0 & v_{21} & v_{22} & v_{23} \\ 0 & v_{31} & v_{32} & v_{33} \end{bmatrix} \begin{bmatrix} 1 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \qquad (2.63)$$

where the expressions for $v_{11}, \ldots, v_{33}$ can be seen in the Appendix A. Eqn. (2.59) can be rewritten as

$$\begin{bmatrix} 1 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & v_{11} & v_{12} & v_{13} \\ 0 & v_{21} & v_{22} & v_{23} \\ 0 & v_{31} & v_{32} & v_{33} \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \qquad (2.64)$$

Thus, the vector $\underline{\dot{\theta}}$ can be easily obtained.

2.5 The Acceleration Analysis of the Open Loop Mechanisms (Robotic Manipulators)

2.5.1 The Methods of the Acceleration Analysis

Similar to the velocity analysis one can use the numerical techniques to obtain the joint accelerations, and if the number of unknowns are three, then one can obtain the exact solution by simultaneously solving the acceleration equations. The two methods, as before, are discussed next.

2.5.2 The Modified Newton-Raphson Technique

The acceleration analysis consists of determining the accelerations of the input-angles, given the

accelerations of the end effector in the global coordinate system. Differentiating Eqn. (2.53) with respect to time, the acceleration of the end effector becomes [8]

$$\ddot{\underline{r}}_i^{\,o} = \sum_{j=1}^{m} \frac{\partial \dot{\underline{r}}_i^{\,o}}{\partial \dot{x}_j} \ddot{x}_j + \sum_{j=1}^{m} \frac{\partial \dot{\underline{r}}_i^{\,o}}{\partial x_j} \dot{x}_j \qquad (2.65)$$

Differentiating the Eqn. (2.53) with respect to $\dot{x}_j$, one can write [8]

$$\frac{\partial \dot{\underline{r}}_i^{\,o}}{\partial \dot{x}_j} = D_j \, \underline{r}_i^{\,o} \qquad (2.66)$$

Therefore, the first term in Eqn. (2.65) becomes

$$\sum_{j=1}^{m} \frac{\partial \dot{\underline{r}}_i^{\,o}}{\partial \dot{x}_j} \ddot{x}_j = \sum_{j=1}^{m} D_j \, \ddot{x}_j \, \underline{r}_i^{\,o} \qquad (2.67)$$

By substituting the value of $\dot{\underline{r}}_i^{\,o}$ from Eqn. (2.53), the second term of the Eqn. (2.65) becomes,

$$\sum_{j=1}^{m} \frac{\partial \dot{\underline{r}}_i^{\,o}}{\partial x_j} \dot{x}_j = \sum_{j=1}^{m} \sum_{k=1}^{m} \frac{\partial}{\partial x_j} ([D_k][T_o^{\,i}]) \, \underline{r}_i \, \dot{x}_k \, \dot{x}_j \qquad (2.68)$$

The partial derivative of $([D_k][T_o^i])$ with respect to x_j has been shown to be [8]

$$\frac{\partial}{\partial x_j}([D_k][T_o^i]) = \overline{[D_k][D_j]}[T_o^i] = \begin{cases} [D_k][D_j][T_o^i] \text{ if } x_k \text{ occurs before } x_j \text{ in } [T_o^i] \\ [D_j][D_k][T_o^i] \text{ if } x_j \text{ occurs before } x_k \text{ in } [T_o^i] \end{cases} \tag{2.69}$$

Thus, the linear acceleration can be given as

$$\ddot{\underline{r}}_i^o = \sum_{j=1}^{m} [D_j]\, \ddot{x}_j\, \underline{r}_i^o + \sum_{j=1}^{m}\sum_{k=1}^{m} \overline{[D_k][D_j]}\, \dot{x}_k\, \dot{x}_j\, \underline{r}_i^o \tag{2.70}$$

Writing Eqn. (2.70) in the matrix form and using Eqn. (2.41) one can write

$$\ddot{\underline{r}}_i^o = [A]\, \ddot{\underline{x}} + [DD]\, \underline{r}_i^o \tag{2.71}$$

where $[DD] = \sum_{j=1}^{m}\sum_{k=1}^{m} \overline{[D_k]\,[D_i]}\, \dot{x}_k\, \dot{x}_j$. Multiplying Eqn. (2.71) by $[A]^T$ yields:

$$[A]^T\, \ddot{\underline{r}}_i^o = [[A]^T\,[A]]\, \ddot{\underline{x}} + [A]^T\,[DD]\, \underline{r}_i^o \tag{2.72}$$

Eqn. (2.72) can be solved for the angular acceleration $\ddot{\underline{x}}$ as

$$\ddot{\underline{x}} = [[A]^T[A]]^{-1} [A]^T [\ddot{\underline{r}}_i^{\circ} - [DD] \underline{r}_i^{\circ}] \qquad (2.73)$$

2.5.3 The Acceleration Analysis of Mechanisms with Three Degrees of Freedom

Acceleration analysis for a T3R3 model robotic manipulator can also be easily derived from the velocity equations, Eqn. (2.62). Differentiating this equation with respect to time, the acceleration equations can be written as

$$\begin{aligned}
\ddot{x} = & - [l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1] \ddot{\theta}_1 \\
& - [l_2 \sin\theta_2 \cos\theta_1 + l_3 \sin(\theta_2 + \theta_3) \cos\theta_1] \ddot{\theta}_2 \\
& - [l_3 \sin(\theta_2 + \theta_3) \cos\theta_1] \ddot{\theta}_3 \\
& + 2[l_2 \sin\theta_1 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3) \sin\theta_1] \dot{\theta}_1 \dot{\theta}_2 \\
& - 2[l_3 \cos(\theta_2 + \theta_3) \cdot \cos\theta_1] \dot{\theta}_2 \dot{\theta}_3 \\
& + 2[l_3 \sin\theta_1 \sin(\theta_2 + \theta_3)] \dot{\theta}_1 \dot{\theta}_3 \\
& - [l_2 \cos\theta_1 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \cos\theta_1] \dot{\theta}_1^2 \\
& - [l_2 \cos\theta_1 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \cos\theta_1] \dot{\theta}_2^2 \\
& - [l_3 \cos(\theta_2 + \theta_3) \cos\theta_1] \dot{\theta}_3^2
\end{aligned} \qquad (2.74)$$

$$\ddot{y} = [l_2 \cos\theta_2 \cos\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\ddot{\theta}_1$$
$$- [l_2 \sin\theta_2 \sin\theta_1 + l_3 \sin(\theta_2 + \theta_3) \sin\theta_1]\ddot{\theta}_2$$
$$- [l_3 \sin(\theta_2 + \theta_3) \sin\theta_1]\ddot{\theta}_3$$
$$- [l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_2$$
$$- [l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_2^2$$
$$- [l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_3^2$$
$$- 2[l_2 \sin\theta_2 \cos\theta_1 + l_3 \sin(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_1\dot{\theta}_2$$
$$- 2[l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_2\dot{\theta}_3$$
$$- 2[l_3 \sin(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_1\dot{\theta}_3 \qquad (2.75)$$

and

$$\ddot{z} = -[l_2 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3)]\dot{\theta}_2^2 - l_3 \sin(\theta_2 + \theta_3)\dot{\theta}_2\dot{\theta}_3$$
$$+ [l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3)]\ddot{\theta}_2$$
$$+ [l_3 \cos(\theta_2 + \theta_3)]\ddot{\theta}_3 - l_3 \sin(\theta_2 + \theta_3)\dot{\theta}_3\dot{\theta}_2$$
$$- l_3 \sin(\theta_2 + \theta_3)\dot{\theta}_3^2 \qquad (2.76)$$

The above equations can be written in the matrix form as

$$\begin{bmatrix} 1 \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \begin{bmatrix} 1 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} \qquad (2.77)$$

Since the left hand side of the equation is known, the angular accelerations are obtained as

$$\begin{bmatrix} 1 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \qquad (2.78)$$

The expressions for $A_{21}, A_{22}, \ldots, A_{44}$ are given in the Appendix A.

2.6 Numerical Example

The two techniques for obtaining the velocities and accelerations are illustrated by applying them to a T3R3 model robotic manipulator. If this robot is subjected to a task of welding a square plate, the end effector moves with constant acceleration until the required velocity is reached; then the acceleration becomes zero. When the end effector is about to finish one side of the plate, the velocity starts decreasing with a constant deceleration. This velocity profile shown in Fig. 2.13 is repeated on all the four sides.

The angular velocity and acceleration results obtained by the modified Newton-Raphson technique and the exact solution method were the same and are shown in Figs. 2.14-2.19. The displacement analysis results for this problem were shown in Figs. 2.10 - 2.12. The displacement analysis results showed abrupt changes or maxima or minima at the corner points or the mid-points of the plate. The angular velocity results show this type of behavior not only at the above mentioned points but also at points where the end effector velocity undergoes sharp changes. The angular acceleration results also show a similar pattern.

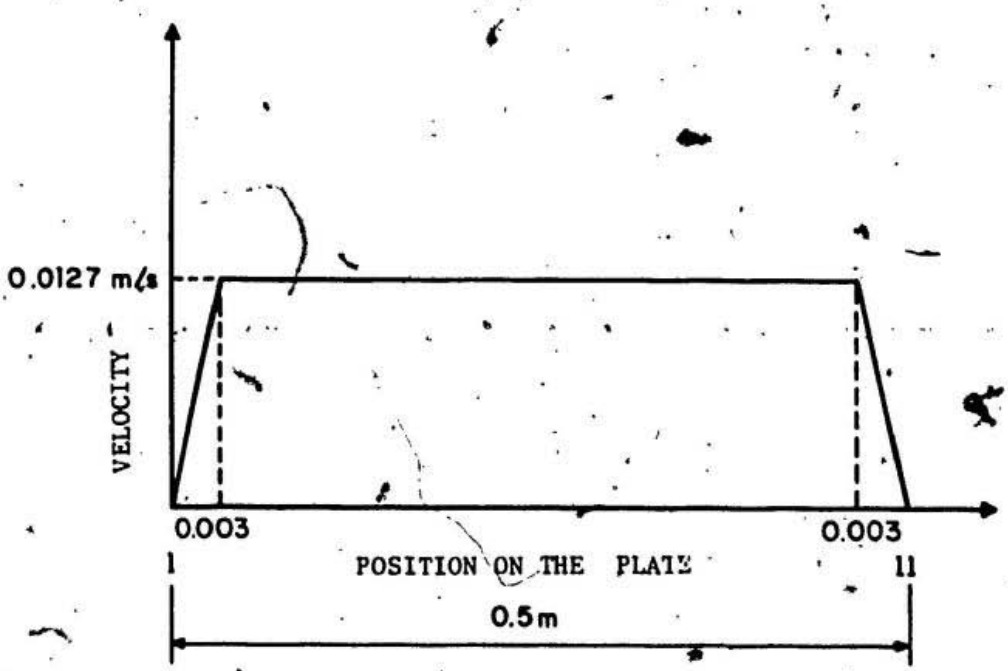

Fig. 2.13: Velocity Profile of the End Effector

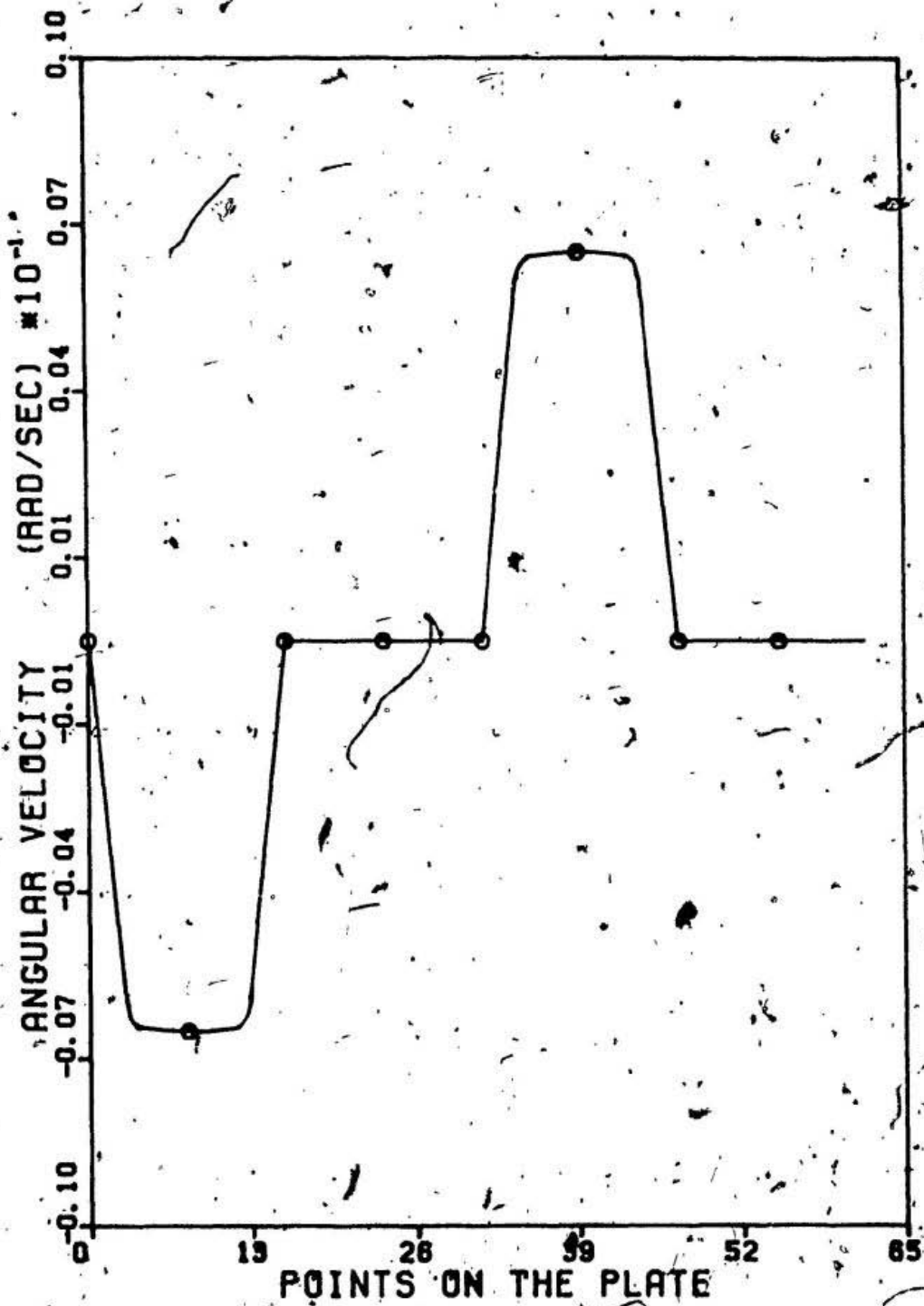

Fig. 2.14: The Angular Velocity $\dot{\theta}_1$ at Various Points on the Plate

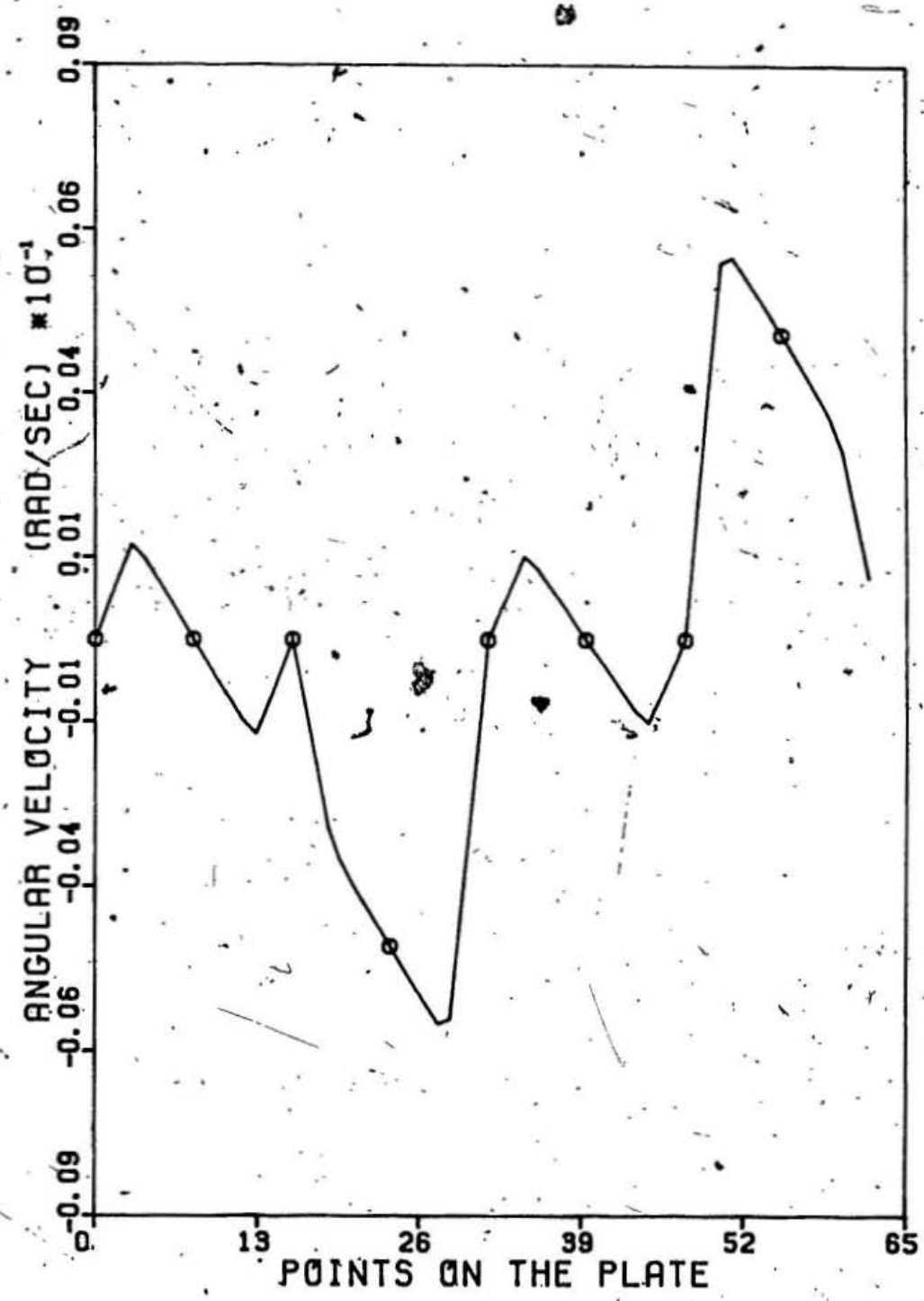

Fig. 2.15: The Angular Velocity $\dot{\theta}_2$ at Various Points on the Plate

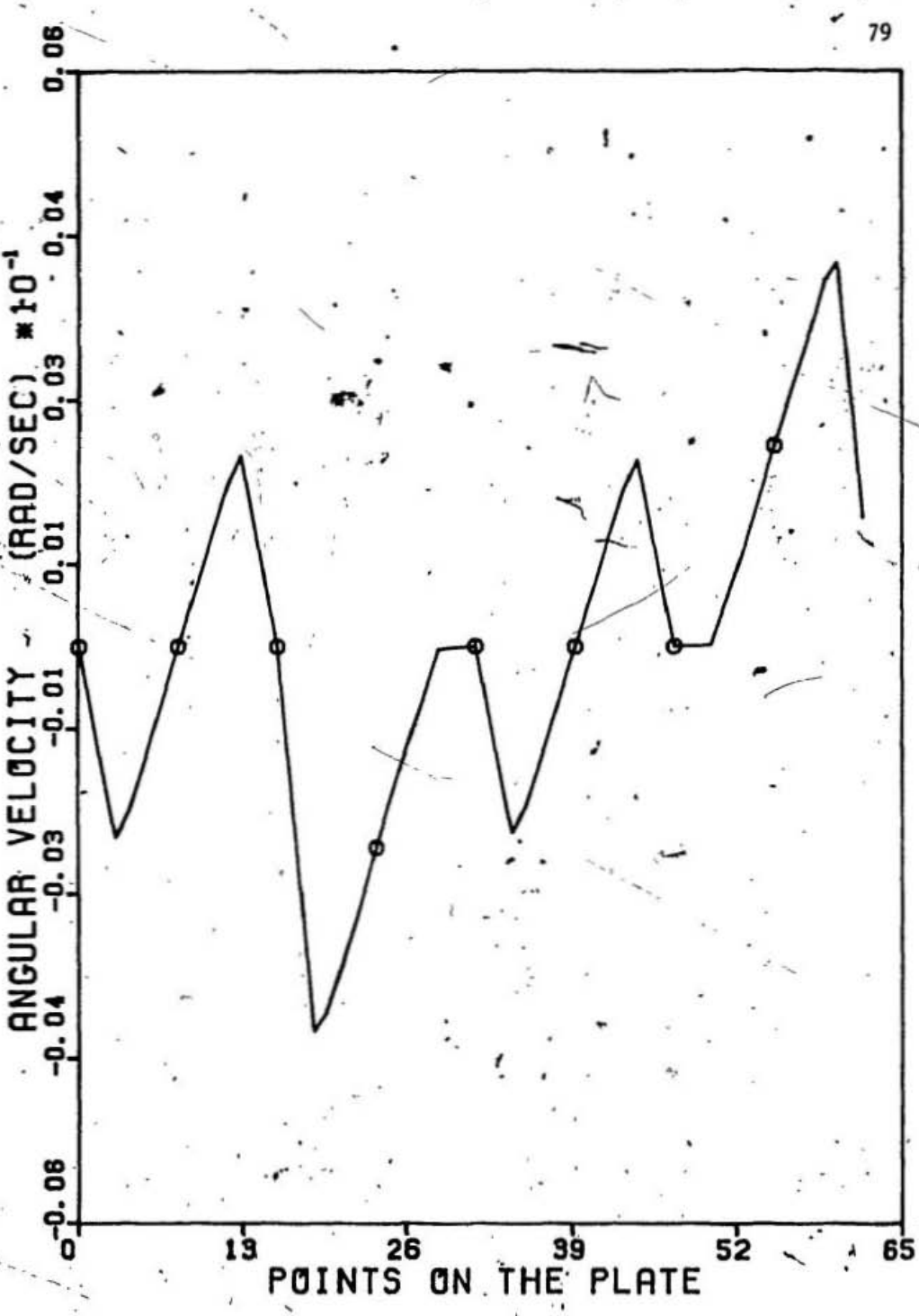

Fig. 2.16: The Angular Velocity $\dot{\theta}_3$ at Various Points on the Plate

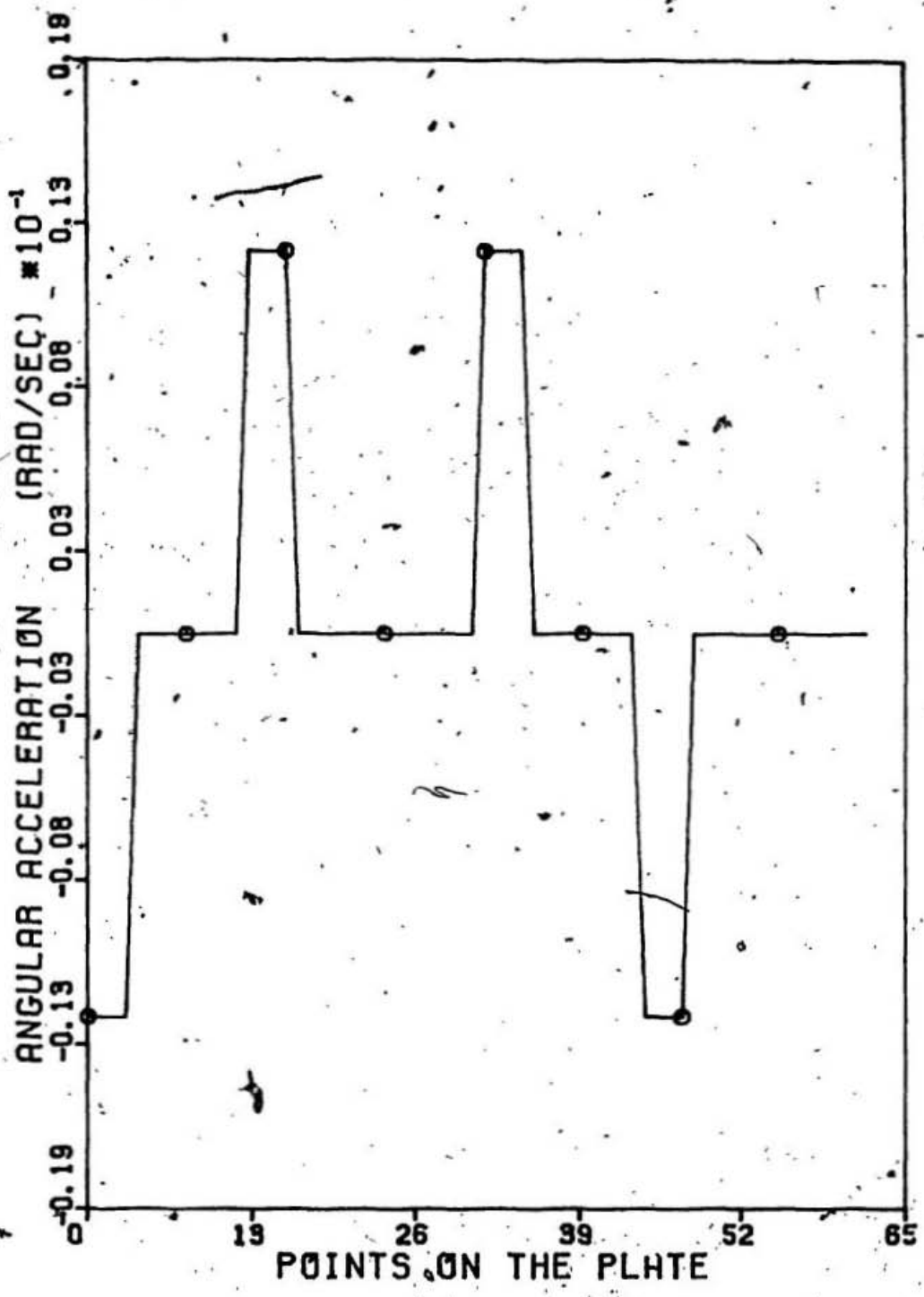

Fig. 2.17: The Angular Acceleration $\ddot{\theta}_1$ at Various Points on the Plate

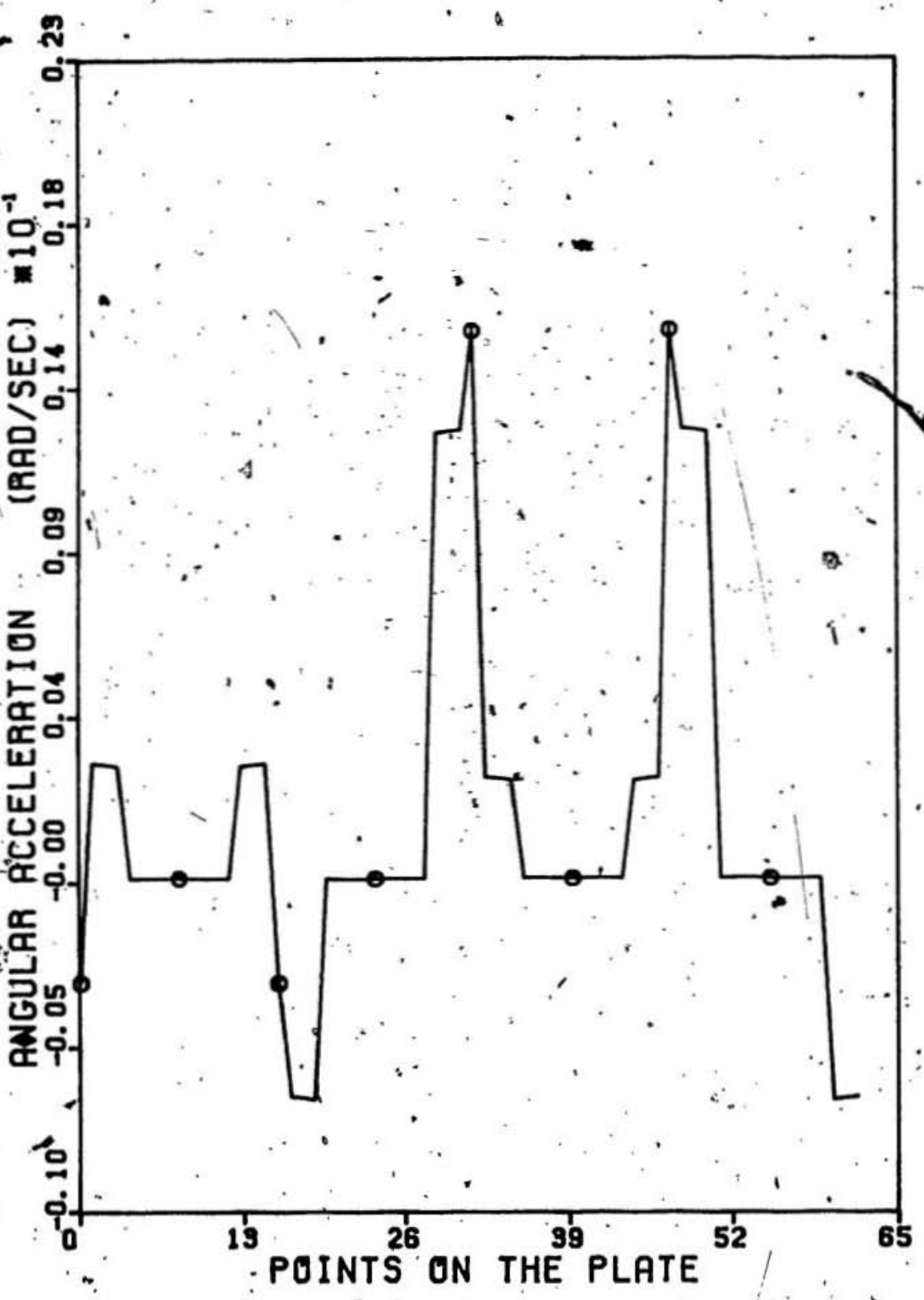

Fig. 2.18: The Angualr Acceleration $\ddot{\theta}_2$ at Various Points on the Plate

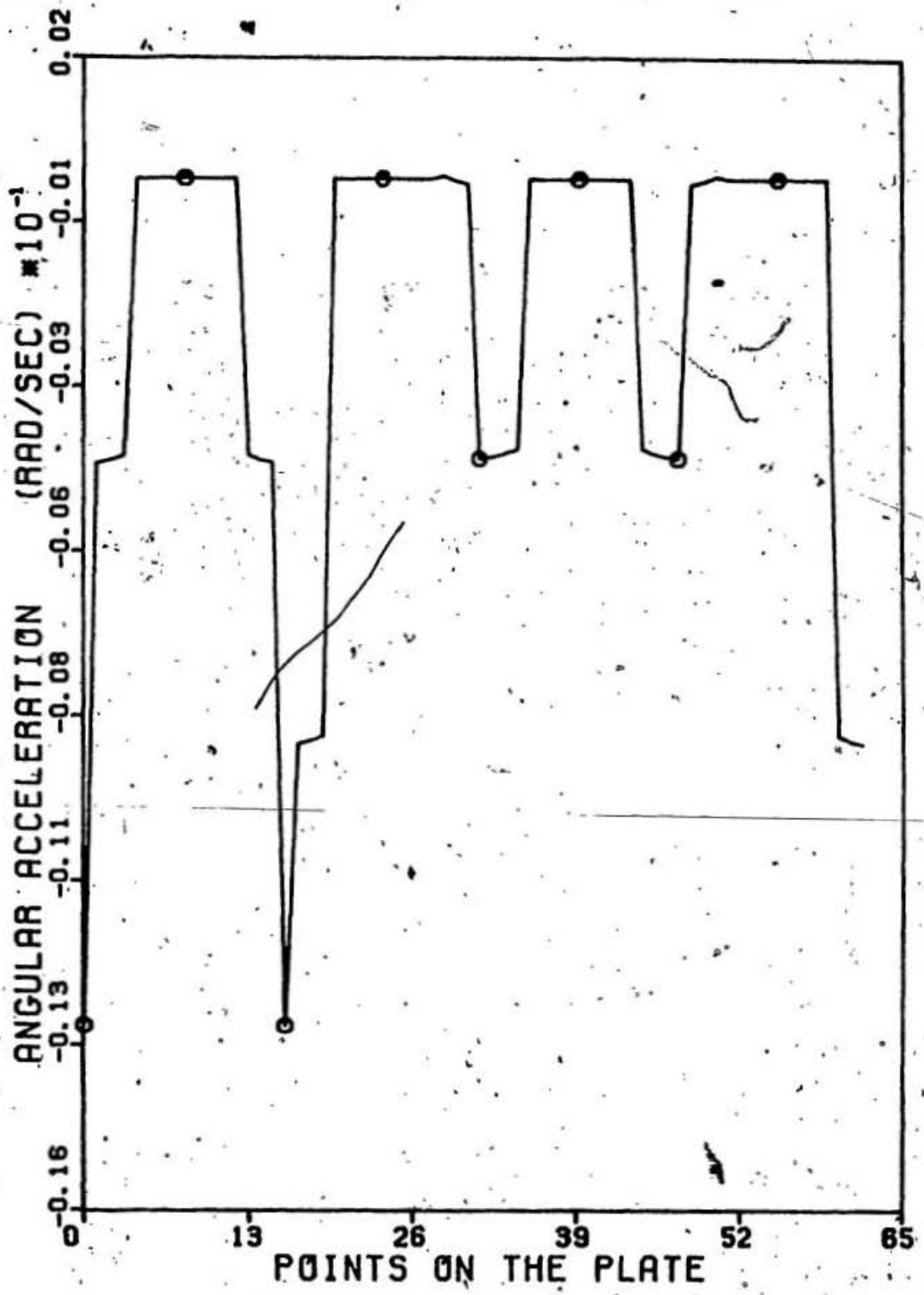

Fig. 2.19: The Angular Acceleration $\ddot{\theta}_3$ at Various Points on the Plate

2.7 Conclusions

In this chapter, the kinematic analysis of the closed and open loop mechanism was carried out. At first, the displacement analysis of the planar and three-dimensional closed loop mechanisms was done and then the open loop mechanisms were analyzed. Both, the Newton-Raphson technique and the optimization principles were used to solve the inverse displacement analysis problems. Since the robotic manipulators are open loop mechanisms, the velocity and the acceleration analysis was performed for the open loop mechanisms only and in these analyses the solutions were obtained analytically as well as by using the modified Newton-Raphson technique. Based on the work carried out in this chapter, the following conclusions can be drawn:

1. The Hooke and Jeeves method can be succesfully used to perform the displacement analysis of the planar and space closed loop mechanisms.
2. The complex optimization method can yield better solution as compared to the modified Newton-Raphson technique for solving inverse displacement analysis problems involving the robotic manipulators.
3. If the open loop mechanisms or robotic manipulators have three degrees of freedom then the exact values of the angular velocities and angular accelerations can be obtained.

4. The modified Newton-Raphson technique or other numerical methods such as the optimization principles have to be used in situations where the degrees of freedom are different from three, in space mechanisms or robotic manipulators.

CHAPTER 3

THE DYNAMIC ANALYSIS OF THE ROBOTIC MANIPULATORS

3.1 Introduction

In this investigation, the kinematic and dynamic analysis of the robotic manipulators are carried out by first obtaining the positions, velocities, and the accelerations of the various links in Chapter 2, and this information is used to study the dynamics of these type of systems in this chapter. In ordinary structures, the system dynamic matrices depend upon the geometry and other mechanical porperties of its components but, in the case of robotic manipulators, the angular orientations, velocities and accelerations also contribute towards these matrices. For example, the stiffness matrix is expressed as a sum of three matrices, one of which is the structural stiffness matrix, and the other two depend upon the kinematic parameters. To evaluate the structural stiffness matrix, the spatial orientation of all the links have to be known. Therefore, four new methods of finding the matrix of direction cosines of the local axes are discussed in the Section 3.2.6. The study of the variation of the natural frequencies due to the selection of the design parameters is discussed in Section 3.3. Finally, a

comparative study of the methods of dynamic condensation of the system matrices is carried out in Section 3.4.

3.2 Formulation of the Dynamic Equation of Motion

3.2.1 Lagrange's Equation

The general form of Lagrange's equation for a system with n independent generalized coordinates x_1 through x_n is expressed as

$$\frac{d}{dt}\left[\frac{\partial KE}{\partial \dot{x}_i}\right] - \frac{\partial KE}{\partial x_i} + \frac{\partial PE}{\partial x_i} = F_i \quad , \quad i = 1, \ldots , n \qquad (3.1)$$

where,

KE is the system kinetic energy in terms of the generalized coordinates $(x_j, \dot{x}_j)$,

PE is the system potential energy in terms of the generalized coordinates x_j,

and

F_i is the generalized force.

Now, this Lagrange's equation is written for each individual link of the robot separately, leaving the inter-joint constraint forces as unknowns.

3.2.2 The Link Kinetic Energy:

The translational kinetic energy of the gth grid point (node) of the ith link is given as

$$(KE)_{ig} = \frac{1}{2} m_{ig} [(\dot{x}^o_{ig})^2 + (\dot{y}^o_{ig})^2 + (\dot{z}^o_{ig})^2] = \frac{1}{2} m_{ig} \{v_{ig}\}^T \{v_{ig}\} \quad (3.2)$$

where, m_{ig} is the mass of the gth grid point of the ith link; $\dot{x}^o_{ig}$, $\dot{y}^o_{ig}$, $\dot{z}^o_{ig}$ are the absolute velocities of the node with respect to the inertial frame in the X_o, Y_o and Z_o directions respectively; and $\{v_{ig}\}$ is the velocity vector of the gth grid point. The rotary inertias of the individual grid points are neglected since it is a common practice in the case of the design of robotic manipulators [1].

If the usual form of the kinetic energy, $\frac{1}{2} m\{v\}^T\{v\}$, is used then, the terms that result from Lagrange's equations have the undesirable characteristic that the time invariant portions of the terms cannot be separated from the time-varying parts [1]. In order to accomplish the task of writing them separately, the kinetic energy of the link is expressed in a form similar to that proposed by Uicker [18] which is written as

$$(KE)_{ig} = \frac{1}{2} m_{ig} \, Tr[\{v_{ig}\}\{v_{ig}\}^T] \qquad (3.3)$$

where, $Tr[\{v_{ig}\}\{v_{ig}\}^T]$ = the sum or the diagional terms of the square matrix $[\{v_{ig}\}\{v_{ig}\}^T]$, which is mathematically expresses as

$$[\{v_{ig}\}\{v_{ig}\}^T] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \dot{x}^2 & \dot{x}\dot{y} & \dot{x}\dot{z} \\ 0 & \dot{y}\dot{x} & \dot{y}^2 & \dot{y}\dot{z} \\ 0 & \dot{z}\dot{x} & \dot{z}\dot{y} & \dot{z}^2 \end{bmatrix} \qquad (3.4)$$

The total kinetic energy for all the grid points on the ith link can be given as

$$(KE)_i = \sum_{g=1}^{NG(i)} (KE)_{ig} = \sum_{g=1}^{NG(i)} \frac{1}{2} m_{ig} \, Tr[\{v_{ig}\}\{v_{ig}\}^T] \qquad (3.5)$$

where NG(i) is the total number of grid points in the ith link.

3.2.3 The Link Potential Energy

The potential energy for the ith link associated with the structural elasticity is given as

$$(PE)_i^E = \frac{1}{2}\{p_i\}^T [k_i]\{p_i\} = \frac{1}{2} \sum_{\beta=1}^{NP(i)} \sum_{\gamma=1}^{NP(i)} (k_{i\beta\gamma})_1 \, p_{i\beta} \, p_{i\gamma} \tag{3.6}$$

where p_i is the elastic displacement which is assumed to be small when compared to the overall link dimensions and $(k_{i\beta\gamma})$, is an element of the stiffness matrix used in the finite element analysis.

The potential energy of the links due to gravitational effects is given as

$$(PE)_{ig}^G = -m_{ig}(g_x X_{ig}^o + g_y Y_{ig}^o + g_z Z_{ig}^o) \tag{3.7}$$

where

$(PE)_{ig}^G$ is the potential energy of the gth grid point of the ith link due to the gravity,

m_{ig} is the mass of the gth grid point of the ith link

and

g_x, g_y, g_z are the acceleration components of gravity.

In this work it is assumed that $g_x = g_y = 0$ and $g_z = -9.81$ m/sec^2 because the global X_o, and Y_o axes lie in the horizontal plane. X_{ig}^o, Y_{ig}^o, Z_{ig}^o are the inertial

coordinates of the gth grid point in the ith link. As before, this expression can also be written in terms of the trace of a matrix for example,

$$(PE)^{G}_{ig} = -m_{ig}\ \mathrm{Tr}(\{r^{o}_{ig}\}\{g\}^{T}) \qquad (3.8)$$

where

$$\{r^{o}_{ig}\} = \begin{bmatrix} 1 \\ X^{o}_{ig} \\ Y^{o}_{ig} \\ Z^{o}_{ig} \end{bmatrix} \qquad (3.9)$$

and

$$\{g\} = \begin{bmatrix} 0 \\ g_x \\ g_y \\ g_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -9.81\ \mathrm{m/sec}^2 \end{bmatrix} \qquad (3.10)$$

The potential energy of all grid points in the ith link can be expressed as

$$(PE)_i^G = - \sum_{g=1}^{NG(i)} m_{ig} \, Tr(\{r_{ig}^{o}\}\{g\}^T) \qquad (3.11)$$

The total potential energy is the sum of the potential energies due to the elasticity and the gravity which can be expressed as

$$(PE)_i = (PE)_i^E + (PE)_i^G \qquad (3.12)$$

Thus, the kinetic and potential energy expressions have been obtained in terms of the elastic displacements, $p_{i\beta}$, which are also known as perturbation coordinates [1].

3.2.4 The Equation of Motion of the ith Link

The equation of motion of the ith link can be obtained from Lagrange's eqution as

$$\frac{d}{dt}\left[\frac{\partial (KE)_i}{\partial p_{i\alpha}}\right] - \frac{\partial (KE)_i}{\partial p_{i\alpha}} + \frac{\partial (PE)_i}{\partial p_{i\alpha}} = f_{i\alpha} \qquad (3.13)$$

where

$$\alpha = 1, 2, \ldots, NP(i)$$

NP(i) = number of perturbation coordinates
= 6 x NG(i)

$p_{i\alpha}$ = elastic displacement of a node in the ith link

Substituting the values for the kinetic and potential energies in Eqn. (3.13) and carrying out the differentiation, the resulting equation can be written as [1],

$$\sum_{\beta=1}^{NP(i)} m_{i\alpha\beta}\, \ddot{p}_{i\beta} + \sum_{\beta=1}^{NP(i)} g_{i\alpha\beta}\, \dot{p}_{i\beta} + \sum_{\beta=1}^{NP(i)} k_{i\alpha\beta}\, p_{i\beta} = f_{i\alpha}$$

$$\alpha = 1, 2, \ldots, NP(i) \qquad (3.14)$$

where

$$m_{i\alpha\beta} = T_r\,[[T_o^{\,i}][J_{i\alpha\beta}][T_o^{\,i}]^T] \qquad (3.15)$$

In this equation

$$[J_{i\alpha\beta}] = \sum_{g=1}^{NG(i)} m_{ig}\, \{\phi_{ig\alpha}\}\{\phi_{ig\beta}\}^T \qquad (3.16)$$

where

$$\{\phi_{i11}\} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \{\Phi_x\} , \qquad (3.17)$$

$$\{\phi_{i12}\} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \{\Phi_y\} , \qquad (3.18)$$

$$\{\phi_{i13}\} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \{\Phi_z\} , \qquad (3.19)$$

and

$$\{\phi_{i14}\} = \{\phi_{i15}\} = \{\phi_{i16}\} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (3.20)$$

Eqn. (3.20) represent rotation along the x_m, y_m and z_m axes, the local axes.

The equations (3.15) to (3.20) can be written in the general form as

$$\{\phi_{ig\beta}\} = \{\Phi_x\} \text{ for } \beta = 1 + 6(g-1)$$

$$= \{\Phi_y\} \text{ for } \beta = 2 + 6(g-1)$$

$$= \{\Phi_z\} \text{ for } \beta = 3 + 6(g-1) \qquad (3.21)$$

$$= \{0\} \text{ for other values of } \beta$$

Simply, replacing β by α we get the terms for $\{\phi_{ig\alpha}\}$. Here, α and β are dummy variables.

In Eqn. (3.16), m_{ig} is the mass of the gth grid point of the ith link.

The damping term $g_{i\alpha\beta}$, in Eqn. (3.14) is defined as

$$g_{i\alpha\beta} = T_r \left[\sum_{j=1}^{NL} 2[T_o^{\,i}][J_{i\alpha\beta}][U_{ij}^T]\, \dot{\theta}_j \right] \qquad (3.22)$$

where

$$[U_{ij}] = \frac{\partial [T_o^{\,i}]}{\partial \theta_j} = [T_o^{\,j-1}][Q][T_{j-1}^{i}] \quad j<i$$

$$= 0 \quad j>i \qquad (3.23)$$

$\dot{\theta}_j$ = angular velocity of the jth link, and

NL = total number of links in the robotic manipulator

The stiffness term, $k_{i\alpha\beta}$ in Eqn. (3.14), is given as

$$k_{i\alpha\beta} = (k_{i\alpha\beta})_1 + (k_{i\alpha\beta})_2 + (k_{i\alpha\beta})_3 \qquad (3.24)$$

where

$(k_{i\alpha\beta})_1$ is the three dimensional finite element stiffness matrix term,

$$(k_{i\alpha\beta})_2 = T_r\left[\sum_{j=1}^{NL}\sum_{k=1}^{NL}[T_o^{\,i}][J_{i\alpha\beta}][U_{ijk}]^T \dot{\theta}_j \dot{\theta}_k\right] \qquad (3.25)$$

where

$$[U_{ijk}] = \frac{\partial[U_{ij}]}{\partial\theta_k} = \frac{\partial[T_o^{\,i}]}{\partial\theta_j\partial\theta_k}$$

$$= [T_o^{\,j-1}]\left[\frac{\partial T_{j-1}^{j}}{\partial\theta_j}\right][T_j^{\,k-1}]\left[\frac{\partial T_{k-1}^{k}}{\partial\theta_k}\right][T_k^{i}] \quad j<k<i$$

$$= [T_o^{\,k-1}]\left[\frac{\partial T_{k-1}^{k}}{\partial\theta_k}\right][T_k^{\,j-1}]\left[\frac{\partial T_{j-1}^{j}}{\partial\theta_j}\right][T_j^{\,i}] \quad k<i<j$$

$$= 0 \qquad j > i \text{ or } k > i \qquad (3.26a)$$

Eqn. (3.26) can be further simplified by using the Q matrix differentiation concept introduced in Chapter 2, so we can write

$$[U_{ijk}] = [T_o^{j-1}][Q][T_{j-1}^{k-1}][Q][T_{k-1}^{i}] \quad j<k<i$$

$$= [T_o^{k-1}][Q][T_{k-1}^{j-1}][Q][T_{j-1}^{i}] \quad k<j<i \qquad (3.26b)$$

$$= 0 \qquad j > i \text{ or } k > i$$

The third term in the Eqn. (3.24) is given as

$$(K_{i\alpha\beta})_3 = \text{Tr} \sum_{j=1}^{NL} [T_o^{i}][J_{i\alpha\beta}][U_{ij}]^T \ddot{\theta}_j] \qquad (3.27)$$

where $\ddot{\theta}_j$ = angular acceleration of the jth link.

The force term $f_{i\alpha}$ is given as

$$f_{i\alpha} = (f_{i\alpha})_1 - (f_{i\alpha})_2 - (f_{i\alpha})_3 + (f_{i\alpha})_4 \qquad (3.28)$$

where,

$(f_{i\alpha})_1$ is due to the externally applied forces and torques including the inter-link joint constraint forces,

$$(f_{i\alpha})_2 = T_r\left[\sum_{j=1}^{NL}\sum_{k=1}^{NL} [T_o^i][J_{i\alpha}][U_{ijk}]^T \dot{\theta}_j \dot{\theta}_k\right], \qquad (3.29)$$

$$(f_{i\alpha})_3 = T_r\left[\sum_{j=1}^{NL} [T_o^i][J_{i\alpha}][U_{ij}]^T \ddot{\theta}_j\right], \qquad (3.30)$$

and

$$(f_{i\alpha})_4 = T_r([T_o^i][M_{i\alpha}]) \qquad (3.31)$$

To evaluate $(f_{i\alpha})_2$, $(f_{i\alpha})_3$ and $(f_{i\alpha})_4$ one has to evaluate the matrices $[J_{i\alpha}]$ and $[M_{i\alpha}]$. These can be evaluated using the following relationships

$$[J_{i\alpha}] = \sum_{g=1}^{NG(i)} m_{ig}\{\phi_{ig\alpha}\}\{b_{ig}\}^T \qquad (3.32)$$

$$[M_{i\alpha}] = \sum_{g=1}^{NG(i)} m_{ig}\{\phi_{ig\alpha}\}\{g\}^T \qquad (3.33)$$

where,

$$\{b_{ig}\} = \begin{bmatrix} 1 \\ (x_m)_{ig} \\ (y_m)_{ig} \\ (z_m)_{ig} \end{bmatrix} \qquad (3.34)$$

is a constant vector representing the undeformed position of the gth grid point in the ith link. Thus all the terms in the Eqn. (3.14) are defined. This equation can be simplified based on certain approximations which are explained next.

3.2.5 The Simplification of the Scalar Equations

The direct implementation of the scalar Eqn. (3.14) can be extremely difficult due to the large amount of calculations involved. Hence, a very important mathematical tool called the matrix dot product is used to simplify this equation.

To begin this simplification, the coefficients of the scalar equation, Eqn. (3.14), are written using the matrix dot product notation, which is

$$m_{i\alpha\beta} = T_r[[T_o^i][J_{i\alpha\beta}][T_o^i]^T] = T_r[[T_o^i]^T[T_o^i][J_{i\alpha\beta}]]$$

In this equation, each of the three matrices $[T_o^i]$, $[J_{i\alpha\beta}]$ and $[T_o^i]^T$ have been rewritten by cyclically permutating them which is a valid operation when the trace of the product of these matrices is required. This equation can be further simplified by writing

$$Tr[[C_i]^T[J_{i\alpha\beta}]] = [\underline{C}_i] \odot [J_{i\alpha\beta}] \qquad (3.35)$$

where

$$[C_i] = [T_o^i]^T[T_o^i] = [C_i]^T \qquad (3.36)$$

and $\odot$ is the notation for matrix dot product. In general, it can be defined as

$$[A] \odot [B] = \text{Tr}\,([A]^T[B]) \tag{3.37}$$

Similarly, the damping term can be simplified by writing

$$\begin{aligned} g_{i\alpha\beta} &= \text{Tr}\Big[\sum_{j=1}^{NL} 2[T_o{}^i][J_{i\alpha\beta}][U_{ij}]^T \dot{\theta}_j\Big] \\ &= \text{Tr}\Big[\Big(\sum_{j=1}^{NL} 2[U_{ij}]^T[T_o{}^i]\,\dot{\theta}_j\Big)\,[J_{i\alpha\beta}]\Big] \\ &= \text{Tr}\,[[D_i]^T[J_{i\alpha\beta}]] = [D_i] \odot [J_{i\alpha\beta}] \end{aligned} \tag{3.38}$$

where

$$[D_i] = \sum_{j=1}^{NL} 2[C_{ij}]\,\dot{\theta}_j \tag{3.39}$$

$$[C_{ij}] = [T_o{}^i]^T \odot [U_{ij}] \tag{3.40}$$

Using the same procedure as before, one can write the second stiffness term as

$$(k_{i\alpha\beta})_2 = \text{Tr}\Big[\sum_{j=1}^{NL}\sum_{k=1}^{NL} [T_o{}^i][J_{i\alpha\beta}][U_{ijk}]^T \dot{\theta}_j\dot{\theta}_k\Big]$$

$$= \mathrm{Tr}\left[\left(\sum_{j=1}^{NL}\sum_{k=1}^{NL} [U_{ijk}]^T [T_o{}^i]\ \dot{\theta}_j\ \dot{\theta}_k\right)[J_{i\alpha\beta}]\right] \quad (3.41)$$

$$= \mathrm{Tr}[[E_i]^T[J_{i\alpha\beta}]] = [E_i] \odot [J_{i\alpha\beta}]$$

where,

$$[E_i] = \sum_{j=1}^{NL}\sum_{k=1}^{NL} [C_{ijk}]\ \dot{\theta}_j\ \dot{\theta}_k \quad (3.42)$$

$$[C_{ijk}] = [T_o{}^i]^T \odot [U_{ijk}] \quad (3.43)$$

The acceleration term in the stiffness matrix can be expressed as

$$(k_{i\alpha\beta})_3 = \mathrm{Tr}\left[\sum_{j=1}^{NL} [T_o{}^i][J_{i\alpha\beta}][U_{ij}]^T\ \ddot{\theta}_j\right]$$

$$= \mathrm{Tr}\left[\left(\sum_{j=1}^{NL} [U_{ij}]^T[T_o{}^i]\ \ddot{\theta}_j\right)[J_{i\alpha\beta}]\right]$$

$$= \mathrm{Tr}[[F_i]^T[J_{i\alpha\beta}]] = [F_i] \odot [J_{i\alpha\beta}]$$

where

$$[F_i] = \sum_{j=1}^{NL} [C_{ij}]\ddot{\theta}_j \qquad (3.45)$$

Similarly, the velocity and acceleration dependent force terms are given as

$$(f_{i\alpha})_2 = T_r\left[\sum_{j=1}^{NL}\sum_{k=1}^{NL} [T_o{}^i][J_{i\alpha}][U_{ijk}]^T \dot{\theta}_j\dot{\theta}_k\right] = [E_i]\odot[J_{i\alpha}] \qquad (3.46)$$

$$(f_{i\alpha})_3 = Tr\left[\sum_{j=1}^{NL} [T_o{}^i][J_{i\alpha}][U_{ij}]^T \ddot{\theta}_j\right] = [F_i]\odot[J_{i\alpha}] \qquad (3.47)$$

Thus, it is obvious that the force terms can be obtained using the already defined matrices from Eqns. (3.42) and (3.45).

The advantages of expressing the coefficients of the system equation in terms of the dot product notation are: i) the calculations are considerably reduced and ii) some important propositions can be made.

The first advantage can be distinctly seen by noting that the calculation of the various terms under the summation sign is transformed to finding the dot products involving the four matrices $[C_i]$, $[D_i]$, $[E_i]$ and $[F_i]$ which are same for all the nodes of a given link. The second advantage can be realized by proving an important proposition that the terms in the mass matrix are time-invariant and represent the lumped element matrix of the finite element analysis technique. The proof of this assumption can be seen in [1].

The matrix form of the simplified scalar equation can be written as

$$[m_i]\{\ddot{p}_i\} + [g_i]\{\dot{p}_i\} + [k]\{p_i\} = \{f_i\} \qquad (3.48)$$

At this point one important characteristic of this equation must be noted. The Eqn. (3.48) is the dynamic equation of the ith link in the three-dimensional space. The force vector $\{f_i\}$ contains the unknown interlink joint constraint forces. These interlink joint constraint forces can be eliminated by assembling all the link dynamic equations. But, before assembling them, each link equation must be written in terms of the global coordinates. For this purpose, the mass, damping and stiffness matrices must be pre

and post multiplied by the matrix of direction cosines, so that the nodal quantities like the displacements, velocities, etc. are oriented along the global coordinate system. Some of the important techniques to obtain this transformation matrix [37] will be discussed in the next section.

3.2.6 The Methods of Obtaining the Matrix of Direction Cosines of Various Links

Since the robotic manipulator is a space-frame structure, one has to determine all the three direction cosines of a given link to carry out the finite element analysis. The method commonly used [38] involves the use of the Euler angles and the rotation about three separate axes. There are two ways of carrying out this process which are called Y-Z-X Transformaton and Z-Y-X Transformation. In these transformations the coordinates of a point on a given link are transformed from the local coordinate system into the global coordinate system. Representing the local set of coordinates by (x_m, y_m, z_m) and the global by (X_o, Y_o, Z_o), the relationship between these two sets can be written as

$$\{X_o, Y_o, Z_o, 1\}^T = [T]\{x_m, y_m, z_m, 1\}^T \qquad (3.49)$$

where the matrix [T] can be written as

$$[T] = \begin{bmatrix} l_1 & l_2 & l_3 & X_B \\ m_1 & m_2 & m_3 & Y_B \\ n_1 & n_2 & n_3 & Z_B \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (3.50)$$

In the Eqn. (3.50), l_1, m_1, n_1 represent the direction cosines of the local x_m axis which rotates with the link. Similarly l_2, m_2, n_2 and l_3, m_3, n_3 repesent the direction cosines of the y_m and z_m axes respectively. The last column element X_B, Y_B and Z_B are the components of the positon vector of the origin of the link coordinate system. The local and the global coordinate systems are shown in the Fig. 3.1. From the displacement analysis carried out in Chapter 2, the coordinates of the end points, A and B, of the link in the global system are obtained. The direction cosines of the x_m axis shown in the Fig. 3.1, can be obtained by using the equations

$$l_1 = \frac{X_B - X_A}{L_i} \qquad (3.51)$$

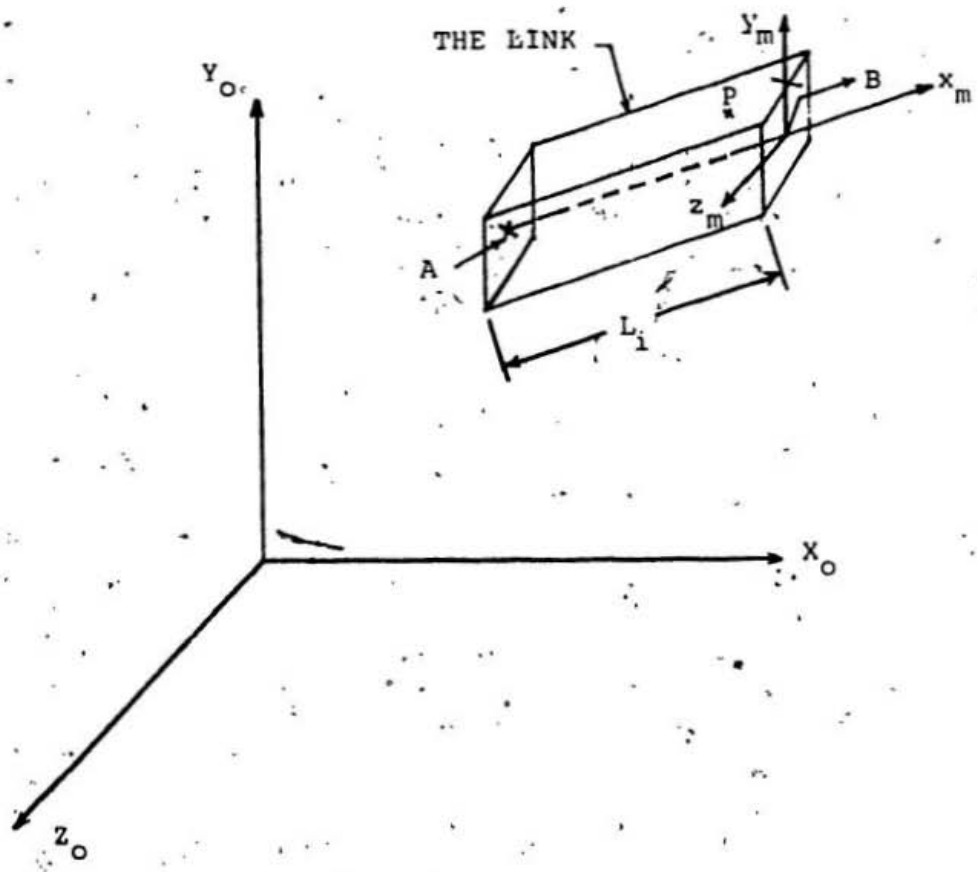

Fig. 3.1: Local and Global Coordinate Systems

$$m_1 = \frac{Y_B - Y_A}{L_i} \tag{3.52}$$

$$n_1 = \frac{Z_B - Z_A}{L_i} \tag{3.53}$$

with

$$L_i = \sqrt{(X_B - X_A)^2 + (Y_B - Y_A)^2 + (Z_B - Z_A)^2} \tag{3.54}$$

To define the stiffness, inertia properties etc., the major and minor axes of the link should be known because it can be non-symmetrical in the general case. Therefore, one needs to have the coordinates of another point P (shown in Fig. 3.1) in the global coordinate system; the restriction on the point P is that it should be in the $x_m - y_m$ plane.

In the Y-Z-X Transformation, by a succession of three Euler angle rotations, the global axes are made to coincide with the local axes in the sense of the directions and a translation is needed to shift from the origin of the global coordinate system to the local coordinate system. This method is given in detail in the Appendix B and it is a fairly lengthy one. In this method, the matrix [T] is

obtained as a product of three transformation matrices which can be mathematically written as

$$[T] = [C_\alpha][C_\beta][C_\gamma] \tag{3.55}$$

where α, β and γ have been used to identify these individual matrices. The details of these matrices $[C_\alpha]$, $[C_\beta]$ and $[C_\gamma]$ are given in [38].

The problem of finding the direction cosines of the y_m and z_m axes can be solved by four other methods which are simpler than the one mentioned above. These four methods are discussed now.

In the first method, instead of using three rotations and a translation, one can make the global axes coincide with the local axes in one rotation, and one translation. The translation vector is the same as before. In Fig. 3.2a, x'-y'-z' system is obtained by translating X_o-Y_o-Z_o system and then this x'-y'-z' system is made to coincide with the x_m-y_m-z_m system. This is possible by defining an axis through the origin of the x'-y'-z' system. The process of this transformation is shown in the Figs. 3.2a and 3.2b. Imagine a point A' on the negative side of the x' axis which can be rotated about a vector $\underline{S}$ in space so that after θ degrees of rotation, it coincides with the

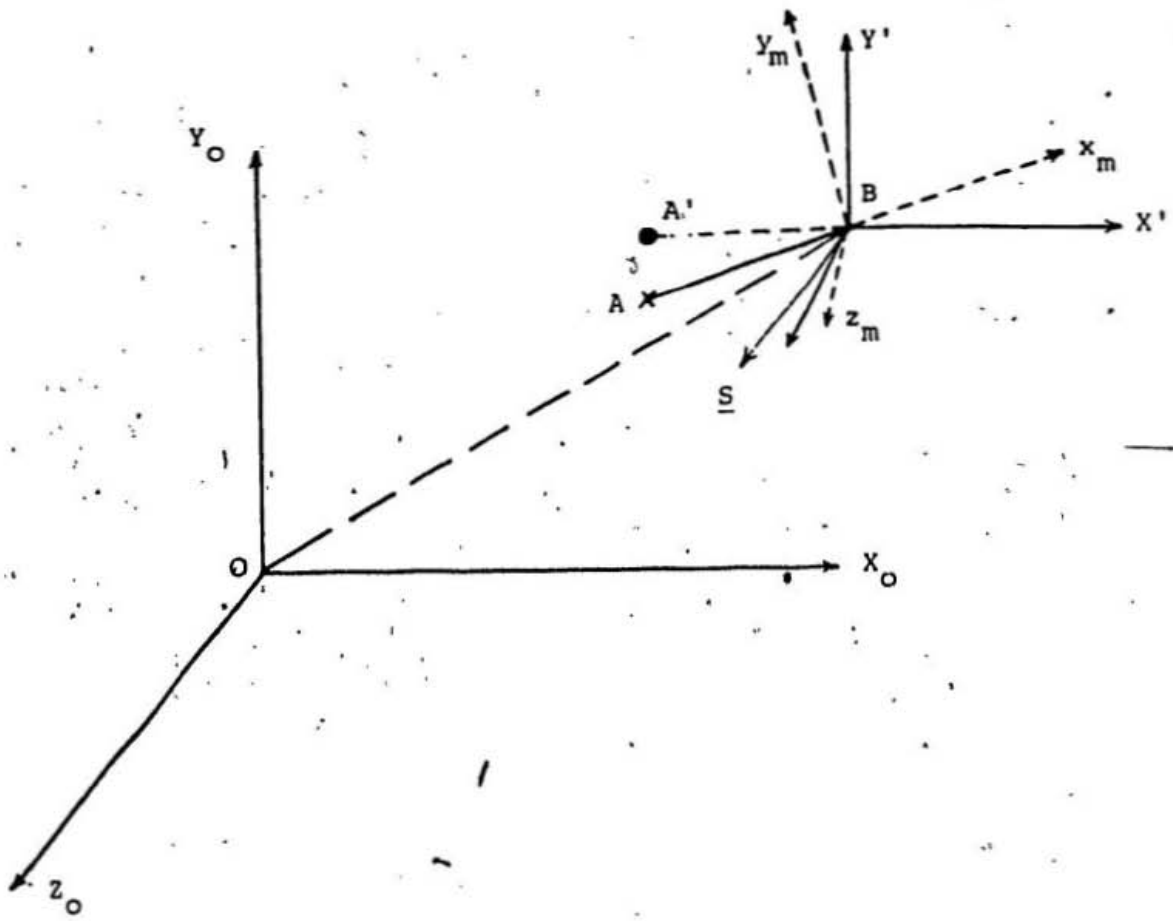

Fig. 3.2a: Transformations Relating the Local and the Global Coordinate Systems

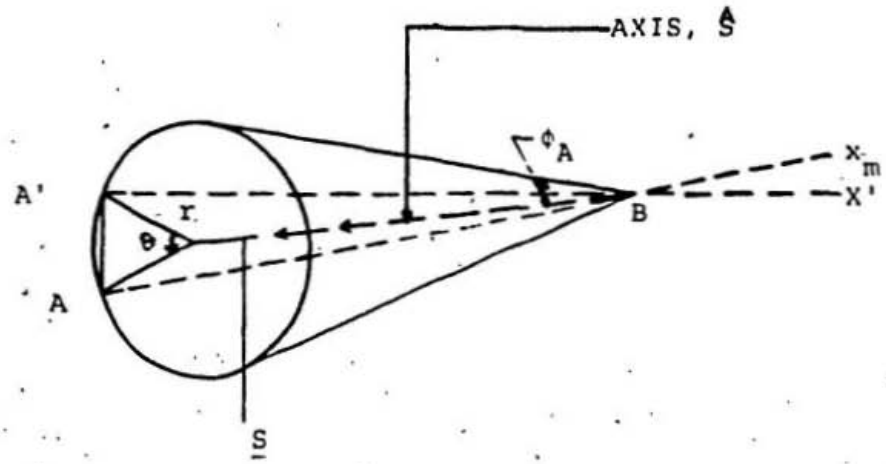

Fig. 3b: Rotation About the Vector S

point A on the x_m axis. This rotation is shown in the Fig. 3.2b. In this transformation the line A'B coincides with the line AB after θ degrees of rotation and this transformation is along the surface of a cone. If k_x, k_y and k_z represent the direction cosines of the vector $\underline{\hat{S}}$ then this unit vector S can be written as

$$\hat{S} = k_x\hat{i} + k_y\hat{j} + k_z\hat{k} \qquad (3.56)$$

where $\hat{i}$, $\hat{j}$ and $\hat{k}$ are the unit vectors along the axes x', y' and z' respectively. The coordinates of the point A in the x'-y'-z' coordinate system can be written as X_A-X_B Y_A-Y_B and Z_A-Z_B. Similarly, for the point P it will be X_P-X_B Y_P-Y_B and Z_P-Z_B. The coordinates of these points, A and P, can always be calculated in $x_m-y_m-z_m$ system. Since the angle ϕ_A (the semi-apex angle of the cone) between $\underline{\hat{S}}$ and $\underline{\hat{BA}}'$ is the same as $\underline{\hat{S}}$ and $\underline{\hat{BA}}$, one can write

$$\cos \phi_A = \underline{\hat{S}} \cdot \underline{\hat{BA}}' = \underline{\hat{S}} \cdot \underline{\hat{BA}} \qquad (3.57)$$

Here, the symbol ^ represents a unit vector.

It should be noted that the coordinates of the points A' and P' in the x'-y'-z' system will be the same as the coordinates of the point A and P in the $x_m-y_m-z_m$ system,

and these coordinates in the x_m-y_m-z_m system can be calculated since their coordinates in the X_o-Y_o-Z_o system are known now. Therefore, the unit vectors $\hat{BA}$, and $\hat{BA}'$ are known. Thus, Eqn. (3.57) using the last equality can be written as

$$k_x(l_A' - l_A) + k_y(m_A' - m_A) = -k_z(n_A' - n_A) \qquad (3.58)$$

When this process is carried out for the point P then one can write

$$k_x(l_p' - l_p) + k_y(m_p' - m_p) = -k_z(n_p' - n_p) \qquad (3.59)$$

Solving for k_x and k_y in terms of k_z using Eqns. (3.58) and (3.59), one can express the unit vector S as

$$\hat{S} = \frac{k_x\hat{i} + k_y\hat{j} + k_z\hat{k}}{\sqrt{k_x^2 + k_y^2 + k_z^2}} \qquad (3.60)$$

Thus, the direction cosines of $\hat{S}$ are known. The angle ϕ_A is calculated from the Fig. (3.2b), where one can write

$$\sin \frac{\theta}{2} = \frac{\frac{1}{2}(A'A)}{r} = \frac{\frac{1}{2}(A'A)}{BA' \sin \phi_A} , \qquad (3.61)$$

where the angle ϕ_A is obtained using Eqn. (3.57) and A'A can be obtained because the coordinates of the points A' and A are known. The direction cosines of x_m-y_m-z_m system are obtained from the matrix equality given by [9].

$$\begin{bmatrix} l_1 & l_2 & l_3 & 0 \\ m_1 & m_2 & m_3 & 0 \\ n_1 & n_2 & n_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \qquad (3.62)$$

$$\begin{bmatrix} k_x k_x \text{vers}\theta + \cos\theta & k_y k_x \text{vers}\theta - k_z \sin\theta & k_z k_x \text{vers}\theta + k_y \sin\theta & 0 \\ k_x k_y \text{vers}\theta + k_z \sin\theta & k_y k_y \text{vers}\theta + \cos\theta & k_z k_y \text{vers}\theta - k_x \sin\theta & 0 \\ k_x k_z \text{vers}\theta - k_y \sin\theta & k_y k_z \text{vers}\theta + k_x \sin\theta & k_z k_z \text{vers}\theta + \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where vers $\theta = 1 - \cos \theta$.

This method is much shorter than the method that is commonly used now because all that is involved is the simultaneous solution of Eqns. (3.58) and (3.59) where one

obtains k_x, k_y in terms of k_z and these are substittued in the Eqn. (3.60). Thus, this vector about which the rotation is made, is defined. The next step is the calculation of $\frac{\theta}{2}$ using Eqn. (3.61). After obtaining θ, one can calculate the direction cosines from the Eqn. (3.62).

The second method can be understood from the Fig. (3.3). Since the coordinates of the points A, B and P are known, the direction cosines of the vectors AB and AP can be calculated using Eqns. (3.51) to (3.54). Since the axis z_m is perpendicular to the plane containing AB and AP, one can use these two conditions by writing the two equations for the unit vectors as mentioned below:

$$\hat{z}_m \cdot \hat{AB} = l_3\, l_1 + m_3\, m_1 + n_3\, n_1 = 0 \qquad (3.63)$$

$$\hat{z}_m \cdot \hat{AP} = l_3\, l_p + l_3\, m_p + h_3\, n_p = 0 \qquad (3.64)$$

where l_p, m_p and n_p are the direction cosines of the vector $\hat{AP}$. Solving l_3 and m_3 in terms of n_3 and using the relationship

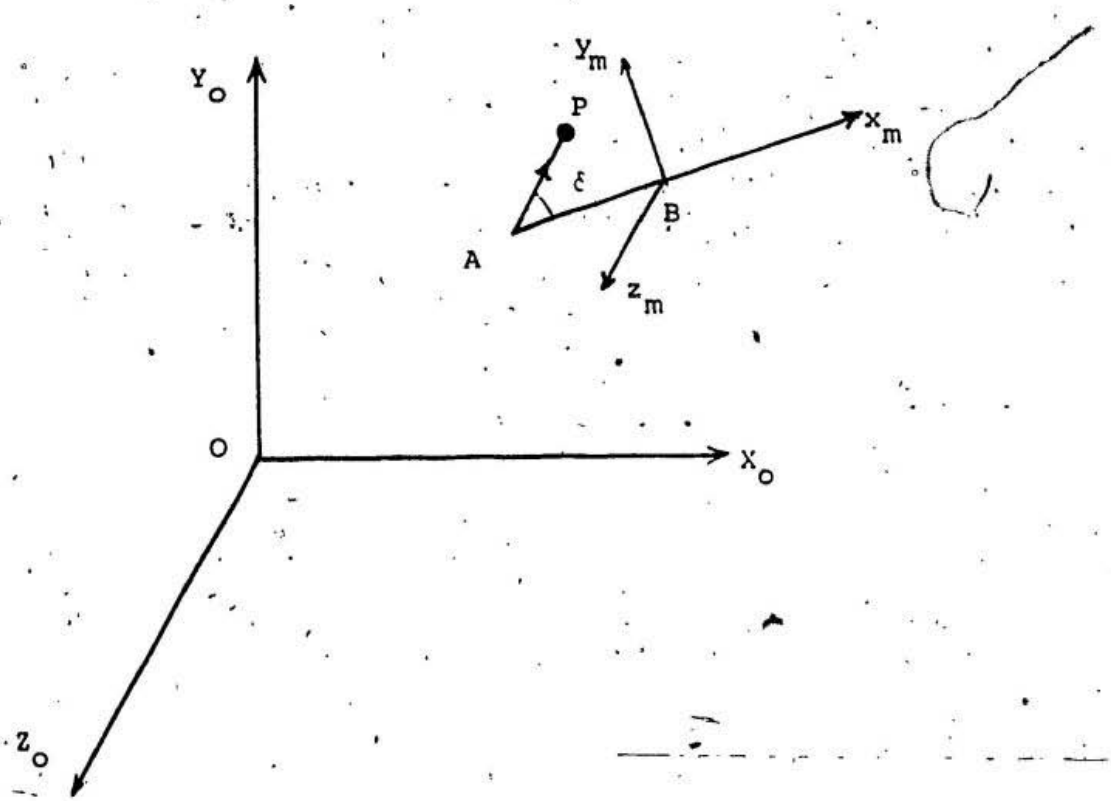

Fig. 3.3: Coordinate Systems Used for Method 2

$$\hat{z}_m = \frac{l_3\, i + m_3\, j + n_3\, k}{\sqrt{l_3^2 + m_3^2 + n_3^2}} \qquad (3.65)$$

one obtains the unit vector z_m. Then, the unit vector y_m is obtained using

$$\hat{y}_m = \hat{z}_m \times \hat{x}_m \qquad (3.66)$$

The direction cosines of $\hat{AB}$ and $\hat{x}_m$ are same.

Another variation of this method that can be used is that the angle δ shown in the Fig. (3.3) can be calculated since the vectors $\underline{AP}$ and $\underline{AB}$ are defined. Therefore, the angle between y_m and $\underline{AP}$ is also defined. Thus, one can use the dot products of the unit vectors in the following way:

$$\cos (90 - \delta) = \hat{y}_m \cdot \hat{AP} = l_2\, l_p + m_2\, m_p + n_2\, n_p \qquad (3.67)$$

$$\cos 90 = 0 = \hat{y}_m \cdot \hat{x}_m = l_2\, l_1 + m_2\, m_1 + n_2\, n_1 \qquad (3.68)$$

Solving for l_2, and m_2 in terms of n_2 and substituting in the relation

$$y_m \mp \frac{l_2\, i + m_2 j + n_2 k}{\sqrt{l_2^2 + m_2^2 + n_2^2}} \qquad (3.69)$$

one obtains the direction cosines of y_m. Then, the direction cosines of the z_m axis are obtained using the relationship

$$\hat{z}_m = \hat{x}_m \times \hat{y}_m \qquad (3.70)$$

The fourth method can be understood from the Fig. (3.4). Here, the points $P_1(X_1, Y_1, Z_1)$, $P_2(X_2, Y_2, Z_2)$ and $P_3(X_3, Y_3, Z_3)$ are co-planar points, and $R(X, Y, Z)$ is any arbitrary point in this plane. Then, the equation of this plane can be obtained using the relation

$$\underline{P_1R} \cdot \underline{P_1P_2} \times \underline{P_1P_3} = 0 \qquad (3.71)$$

In terms of the determinant, this equation can be rewritten as

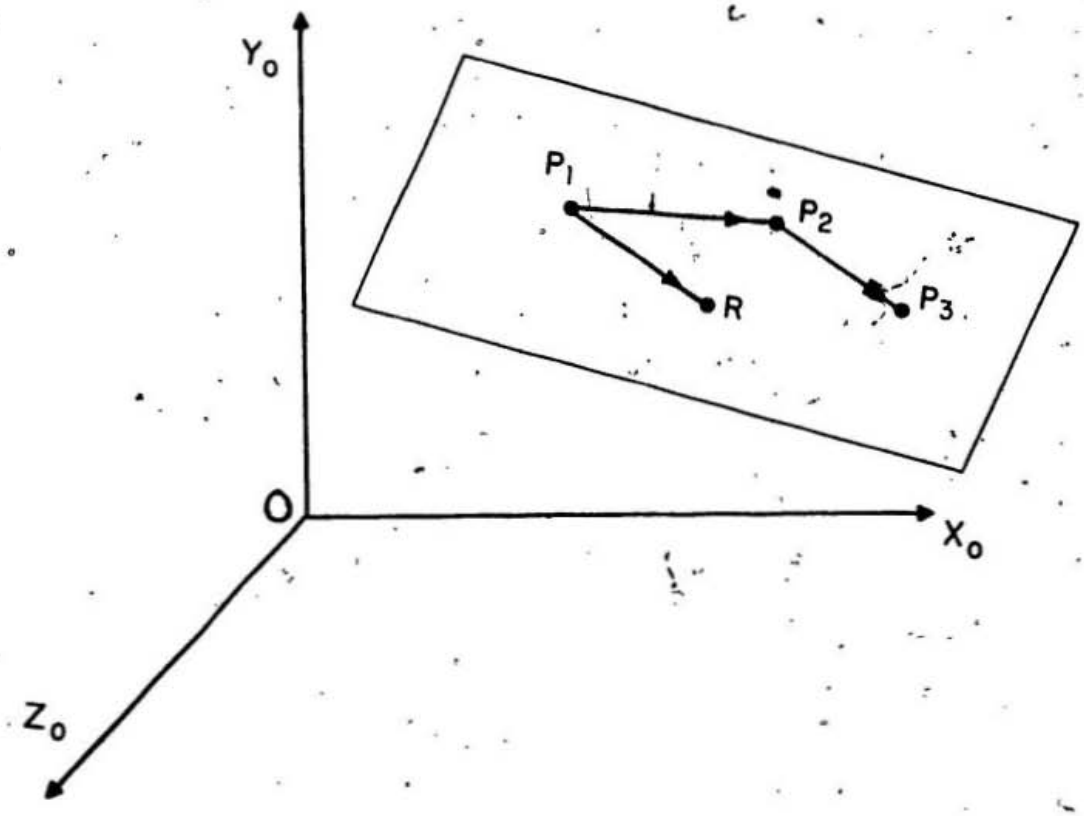

Fig. 3.4: The Co-planar Vectors

$$\begin{bmatrix} (X - X_1) & (Y - Y_1) & (Z - Z_1) \\ (X_2 - X_1) & (Y_2 - Y_1) & (Z_2 - Z_1) \\ (X_3 - X_1) & (Y_3 - Y_1) & (Z_3 - Z_1) \end{bmatrix} = 0 \qquad (3.72)$$

This determinant can be expanded and the equation of the plane can be written in the form

$$a_1 X + a_2 Y + a_3 Z = c \qquad (3.73)$$

where a_1, a_2, a_3 and c are functions of the coordinates of the points P_1, P_2 and P_3 respectively. Introducting a vector $\underline{a}$ such that

$$\underline{a} = a_1 \hat{i} + a_2 \hat{j} + a_3 \hat{k} \qquad (3.74)$$

then the direction cosines of the normal to the plane are given by

$$n_x = \frac{a_1}{|\underline{a}|} \quad , \quad n_y = \frac{a_2}{|\underline{a}|} \quad , \quad \text{and } n_z = \frac{a_3}{|\underline{a}|} \quad . \qquad (3.75)$$

For our problem, the points A, B and P in the method 3 can be

represented by the points P_1, P_2 and P_3 respectively. This method could possibly be the easiest to use. The direction cosines n_x, n_y and n_z would correspond to that of the $\hat{z}_m$ axis and the direction cosines of the $\hat{y}_m$ axis can be obtained by taking the cross product of the $\hat{z}_m$ and $\hat{x}_m$ vectors.

In using all the five methods, including the conventional method given in [38], the direction cosines of the x_m axis are calculated using the points A and B. Whether the direction cosines of the z_m or y_m axis are obtained next depending upon the method used, there are always two solutions. Therefore one has to choose the correct solution and this can be accomplished by using the relationship

$$\begin{bmatrix} X_P \\ Y_P \\ Z_P \\ 1 \end{bmatrix} = \begin{bmatrix} l_1 & l_2^I & l_3^I & X_B \\ m_1 & m_2^I & m_3^I & Y_B \\ n_1 & n_2^I & n_3^I & Z_B \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} \tag{3.76}$$

where, the superscript I represents the first set of solutions. The position vector of the point P in the local coordinate system is on the right hand side and in the global

coordinates, it is on the left-hand side. These two vectors are related by the transformation matrix. Here, the position vector of the point P with respect to the global and the local axes are known, and the direction cosines l_1, m_1 and n_1 are also known. One has to substitute the first set of values for the direction cosines of y_m and z_m axes to check if the Eqn. (3.76) is satisfied. If not, then one has to use the second set of the solutions; the direction cosines of the second set would be 180 degrees apart from the first set.

Using these direction cosines of the link coordinate system one can write the transformation equation relating the local and global coordinator for the link as

$$\{p_i\} = [T1]\{x\} \tag{3.77}$$

In this equation, the elements of the vector $\{x\}$ are parallel to the global axes. Now, each of these links can be assembled and the global equation of motion for the entire structure can be written as

$$[M]\{\ddot{w}(t)\} + [G]\{\dot{w}(t)\} + [K]\{w(t)\} = \{F(t)\} \tag{3.78}$$

In Eqn. (3.78), the vector $\{w(t)\}$ contains the nodal displacements for the entire structure. The matrices [G] and [K] are functions of the angular velocities and accelerations at all the nodes as given in Eqns. (3.22) and (3.24). It is possible to neglect some of these component matrices which is discussed in the next section. It should be added here that one can arrive at the global system of equations, Eqn. (3.78), also by writing the equations for each of the elements of the links which is normally done in the finite element methods used for the structural analysis.

3.2.7 The Simplification of the Global Equations Using the Mathematical Approximations

The direct implementation of Eqn. (3.78) is involved and will require large computer memory and CPU time. The equation can be simplified by examining the absolute values of the dynamic matrices, which are given as

$$|[M]| \cong m \tag{3.79}$$

$$|[G]| \cong m\,\dot{\theta} + |[C]| \tag{3.80}$$

$$|[K]| \cong k + m\,\dot{\theta}^2 + m\,\ddot{\theta} \tag{3.81}$$

where

m is the largest lumped mass of the grid point in the robot link;

k is the generalized stiffness term;

$\ddot{\theta}, \dddot{\theta}$ are the maximum nominal velocities and accelerations obtained in Section 2.5 and 2.6;

$|[C]|$ = absolute value of the viscous damping matrix

$$= 2\zeta\sqrt{km} \qquad (3.82)$$

where,

ζ = percent viscous damping factor.

By substituting numerical values to these terms that are appropriate to a robotic manipulator, we have

$m = 175.13$ Kg (1.0 lb. sec^2/inch)

$K = 1.7513$ E7 N/m (10^5 lb/inch)

$\ddot{\theta} = 1$ rad/sec

$\dddot{\theta} = 4$ rad/sec^2

$\zeta = 0.05$

The above mentioned values are typical for the upperarm of a T3R3 model robotic manipulator with v = 1.27 m/sec (50 inch/sec), the tool tip velocity. By substituting these values into the above equation, Eqns. (3.79) to (3.81), we get

$$|M| = 175.13$$

$$|G| = (175.13)(1)^2 + 2(0.05) \checkmark 1.7513E7 * 175.13$$

$$= 175.13 + 5538.1$$

$$|K| = 1.7513E7 + (175.13)(1)^2 + (175.13)(4)$$

Thus, it can be seen that, the quantities involving $\dot{\theta}$ and $\ddot{\theta}$ are small as compared to the other terms in these equations. By neglecting these terms, one may incur an error up to 3%. But this will result in a very large saving of computer time and memory. Hence, one can neglect these terms and calculate the remaining terms. These numerical values used were for the gripper which gives rise to the most dominant term in the system matrices.

Therefore, the simplified equation can be written as

$$[M]\{\ddot{w}(t)\} + [C]\{\dot{w}(t)\} + [K]\{w(t)\} = \{F(t)\} \qquad (3.83)$$

where

[M] is the system mass matrix

[C] is the system viscous damping matrix

and

[K] is the structural stiffness matrix

Here, the matrices [M], [C] and [K] are the same as those used in the lumped mass finite element analysis. It must be noted that, eventhough the terms involving $\dot{\theta}$ and $\ddot{\theta}$ were omitted in the damping and stiffness coefficients, they cannot be neglected in the force vector. An element of the force vector can be written as

$$\begin{aligned} F(t) &= F_1 - F_2 - F_3 + F_4 \\ &\approx F_1 - m x_{ig} \dot{\theta}^2 - m\ddot{\theta} + mg \qquad (3.84) \\ &\approx F_1 - (175.13)(1.516)(1)^2 - (175.13)(4) \\ &\quad + (175.13)(-9.81) \end{aligned}$$

By comparing the other three terms it is obvious that the terms involving $\dot{\theta}$ and $\ddot{\theta}$ are also dominant and so these terms cannot be neglected.

In order to clarify the concepts and notations used in this chapter, the force term is expanded for one of the nodes on the forearm in Appendix C.

The static or dynamic behaviour of the robotic structures can be analyzed using Eqn. (3.83). The natrual frequencies are important dynamic characteristics of a given system and these can be calculated using this equation. In the next section, these frequencies are calculated and the variations of these frequencies as a function of several design variables is also studied.

3.3 The Study of the Variation of the Natural Frequencies of the Robotic Manipulators Due to Several Design Parameters

3.3.1 The Use of the Natural Frequency Calculations in the Transient Steady State Analysis

The calculations of the natural frequencies are quite useful in the design of the mechanical systems because they represent its overall dynamic characteristics. For example, in the design of the blades of an auto cooling fan [39], these were modelled as rotating beams with tip masses and their natural frequencies were calculated for a given set of values of the design parameters such as the setting angle and the tip mass. Then, these variables were altered and the natural frequencies were correspondingly obtained. From these results, the acceptable values of the parameters were

selected. As another example, in the work carried out in [40], the design of a milling machine was based on the effect of the modification of its structural components on the natural frequencies as well as the mode shapes. Similar studies on a lathe spindle-workpiece system were done in [41,42].

Since the natural frequencies depend upon the inertia and the stiffness properties of each of the components of the system, it is quite desirable that in the design process one should vary the parameters of at least the most important structural components and study their effects on these frequencies. The natural frequencies calculations can be used in another way in the case of the robotic structures; these robotic manipulators are moving structures whose stiffness and inertia matrices vary depending upon the location of the end effector. Therefore, one has to calculate the elemental and global matrices corresponding to each location of the end effector which is moved along a specified trajectory while the robot is performing a job. By suitable mathematical approximations it was shown earlier that one can obtain the inertia matrices as invariant with respect to the position of the end effector, i.e. they need not be calculated again and again. With this approximation, if it is found that for a particular trajectory the natural frequencies also do not change at several widely spaced

points, then one need not calculate the stiffness matrices also repeatedly between these points. This way, these natural frequency calculations would be quite useful later on, in the study of the steady state or transient analysis of the robotic structures. On the other hand, if it is found that these natural frequencies change very slowly as the end effector moves along its trajectory then one should compute these matrices at only a certain number of points. Even then, there would be considerable saving in the computer CPU time.

To use the system dynamic equation, Eqn. (3.83), one needs to calculate the matrix of direction cosines for each link and use this matrix for each of the elements on this link. The computations of these matrices for the forearm of the robotic manipulator and a spatial link is explained in detail next.

3.3.2. The Determination of the Matrix of Direction Cosines

As stated in Section 3.2.6, the direction cosines of the x_m axis can be calculated because the end points of a given link are known from the inverse kinematic analysis. In addition, the point P is used to define the plane containing the x_m and y_m axes. These direction cosines, for two different systems shown in Figs. 1.1b and 3.5, were calculated. The Table 3.1 shows the dimensions of the

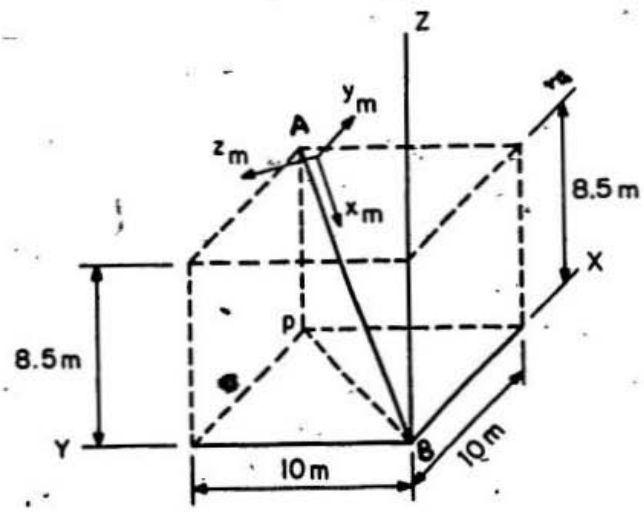

Fig. 3.5: Global and Local Coordinates of the Spatial Link AB

various links of the robotic manipulator. The results in the Table 3.2 clearly show that any of the five methods can be used to calculate the direction cosine matrix as the results obtained by all these methods are identical. However, the methods 3, 4 and 5, considering the already existing method as method 1, are simpler to use and one does not have to use translational transformation in these methods because the coordinates with respect to the X_o-Y_o-Z_o system are sufficient for the calculations.

3.3.3 The Dependence of the Undamped Natural Frequencies on the Design Parameters

The first parameter to be varied was the mass of the gripper, and the results of this variation are shown in the Table 3.3. The results show that the system is quite sensitive to any variations of the gripper mass. This is because the gripper is located at the end of the forearm, i.e., it is quite far from the joint connecting the upper arm and the forearm. Therefore, in the design of the robotic manipulator one has to be quite careful in the selection of the gripper mass.

The results of the selection of different materials for the upper arm and forearm can be an important factor in the design of such systems. In the combination numbers 1 and 2 in the Table 3.4, the natural frequencies are quite close.

Table 3.1

The Dimensions of the Robot

Parameters	Link		
	1	2	3
Description	Shoulder	Upper arm	Forearm
Angle(Degrees)	θ_1	θ_2	θ_3
Length (m)	0	1.016	1.5113
Cross-Section	-	Hollow Square	Hollow Circle
Outer Dimensions (m)	-	0.1524	0.1524
Inner Dimensions (m)	-	0.1143	0.1143
Material	-	Steel	Aluminum Alloy

Table 3.2

Various Parameters and the Elements of Direction Cosine Matrix

Parameters	Spatial Link, Fig. 3.5	Forearm of a Robotic Manipulator, Fig. 1.1b
X_A^*, Y_A^*, Z_A^* (m)	10., 10., 8.5	0.543, 0.068, 0.856
X_B^*, Y_B^*, Z_B^* (m)	0.0, 0.0, 0.0	2.0, 0.25, 0.5
X_P^*, Y_P^*, Z_P^* (m)	10., 10., 0.0	0., 10., 0.0
$(x_m)_P$, $(y_m)_P$, $(z_m)_P$	4.379, -7.285, 0.0	0.879, 9.944, 0.0
l_1, m_1, n_1	-0.606, -0.606, -0.516	0.964, 0.120, -0.235
l_2^I, m_2^I, n_2^I	0.366, 0.366, -0.856	0.225, 0.096, 0.970
l_3^I, m_3^I, n_3^I	-0.707, 0.707, 0.000	0.139, -0.988, 0.066
k_x, k_y, k_z, θ	-0.437, 0.559, 0.704, -2.971	-0.290, 0.834, 0.470, 0.28

Table 3.3

The Effect of the Variation of the Gripper Mass on the Natural Frequencies

No.	Mass of the gripper Kg	Frequencies (Hertz)					
		First	Second	Third	Fourth	Fifth	Sixth
1	25	21.755	22.661	70.805	76.259	225.466	260.296
2	20	23.500	24.448	72.571	76.563	228.263	263.755
3	15	25.738	26.729	74.657	77.007	232.452	268.850
4	10	28.751	29.770	77.295	77.714	239.390	277.233
5	5	33.093	34.082	79.007	81.086	253.027	293.864
6	0	39.994	40.706	82.036	88.125	290.806	342.667

Table 3.4

Effect of the Variation of the Materials of the Links on the Natural Freqencies

Combination No.	Material		Frequencies (Hertz)					
	Upper Arm	Forearm	First	Second	Third	Fourth	Fifth	Sixth
1	Steel	Aluminum	21.755	22.661	70.805	76.259	225.466	260.296
2	Steel	Steel	21.674	22.840	70.664	72.619	231.392	259.625
3	Aluminum	Steel	14.018	14.768	46.358	56.446	213.234	228.863
4	Aluminum	Aluminum	16.001	16.802	51.016	54.390	195.854	236.593

In these combinations, steel has been used as the structural material for the upper arm where the bending moment is higher. The use of the steel for the forearm does not affect the frequencies very much. On the other hand, when aluminum alloy is used as the structural material for the upper arm, the natural frequencies significantly decrease owing to the lower modulus of elasticity of the aluminum alloy.

The third parameter varied was the ratio of the outer and inner diameters of the forearm which had a hollow circular section. The fourth variable parameter was the ratio of the lengths of the outer and inner sides of the square cross section of the upper arm. These results are shown in the Tables 3.5 and 3.6 respectively. In these cases, the outer dimensions were fixed but the inner dimensions were varied. The results in these tables show that the natural frequencies are not very sensitive to these variations.

The location of the end effector can be important in the study of the variation of the natural frequencies. The Table 3.7 shows that the natural frequencies corresponding to various locations of the end effector on the specified trajectory shown in Fig. 2.9, do not change very significantly. This is an useful information because if the robot is used to perform a welding job where the trajectory is as shown in the Fig. 2.9 then this natural frequency

Table 3.5

The Effect of the Variation of D_o/D_i of the Forearm on the Natural Frequencies

No.	$\frac{D_o}{D_i}$	Frequencies (Hertz)					
		First	Second	Third	Fourth	Fifth	Sixth
1	1.333	21.755	22.661	70.805	76.259	225.461	260.296
2	1.5	21.892	22.869	70.240	74.981	218.983	252.065
3	2.000	21.505	22.534	68.653	72.876	208.782	237.918
4	3.000	20.980	22.013	67.077	71.339	201.124	227.097
5	6.000	20.602	21.631	65.968	70.392	196.217	220.215

Table 3.6

The Effect of the Change in S_o/S_i of the Upperarm on the Natural Frequencies

No.	$\frac{S_o}{S_i}$	Frequencies (Hertz)					
		First	Second	Third	Fourth	Fifth	Sixth
1	1.333	21.755	22.661	70.805	76.259	225.461	260.296
2	1.5	22.502	23.379	73.400	79.678	230.377	262.533
3	2	23.177	24.013	75.311	81.863	234.874	264.148
4	3	23.390	24.206	75.215	81.451	236.149	264.323
5	6	23.434	24.434	74.647	80.549	236.290	264.114

Table 3.7

The Effect of the Gripper Position on the Natural Frequency

Gripper Position Coordinates			Frequencies (Hertz)					
X	Y	Z	First	Second	Third	Fourth	Fifth	Sixth
2	0.25	0.5	21.754	22.660	70.808	76.276	226.513	262.130
2	0	0.5	21.791	22.717	70.488	76.012	226.100	261.965
2	-0.25	0.5	21.754	22.660	70.807	76.276	226.513	262.130
2	-0.25	0.25	21.866	22.830	69.881	75.503	225.357	261.658
2	-0.25	0.0	21.905	22.888	69.595	75.260	225.026	261.513
2	0.0	0.0	21.944	22.945	69.318	75.024	224.720	261.376
2	0.25	0.0	21.905	22.888	69.595	75.260	225.026	261.514
2	0.25	0.25	21.866	22.831	69.882	75.503	225.357	261.657

information assures that the dynamic stiffness will be almost same all along the trajectory. In fact, the natural frequency variation can serve as an important criteria for the trajectory planning of the robotic structures. It is quite obvious that when both the upper and forearm extend parallel to either X_o or Y_o axis in Fig. 2.9, this system would have very low natural frequency. These types of configurations are undesirable. Therefore, the trajectory should be such that the robotic structures should have sufficiently high stiffness or the natural frequencies and the variation of the natural frequencies should be a minimum. If the natural frequencies do not change very appreciably then one can safely assume that the stiffness matrix is nearly constant over the trajectory because the inertia matrix is constant. Therefore, while carrying out the transient analysis, or the steady state analysis later on, the computations would be greatly reduced. Finally, the natural frequencies of the control system must be lower than the natural frequencies of the structure [9]; therefore, the parametric variations study such as in the present work, would also be helpful in the design of the control systems.

In the natural frequency calculations the total number of degrees of freedom were 156 and the inertia matrices were calculated using the lumped mass approach. This was done because these lumped type of inertia matrices

are invariant with position of the end effector. However, the other matrices have to be computed based on the consistent matrix formulation. But, the disadvantage in using the lumped mass approach is that for a given level of accuracy the larger number of degrees of freedom have to be used as compared to a situation where the inertia matrices are calculated using the consistent formulation. In either of these cases, one can reduce the degree of freedom further by using the dynamic condensation techniques. In the next section, the use of these condensation techniques where the system matrices were calculated using the consistent formulation is discussed.

3.4 Various Methods for the Martix Size Reduction

It was mentioned earlier in the literature survey that two methods have been used to reduce the size of each of the system matrices in the study of the link dynamics. However, presently there is no information available regarding the relative effectiveness of these two techniques. Therefore, in the present investigation a comparative study regarding these two techniques is carried out.

3.4.1 The Guyan's Reduction Technique

In the Guyan's reduction technique [43], the selection of master degrees of freedom or the retained

degrees of freedom, r, and the slave degrees of freedom or the discarded degrees of freedom, d, is carried out by scanning the diagonal terms of [M] and [K]. The ith degree of freedom for which K_{ii}/M_{ii} is largest is selected as the first slave. This process is continued until required number of slaves are chosen. This criteria of selection of the slave degrees of freedom ensures accurate representation of the lower vibration modes in the condensed system. After the selection of the slaves the dynamic equation of motion is represented in the following rearranged form:

$$\begin{bmatrix} [M_{rr}] & [M_{rd}] \\ [M_{dr}] & [M_{dd}] \end{bmatrix} \begin{bmatrix} \ddot{w}_r(t) \\ \ddot{w}_d(t) \end{bmatrix} + \begin{bmatrix} [C_{rr}] & [C_{rd}] \\ [C_{dr}] & [C_{dd}] \end{bmatrix} \begin{bmatrix} \dot{w}_r(t) \\ \dot{w}_d(t) \end{bmatrix}$$

$$+ \begin{bmatrix} [K_{rr}] & [K_{rd}] \\ [K_{dr}] & [K_{dd}] \end{bmatrix} \begin{bmatrix} w_r(t) \\ w_d(t) \end{bmatrix} = - \begin{bmatrix} F_r(t) \\ F_d(t) \end{bmatrix} \quad (3.85)$$

Then, the transformation matrix $[\phi]$ for the matrix size reduction is expressed as

$$[\phi] = \begin{bmatrix} I \\ -[K_{dd}]^{-1}\,[K_{rd}]^T \end{bmatrix} \qquad (3.86)$$

Thus, the dynamic equation of motion of the condensed system is written as

$$[M_{cd}]\{\ddot{w}_r(t)\} + [C_{cd}]\{\dot{w}_r(t)\} + [K_{cd}]\{w_r(t)\} = \{F_{cd}(t)\} \qquad (3.87)$$

where

$$[M_{cd}] = [\phi]^T[M][\phi]$$

$$[C_{cd}] = [\phi]^T[C][\phi] \quad \text{and}$$

$$[K_{cd}] = [\phi]^T[K][\phi] \,.$$

$$\{F_{cd}(t)\} = [\phi]^T\{F(t)\}$$

3.4.2 The Component Mode Synthesis

This is another important technique for reducing the size of the matrices [22]. The basic procedure involves separating the complete set of system coordinates $\{w(t)\}$ into a set of interface coordinates $\{w(t)\}^I$ and into a set of free coordinates $\{w(t)\}^F$, where the free coordinates describe the

system's internal degrees of freedom. Hence, $\{w(t)\}$ can be written as

$$\{w(t)\} = \begin{bmatrix} w(t)^I \\ w(t)^F \end{bmatrix} \tag{3.88}$$

After partitioning the system structural stiffness matrix [K], the matrix equation governing the static behaviour of the system is given as

$$\begin{bmatrix} [K^{II}]_{nxn} & [K^{IF}]_{nxm} \\ [K^{FI}]_{mxn} & [K^{FF}]_{mxm} \end{bmatrix} \begin{bmatrix} w(t)^I \\ w(t)^F \end{bmatrix} = \begin{bmatrix} F(t)^I \\ F(t)^F \end{bmatrix} \tag{3.89}$$

where $\{F(t)^I\}$ and $\{F(t)^F\}$ represent the forces on the interface and free degrees of freedom respectively. Condensation is carried out by replacing the large number of coordinates in the free displacements $\{w(t)^F\}$ with a smaller set of modal coordinates. The actual free displacements $\{w(t)^F\}$ can be described by adding a vector which is associated with the interface displacements $\{w(t)^I\}$ with a

vector which is dependent on a small set of modal coordinates. Mathematically, this is written as

$$\underset{mx1}{\{w(t)^F\}} = \underset{mx1}{\{w(t)^F\}^I} + \underset{mx1}{\{w(t)^F\}^M} \tag{3.90}$$

The $\{w(t)^F\}^I$ displacements result from the motion of the $\{w(t)^I\}$ coordinates when the forces on the free coordinates are zero, i.e., $\{F(t)^F\} = \{0\}$. Solving for $\{w(t)^F\}$ in terms of $\{w(t)^I\}$ in Eqn. (3.89) and redefining this term as $\{w(t)^F\}^I$ produces

$$\underset{mx1}{\{w(t)^F\}^I} = -\underset{mxm}{[K^{FF}]^{-1}}\underset{mxn}{[K^{FI}]}\underset{nx1}{\{w(t)^I\}} \tag{3.91}$$

The $\{w(t)^F\}^I$ deformations are synthesized from the stiffness properties of the system and represent the static deformation states which result from the motion of the interface coordinates.

The modal coordinates are obtained by solving the eigenvalue problem with the interface coordinates fixed, i.e. $\{w(t)^I\} = \{0\}$.

$$[M^{FF}]\{\ddot{w}(t)^F\}^M + [K^{FF}]\{w(t)^F\}^M = \{0\} \qquad (3.92)$$

where $[M^{FF}]$ is obtained by partitioning the mass matrix [M] similar to [K] shown in Eqn. (3.89). The solution of the eigenvalue problem gives

$$\{w(t)^F\}^M = [\Phi]\{\eta\} \qquad (3.93)$$

where $[\Phi]$ contains the reduced set of eignevectors. The $\{w(t)^F\}$ vector in Eqn. (3.90) can now be written as

$$\{w(t)^F\} = -[K^{FF}]^{-1}\,[K^{FI}]\{w(t)^I\} + [\Phi]\{\eta\} \qquad (3.94)$$

Thus the linear transformation can now be written as

$$\{w(t)\} = \begin{bmatrix} w(t)^I \\ w(t)^F \end{bmatrix} = \begin{bmatrix} [I] & [0] \\ -[K^{FF}]^{-1}[K^{FI}] & [\Phi] \end{bmatrix} \begin{bmatrix} w(t)^I \\ \eta(t) \end{bmatrix}$$

$$= [T_c]\{w_c(t)\} \qquad (3.95)$$

where the transformation matrix $[T_c]$ can be written as

$$[T_c]_{(m+n)x(n+r)} = \begin{bmatrix} [I]_{nxn} & [0]_{nxr} \\ -[K^{FF}]^{-1}_{mxm}[K^{FI}]_{mxn} & [\Phi]_{mxr} \end{bmatrix} \quad (3.96)$$

This non-square matrix can be used to transform the unreduced dynamic matrices to a much smaller size. Substituting for $\{w(t)\}$ from Eqn. (3.95) into Eqn. (3.88) and premultiplying by $[T_c]^T$ one obtains

$$[M_{cc}]\{\ddot{w}_c(t)\} + [C_{cc}]\{\dot{w}_c(t)\} + [K_{cc}]\{w_c(t)\} = \{F_{cc}(t)\} \quad (3.97)$$

where

$$[M_{cc}] = [T_c]^T [M] [T_c]$$

$$[C_{cc}] = [T_c]^T [C] [T_c]$$

$$[K_{cc}] = [T_c]^T [K] [Tc]$$

$$[F_{cc}(t)] = [T_c]^T \{F(t)\}$$

The suffix cc denotes the dynamic matrices and force vectors obtained using the component mode synthesis.

3.4.3 The Study of the Machine-Tool Spindle

In order to study the use of the two condensation techniques dicussed earlier, the lathe spindle shown in Fig. 3.6 was divided into twelve finite beam elements with two degrees of freedom at each node. This spindle is supported by two bearings which were represented by a linear spring and a damper. The discretization of this spindle is also shown in this figure. The details about this spindle are given in the Table 3.8. It has been experimentally established that for this system the workpiece-live center connection should be approximated as a hinged connection [41]. The damped natural frequencies of this system were calculated by using the velocity vector as an auxiliary variable and n second order differential equations were transformed into 2n first order differential equations. The details of the calculation of the damped natural frequencies are given in [44]. Using this technique, the damped natural frequencies of this system were calculated as per Eqns. (3.87) and (3.97) which are the condensed form of the Guyan's reduction scheme and the component mode synthesis respectively. The results obtained [45] are shown in the Table 3.9.

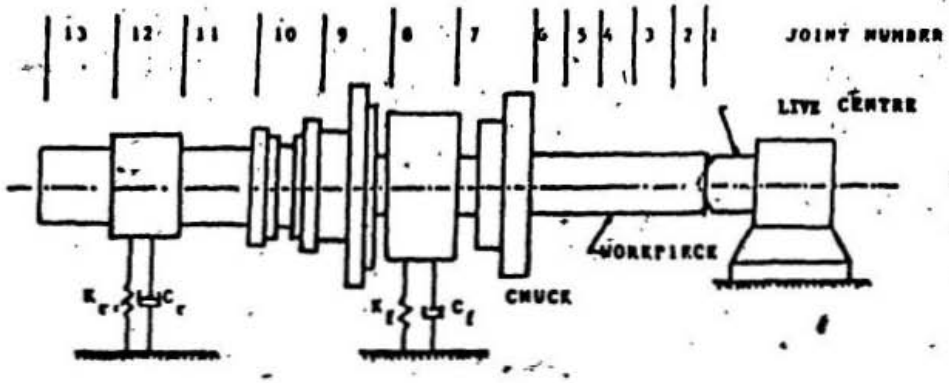

Fig. 3.6: Schematic Model of a Lathe Spindle-Workpiece System

Table 3.8

Parameter Values of the Lathe Spindle-Workpiece System [43]

Parameters	Values of Parameters
The Modulus of Elasticity	$206.456 \times 10^9 \frac{N}{m}$
The Mass of the Chuck	34.012 Kg.
The Diameter of the Chuck	0.254 m
The Stiffness of the Front Bearing (K_F)	$2.2703939 \times 10^9 \frac{N}{m}$
The Stiffness of the Rear Bearing (K_R)	$7.1172232 \times 10^8 \frac{N}{m}$
The Damping at the Front Bearing Location (C_F)	$28.632 \times 10^3 \frac{N\text{-sec}}{m}$
The Damping at the Rear Bearing Location (C_R)	$22.328 \times 10^3 \frac{N\text{-sec}}{m}$
The Length of the Workpiece	0.3302 m
The Length of the Spindle	0.822325 m
The Diameter of Elements 1 - 5	0.051 m
The Diameter of Element 6	0.254 m
The Diameter of Element 7	0.200 m
The Diameter of Element 8	0.228 m
The Diameter of Elements 9, 10, and 12	0.170 m
The Diameter of Element 11	0.188 m

Table 3.9

The Comparison of Damped Natural Frequencies of the Lathe Spindle Obtained by the Condensation Techniques

Number of Master Degrees of Freedom	The First Natural Frequency (Hertz)		The Second Natural Frequency (Hertz)		The Third Natural Frequency (Hertz)		The Fourth Natural Frequency (Hertz)		The Fifth Natural Frequency (Hertz)	
	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode
5	585.18	517.78	796.30	675.21	1610.71	1214.75	5384.27	2066.97	12931.96	7435.41
6	520.66	517.90	780.27	675.04	1227.35	1202.81	1697.18	1555.77	9416.96	6652.36
7	517.76	516.77	776.46	675.04	1218.88	1198.83	1696.71	1552.26	5294.47	3744.60
8	517.64	516.77	772.07	675.04	1218.36	1198.36	1692.30	1551.15	4685.23	5733.46
9	517.62	516.69	724.88	675.04	1214.13	1197.56	1658.30	1550.67	3971.08	3723.90
10	517.25	516.69	689.89	675.04	1210.18	1197.40	1592.01	1550.62	3959.63	3720.73
11	516.70	516.69	675.83	675.04	1199.26	1197.24	1555.25	1550.51	3780.81	3719.13
12	516.63	516.69	674.74	675.05	1197.65	1197.24	1552.00	1550.35	3722.22	3717.54
13	516.62	516.69	674.73	675.05	1197.64	1197.24	1552.00	1550.35	3722.18	3717.54
14	516.63	516.69	674.75	675.05	1197.64	1197.24	1551.99	1550.35	3722.18	3717.54
25	516.69	516.69	675.05	675.05	1197.56	1197.56	1551.95	1551.95	3717.53	3717.54

In this table, due to the hinged boundary condition, the total number of the degrees of freedom for the 12 elements having two degrees of freedom at each node are 25. The system matrices were condensed to smaller sizes by varying the master degrees of freedom and the corresponding natural frequencies are shown here. It is evident from this table that fewer the master degrees of freedom greater is the deviation (the 25 degrees of freedom have zero deviation) of a particular natural frequency. Another important fact to be noted is that higher the natural frequency greater are the number of the master degrees of freedom required to attain a certain deviation. Thirdly, as the master degrees of freedom are increased the approximate frequencies approach the frequency corresponding to the full system (25 degrees of freedom) from the higher side, i.e. the deviation is positive. In the case of the component mode synthesis, the number of the interface coordinates remain fixed but the number of the modal coordinates are varied. The results very clearly show the superiority of the use of the component mode synthesis. If the condensed system is to be used to represent the first five modes very accurately then one should use a minimum of 12 degrees of freedom.

3.4.4 The Study of the Robotic Manipulator

The use of the industrial robots is increasing every day. In order to save CPU times in the dynamic analysis of these types of systems it was thought that the present type of study would be quite useful. It should be borne in mind that these robotic structures are more flexible than the machine tools and the stiffness as well as the inertia matrices are nonlinear because they are dependent on the position of the end effector. One such robotic system is shown in the Fig. 1.1b, and its kinematic model is shown in the Fig. 2.8.

In the Fig. 2.8, the moving coordinate system (x_3, y_3, z_3) is located at the end effector. The system matrices i.e. [M], [K] etc. vary depending on the location of the end effector when measured with respect to the global coordinate system (X_o, Y_o, Z_o). In the present study the global coordinates of the end effector were (2, 0.25, 0.5).

Damping was not included for this system. The stiffness and the inertia matrices were obtained after carrying out the inverse kinematic analysis corresponding to a given position of the end effector as shown in Chapter 2. Since a robot is represented by a space-frame structure, the finite beam elements in this case had six degrees of freedom at each node. These degrees of freedom include the three displacements and three rotations. The local and the global

set of coordinate systems were related by the matrix of direction cosines for the space-frames as explained in Section 3.2.6. The various details of this manipulator are given in the Table 3.1, and the undamped natural frequencies of this system using the two condensation techniques are shown in the Table 3.10.

There were 66 total number of degrees of freedom, which were condensed between 10 and 15. The table shows that the first four modes can be roughly represented by using 11 degrees of freedom by using either of the two techniques but to include the first six modes, the component mode systhesis is certainly better and it would require 13 degrees of freedom; on the other hand, to represent the system with the same degrees of accuracy, it was found (not shown in the Table 3.10) that it would need 36 degrees of freedom if the Guyan's reduction technique is to be used. Therefore, in this case also the use of the component mode synthesis would be recommended over the Guyan's reduction technique.

3.5 The Conclusions

In this chapter, the dynamic analysis of the robotic manipulator was carried out by first deriving the dynamic equations of motion. To transform the link coordinate system along the global coordinate system, the matrix of direction cosines were used and four new methods of

Table 3.10

The Comparison of Undamped Natural Frequencies of the Robotic Manipulator by Various Condensation Techniques

Number of Master Degrees of Freedom	The First Natural Frequency (Hertz)		The Second Natural Frequency (Hertz)		The Third Natural Frequency (Hertz)		The Fourth Natural Frequency (Hertz)		The Fifth Natural Frequency (Hertz)		The Sixth Natural Frequency (Hertz)	
	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode	Guyan's Reduction	Component Mode
10	21.756	21.755	22.676	22.662	70.859	70.816	76.417	76.257	244.596	225.679	533.562	260.685
11	21.755	21.755	22.676	22.662	70.859	70.806	76.358	76.257	230.732	225.679	533.540	260.462
12	21.755	21.753	22.676	22.662	70.859	70.806	76.324	76.257	226.501	225.535	533.535	260.462
13	21.753	21.753	22.676	22.662	70.859	70.806	76.313	76.257	225.707	225.535	533.511	260.350
14	21.753	21.753	22.676	22.662	70.859	70.805	76.312	76.257	225.640	225.472	533.495	260.350
15	21.753	21.753	22.664	22.662	70.846	70.805	76.310	76.257	225.634	225.472	293.553	260.307
66	21.753	21.753	22.660	22.660	70.803	70.803	76.257	76.257	225.454	225.454	260.288	260.288

calculating these matrices were discussed. The variation of the natural frequencies of the robotic manipulator as a function of several design parameters were studied next, and finally, two dynamic condensation techniques were analyzed. The conclusions that can be drawn from these studies are:

1. The matrix of direction cosines of the link coordinate system can be calculated using the four new methods discussed in this work. Some of these new methods are much easier to use as compared to the one presently used.
2. The natural frequencies of the robotic manipulators are quite sensitive to any variations of the mass of the gripper.
3. The upper arm should be made of a material which should have a high modulus of elasticity.
4. The large variations of the natural frequencies along a trajectory are undesirable.
5. Both, the Guyan's reduction technique and the component mode synthesis can be used to reduce the size of the system matrices but, the degrees of freedom required to represent the uncondensed system in the case of the component mode synthesis are less than the other technique.

CHAPTER 4

CONCLUSIONS AND RECOMMENDATIONS

4.1 A Brief Discussion About This Investigation and the Conclusions

The objective of this work was to carry out the kinematic as well as the dynamic analyses of the robotic manipulators. The kinematic analysis involved the displacement, velocity, and acceleration analysis of the planar and spatial links. The techniques used for these purposes were: (a) the algebraic method, (b) the Newton-Raphson technique, (c) the Hooke and Jeeves method, (d) the Complex optimization method, and (e) the modified Newton-Raphson technique. To carry out the dynamic analysis, the equations of motion were discussed in detail and appropriate approximations were made. To express link coordinate system in terms of the global coordinate system, several methods of obtaining the matrix of direction cosines were analyzed. The effects of the variation of the design variables on the system natural frequencies were studied next and finally, two techniques to reduce the size of system matrices were discussed in detail and the better of the two suggested. Based on the studies carried out in this investigation the following conclusions can be drawn:

1. The Hooke and Jeeves method can be successfully used to perform the displacement analysis of the planar and space closed loop mechanisms.
2. The complex optimization method can yield better solution as compared to the modified Newton-Raphson technique for solving inverse displacement analysis problems involving the robotic manipulators.
3. If the open loop mechanisms or robotic manipulators have three degrees of freedom then the exact values of the angular velocities and angular accelerations can be obtained.
4. The modified Newton-Raphson technique or other numerical methods such as the optimization principles have to be used for velocity and acceleration analyses in situations where the degrees of freedom are different from three in space mechanisms or robotic manipulators.
5. The matrix of direction cosines of the link coordinate system can be calculated using the four new methods discussed in this work. Some of these new methods are much easier to use as compared to the one presently used.
6. The natural frequencies of the robotic manipulators are quite sensitive to any variations of the mass of the gripper.

7. The upper arm should be made of a material which should have a high modulus of elasticity.
8. The large variations of the natural frequencies along a trajectory are undesirable.
9. Both, the Guyan's redution technique and the component mode synthesis can be used to reduce the size of the system matrices but, the degrees of freedom required to represent the uncondensed system in the case of the component mode synthesis are less than the other technique.

4.2 Limitations of the Investigtion and Recommendations for Future Work

The kinematic and dynamic analyses carried out in this work had the following limitations:

1. In the derivation of the dynamic equation, the link lengths were of constant magnitude, because of the rotary joints. In the more general approach, the link lengths can be considered as a variable.
2. In the present model the multiple-loop linkages could not be analyzed. This can be an important area for future work.
3. The design of the optimal control systems for the robotic manipulator can be analyzed as a part of the future work.

4. The forced response and the transient response analysis can be done as a natural extension of this work.
5. The non-linearities introduced due to the gear drives or chain drives can be an interesting area for future work.

REFERENCES

1. Sunada, W. H., "Dynamic Analysis of Flexible Spatial Mechanisms and Robotic Manipulators", Ph.D. Thesis, University of California, Los Angeles, 1981.

2. Shigley, J. E. and Uicker, Jr. J. J., "Theory of Machines and Mechanisms", McGraw Hill Book Company, N.Y., 1980.

3. Wilson, C. E., et. al., "Kinematics and Dynamics of Machinery", Harper & Row, Publishers, N.Y., 1985.

4. Suh, C. H., and Radcliffe, C. W., "Kinematics and Mechanisms Design", John Wiley and Sons, 1978.

5. Burton Paul, "Kinematics and Dynamics of Planar Machinery", Prentice-Hall, N.J., 1979.

6. Uicker, Jr. J. J., Denavit, S. and Hartenberg, R. S., "An Iterative Method for the Displacement Analysis of Spatial Mechanisms", Journal of Applied Mechanics, Vol. 31, Trans. ASME Vol. 86, Series E, 1964, pp. 309-314.

7. Hall, Jr., A. S., Root, R. R. and Sandgren, E., "A Dependable Method for Solving Matrix Loop Equations for the General Three-dimensional Mechanism", Journal of Engineering for Industry, Aug. 1977, pp. 547-550.

8. Turcic, D., "A Point-Oriented Kinematic Analysis Procedure for Unconstrained Mechanisms with Interactive Computer Graphics Applications", M. S. Thesis, The Pennsylvania State University, University Park, PA., Aug. 1979.

9. Paul, R. P., "Robot Manipulators: Mathematics, Programming and Control", MIT Press, Massachusetts, 1981.

10. Craig, J. J., "Introduction to Robotics Mechanics & Control", Addison-Wesley Publishing Co., California, 1985.

11. Duffy, J., and Crane, C., "A Displacement Analysis of the General Spatial 7-link, 7R Mechanism", Mechanism Mach. Theory 15(3):153-159, 1980.

12. Tsai, L. W., and Morgan, A. P., "Solving the Kinematics of the Most General Six- and Five-Degrees-of-Freedom Manipulators by Continuation Methods", Asme paper 84-DET-20, ASME Design Engineering Technical Conference, Cambridge, Mass.

13. Angeles, J., "On the Numerical Solution of the Inverse Kinematic Problem", The International Journal of Robotics Reserch, Vol. 4, No. 2, Summer 1985.

14. Stanisic, M. M. and Pennock, G. R., "A Nondegenerate Kinematic Solution of a Seven-Jointed Robot Manipulator", The International Journal of Robotics Research, Vol. 4, No. 2, Summer 1985.

15. Yoshikawa, T., "Manipulability of Robotic Mechanisms", The International Journal of Robotics Research, Vol. 4, No. 2, Summer 1985.

16. Woo, L. S., and F. Freudenstein, "Dynamic Analysis of Mechanisms Using Screw Coordinates, "Transactions of the ASME, Journal of Engineering for Industry, Vol. 93, No. 1, pp. 273-276, February 1971.

17. Yang, A. T., "Inertia Force Analysis of Spatial Mechanisms", Transactions of the ASME, Journal of Engineering for Industry, Vol. 93, No. 1, pp. 27-33, February 1971.

18. Uicker, J. J., "Dynamic Behaviour of Spatial Linkages", Transactions of the ASME, Journal of Engineering for Industry, Vol. 91, No. 1, pp. 251-258, February 1969.

19. Maatuk, J., "A Study of the Dynamics and Control of Flexible Spatial Manipulators", Ph.D. Thesis, Mechanics and Structures Department, University of California, Los Angeles, 1976.

20. Winfrey, R. C., "Elastic Link Mechanism Dynamics, "Transactions of the ASME, Journal of Engineering for Industry, Vol. 93, No. 1, pp. 268-272, February 1971.

21. Winfrey, R. C., "Dynamic Analysis of Elastic Mechanisms by Reduction of Coordinates", with Discussion by I. Iman, A. G. Erdman, and G. N. Sandor, Transactions of the ASME, Journal of Engineering for Industry, Vol. 94, No. 1, pp. 557-581, May 1972.

22. Hurty, W. C., "Dynamic Analysis of Structural Systems Using Component Modes", AIAA Journal, Vol. 3, No. 4, April 1965.

23. Thompson, W. T., "Theory of Vibration with Applications", Prentice Hall, Inc., NJ, 1981.

24. Mirro, J., "Automatic Feedback Control of a Vibrating Beam", Master's Thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, Rept. T-571, Cambridge, Mass, C. S. Draper Laboratory, August 1972.

25. Book, W. J., "Modeling Design and Control of Flexible Manipulator Arms", Ph.D. thesis, Massachusetts Institute of Technology Department of Mechanical Engineering, April 1974.

26. Book, W. J., Maizza-Neto, O., and Whitney, D. E., "Feedback Control of Two Beam, Two Joint Systems with Distributed Flexibility", Trans. ASME J. Dyn. Syst. Measurement Contr. 97G(4):424-431, Dec. 1975.

27. Dubowsky, S. and Gardner, T. N., "Design and Analysis of Multi-link Flexible Mechanisms with Multiple Clearance Connections", ASME J. Engineering for Industry 99(1), Feb. 1977.

28. Book, W. J., "Recursive Lagrangian Dynamics of Flexible Manipulator Arms", The International Journal of Robotics Research, Vol. 3, No. 3, Fall 1984.

29. Naganathan, G., and Soni, A. H., "Non-linear Flexibility Studies for Spatial Manipulators", IEEE International Conference on Robotics and Automation, April 1986.

30. Gupta, K. C., Banerjee, A., and Fox, R. L., "Automated Kinematic Analysis of Planar Mechanisms", ASME Paper 72-Mech.-90, ASME, New York, 1972.

31. Molian, S., "Solution of Kinematic Equtions by Newton's Method", Journal of Mechanical Engineering Science, vol. 10, pp. 360-362.

32. Sharan, A. M. and Subbiah, M., "A Direct-Search Optimization Method for the Displacement Analysis of the Planar and Space Mechanisms", ASME OMAE Conference, Tokyo, April 1986.

33. Rao, S. S., Optimization Theory and Applications, Wiley Eastern Ltd., New Delhi, 1978, pp. 261-265.

34. Subbiah, M. and Sharan, A. M., "The Nonlinear Displacement Analysis of Robotic Manipulators Using Complex Optimization Method", Accepted for Publication in the Journal of Mechanisms and Machine Theory, Jan. 1986.

35. Beveridge, S. G., and Schechter, R. S., "Optimization: Theory and Practice", McGraw-Hill Book Company, New York, 1970, pp. 384-389.

36. Koren, Y., "Robotics for Engineers", McGraw-Hill Book Company, N.Y., 1985.

37. Subbiah, M. and Sharan, A. M., "The Determination of the Matrix of Direction Cosines of Spatial Links and the Natural Frequency Calculations of the Robotic Structures", Sent for publication to the Shock and Vibrations Bulletin.

38. Beaufort, F. W., et. al., "Computer Methods of Structural Analysis", Prentice Hall International, Inc., London, 1970.

39. Hoa, S. V., "Vibration of a Rotating Beam with Tip Mass", Journal of Sound and Vibration, Vol. 67(3), 1979, pp. 369-381.

40. Taylor, S., "The Design of Machine Tool Structures Using a Digital Computer", Proc. 7th Int. Machine Tool Design Conference, Birmingham, 1966.

41. Sharan, A. M., Sankar, T. S., and Sankar, S., "Dynamic Behaviour of Lathe Spindles with Elastic Supports Including Damping by Finite Element Analysis", 51st Shock and Vibrations Bulletin, May 1981.

42. Bollinger, J. G. and Geiger, G., "Analysis of the Static and Dynamic Behaviour of Lathe Spindles", Int. J. Mach. Tool Des. Res., Vol. 3, 1964, pp. 193-209.

43. Reddy, V. R., Sharan, A. M., "Static and Dynamic Analysis of Machine Tools using Dynamic Matrix Reduction Technique", Accepted for publication in the Int. J. of Machine Tool Des. Res., Oct. 1985.

44. Meirovitch, L., "Analytical Methods in Vibrations", the MacMillan Collier Ltd., Toronto, Ontario, 1969, p. 410.

45. Subbiah, M. and Sharan, A. M., "A Study of the Dynamic Condensation Techniques for the Machine Tools and Robotic Manipulators", Accepted for presentation at the Seventh World Congress, IFTOMM, Seville, Spain, September, 1987.

APPENDIX A

EXPANSION OF VELOCITY AND ACCELERATION EQUATIONS

The velocity equation for a spatial open-loop mechanism can be written in the matrix form as

$$\begin{bmatrix} 1 \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & v_{11} & v_{12} & v_{13} \\ 0 & v_{21} & v_{22} & v_{23} \\ 0 & v_{31} & v_{32} & v_{33} \end{bmatrix} \begin{bmatrix} 1 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad \text{(A.1)}$$

where

$$v_{11} = -(l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1) \quad \text{(A.2)}$$

$$v_{12} = -(l_2 \sin\theta_2 \cos\theta_1 + l_3 \sin(\theta_2 + \theta_3) \cos\theta_1) \quad \text{(A.3)}$$

$$v_{13} = -(l_3 \sin(\theta_2 + \theta_3) \cos\theta_1) \quad \text{(A.4)}$$

$$v_{21} = (l_2 \cos\theta_2 \cos\theta_1 + l_3 \cos(\theta_2 + \theta_3) \cos\theta_1) \quad \text{(A.5)}$$

$$v_{22} = -(l_2 \sin\theta_2 \sin\theta_1 + l_3 \sin(\theta_2 + \theta_3) \sin\theta_1) \quad \text{(A.6)}$$

$$v_{23} = -(l_3 \sin(\theta_2 + \theta_3) \sin\theta_1) \quad \text{(A.7)}$$

$$v_{31} = 0 \quad \text{(A.8)}$$

$$v_{32} = l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \quad \text{(A.9)}$$

$$v_{33} = l_3 \cos(\theta_2 + \theta_3) \quad \text{(A.10)}$$

The acceleration equation for a spatial open-loop mechanism can be written in the following matrix form:

$$\begin{bmatrix} 1 \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \begin{bmatrix} 1 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} \qquad \text{(A.11)}$$

where

$$\begin{aligned} A_{21} = {} & 2[l_2 \sin\theta_1 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3)\sin\theta_1]\,\dot{\theta}_1\,\dot{\theta}_2 \\ & - 2[l_3 \cos(\theta_2 + \theta_3)\cos\theta_1]\,\dot{\theta}_2\,\dot{\theta}_3 \\ & + 2[l_3 \sin\theta_1 \sin(\theta_2 + \theta_3)]\,\dot{\theta}_1\,\dot{\theta}_3 \\ & - [l_2 \cos\theta_1 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3)\cos\theta_1]\,\dot{\theta}_1^2 \\ & - [l_2 \cos\theta_1 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3)\cos\theta_1]\,\dot{\theta}_2^2 \\ & - [l_3 \cos(\theta_2 + \theta_3)\cos\theta_1]\,\dot{\theta}_3^2 \end{aligned} \qquad \text{(A.12)}$$

$$A_{22} = -[l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3)\sin\theta_1] \qquad \text{(A.13)}$$

$$A_{23} = -[l_2 \sin\theta_2 \cos\theta_1 + l_3 \sin(\theta_2 + \theta_3)\cos\theta_1] \quad (A.14)$$

$$A_{24} = -[l_3 \sin(\theta_2 + \theta_3) \cos\theta_1] \quad (A.15)$$

$$\begin{aligned} A_{31} = &-[l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_1^2 \\ &- [l_2 \cos\theta_2 \sin\theta_1 + l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_2^2 \\ &- [l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_3^2 \\ &- 2[l_2 \sin\theta_2 \cos\theta_1 + l_3 \sin(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_1\dot{\theta}_2 \\ &- 2[l_3 \cos(\theta_2 + \theta_3) \sin\theta_1]\dot{\theta}_2\dot{\theta}_3 \\ &- 2[l_3 \sin(\theta_2 + \theta_3) \cos\theta_1]\dot{\theta}_1\dot{\theta}_3 \end{aligned} \quad (A.16)$$

$$A_{32} = [l_2 \cos\theta_2 \cos\theta_1 + l_3 \cos(\theta_2 + \theta_3) \cos\theta_1] \quad (A.17)$$

$$A_{33} = -[l_2 \sin\theta_2 \sin\theta_1 + l_3 \sin(\theta_2 + \theta_3) \sin\theta_1] \quad (A.18)$$

$$A_{34} = -[l_3 \sin(\theta_2 + \theta_3) \sin\theta_1] \quad (A.19)$$

$$\begin{aligned} A_{41} = &-[l_2 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3)]\dot{\theta}_2^2 \\ &- l_3 \sin(\theta_2 + \theta_3)\dot{\theta}_2\dot{\theta}_3 - l_3 \sin(\theta_2 + \theta_3)\dot{\theta}_3\dot{\theta}_2 \\ &- l_3 \sin(\theta_2 + \theta_3)\dot{\theta}_3^2 \end{aligned} \quad (A.20)$$

$$A_{42} = 0 \quad (A.21)$$

$$A_{43} = l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \quad (A.22)$$

$$A_{44} = l_3 \cos(\theta_2 + \theta_3) \quad (A.23)$$

APPENDIX B

CONVENTIONAL METHOD OF FORMING GLOBAL MATRICES

The consistent finite element mass and stiffness matrices for a three-dimensional beam element can be written as

$$[m] = \rho A \ell \begin{bmatrix}
\frac{1}{3} & & & & & & & & & & & \\
0 & \frac{13}{35} & & & & & & & & & & \\
0 & 0 & \frac{13}{35} & & & & & & & & & \\
0 & 0 & 0 & \frac{I_x}{3A} & & & & & & & & \\
0 & 0 & -\frac{11\ell}{210} & 0 & \frac{\ell^2}{105} & & & & & & & \\
0 & \frac{11\ell}{210} & 0 & 0 & 0 & \frac{\ell^2}{105} & & & & & & \\
\frac{1}{6} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & & & & & \\
0 & \frac{9}{70} & 0 & 0 & 0 & \frac{13\ell}{420} & 0 & \frac{13}{35} & & & & \\
0 & 0 & \frac{9}{70} & 0 & -\frac{13\ell}{420} & 0 & 0 & 0 & \frac{13}{35} & & & \\
0 & 0 & 0 & \frac{I_x}{6A} & 0 & 0 & 0 & 0 & 0 & \frac{I_x}{3A} & & \\
0 & 0 & \frac{13\ell}{420} & 0 & -\frac{\ell^2}{140} & 0 & 0 & 0 & \frac{11\ell}{210} & 0 & \frac{\ell^2}{105} & \\
0 & -\frac{13\ell}{420} & 0 & 0 & 0 & -\frac{\ell^2}{140} & 0 & -\frac{11\ell}{210} & 0 & 0 & 0 & \frac{\ell^2}{105}
\end{bmatrix} \quad \text{Symmetric} \qquad \text{(B.1)}$$

$$
[k] = \begin{bmatrix}
\frac{EA}{\ell} & & & & & & & & & & & \\
0 & \frac{12EI_z}{\ell^3} & & & & & & & & & & \\
0 & 0 & \frac{12EI_y}{\ell^3} & & & & & \text{Symmetric} & & & & \\
0 & 0 & 0 & \frac{GI_x}{\ell} & & & & & & & & \\
0 & 0 & \frac{-6EI_y}{\ell^2} & 0 & \frac{4EI_y}{\ell} & & & & & & & \\
0 & \frac{6EI_z}{\ell^2} & 0 & 0 & 0 & \frac{4EI_z}{\ell} & & & & & & \\
-\frac{EA}{\ell} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{\ell} & & & & & \\
0 & \frac{-12EI_z}{\ell^3} & 0 & 0 & 0 & \frac{-6EI_z}{\ell^2} & 0 & \frac{12EI_z}{\ell^3} & & & & \\
0 & 0 & \frac{-12EI_y}{\ell^3} & 0 & \frac{6EI_y}{\ell^2} & 0 & 0 & 0 & \frac{12EI_y}{\ell^3} & & & \\
0 & 0 & 0 & \frac{I_x}{\ell} & 0 & 0 & 0 & 0 & 0 & \frac{I_x}{\ell} & & \\
0 & 0 & \frac{-6EI_y}{\ell^2} & 0 & \frac{2EI_y}{\ell} & 0 & 0 & 0 & \frac{6EI_y}{\ell^2} & 0 & \frac{4EI_y}{\ell} & \\
0 & \frac{6EI_z}{\ell^2} & 0 & 0 & 0 & \frac{2EI_z}{\ell} & 0 & \frac{-6EI_z}{\ell^2} & 0 & 0 & 0 & \frac{4EI_z}{\ell}
\end{bmatrix} \qquad \text{(B.2)}
$$

If lumped mass matrix is used, then the mass of the finite element is lumped at both ends and this mass is present in all the three translation degrees of freedom in one node.

After getting the elemental mass and stiffness matrices, they are pre and post multiplied by transformaton matrix which is given as

$$[T1] = \begin{bmatrix}
l_1 & m_1 & n_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
l_2 & m_2 & n_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
l_3 & m_3 & n_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & l_1 & m_1 & n_1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & l_2 & m_2 & n_2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & l_3 & m_3 & n_3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & l_1 & m_1 & n_1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & l_2 & m_2 & n_2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & l_3 & m_3 & n_3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & l_1 & m_1 & n_1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & l_2 & m_2 & n_2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & l_3 & m_3 & n_3
\end{bmatrix}$$

where l_1, m_1, n_1 are given in Eqns. (3.51) - (3.53)

$$l_2 = \frac{-l_1 m_1 \cos \alpha - n_1 \sin \alpha}{\sqrt{l_1^2 + n_1^2}} \tag{B.4}$$

$$m_2 = \sqrt{l_1^2 + n_1^2} \cos \alpha \tag{B.5}$$

$$n_2 = \frac{-m_1 n_1 \cos \alpha + l_1 \sin \alpha}{\sqrt{l_1^2 + n_1^2}} \tag{B.6}$$

$$l_3 = \frac{l_1 m_1 \sin \alpha - n_1 \cos \alpha}{\sqrt{l_1^2 + n_1^2}} \tag{B.7}$$

$$m_3 = -\sqrt{l_1^2 + n_1^2} \sin \alpha \tag{B.8}$$

$$n_3 = \frac{m_1 n_1 \sin \alpha + l_1 \cos \alpha}{\sqrt{l_1^2 + n_1^2}} \tag{B.9}$$

In Eqns. (B.4 - B.9) angle α is found by specifying the coordinates of ~~a point~~ P which lies ~~in the local~~ $x_m - y_m$ plane.

$$\alpha = \tan^{-1} \left(\frac{z_{p\gamma}}{y_{p\gamma}}\right) \qquad \text{(B.10)}$$

where

$$y_{p\gamma} = \frac{-l_1 m_1}{\sqrt{l_1^2 + n_1^2}} x_{ps} + \sqrt{l_1^2 + n_1^2}\, y_{ps} - \frac{m_1 n_1}{\sqrt{l_1^2 + n_1^2}} z_{ps} \qquad \text{(B.11)}$$

$$z_{p\gamma} = -\frac{n_1}{\sqrt{l_1^2 + n_1^2}} x_{ps} + \frac{l_1}{\sqrt{l_1^2 + n_1^2}} z_{ps} \qquad \text{(B.12)}$$

and x_{ps}, y_{ps} and z_{ps} are the coordinates of the point P with respect to the global axes.

APPENDIX C

EXPANSION OF FORCE VECTOR

In order to write the computer program to calculate the total force vector for a robotic manipulator, the force terms are expanded here.

Rewriting Eqn. (3.28) we have

$$f_{i\alpha} = (f_{i\alpha})_1 - (f_{i\alpha})_2 - (f_{i\alpha})_3 + (f_{i\alpha})_4 \qquad (C.1)$$

where

$(f_{i\alpha})_1$ is due to the externally applied forces and torques,

$(f_{i\alpha})_2$ is the velocity force,

$(f_{i\alpha})_3$ is the acceleration force and

$(f_{i\alpha})_4$ is the gravity force.

From Eqn. (3.46), we have

$$(f_{i\alpha})_2 = [E_i] \odot [J_{i\alpha}] \qquad (C.2)$$

where

$$[E_i] = \sum_{i=1}^{NL} \sum_{k=1}^{NL} [C_{ijk}]\, \dot{\theta}_j\, \dot{\theta}_k \qquad (C.3)$$

$$[C_{ijk}] = [T_o^i]^T \odot [U_{ijk}] \qquad (C.4)$$

$[U_{ijk}]$ is given in Eqn. 3.26 and

$$[J_{i\alpha}] = \sum_{g=1}^{NG(i)} m_{ig}\, \Phi_{ig\alpha}\, \underline{b}_{ig}^T \qquad (C.5)$$

Let us consider the forearm of the T3R3 model robotic manipulator. This is the third link in the system i.e. i=3. In this link, the first node has six degrees of freedom, three displacements and three rotations. This node has six perturbation coordinates corresponding to the six degrees of freedom. The force term corresponding to the first perturbation coordinate, i.e., the x-displacement degree of freedom of the node, is derived here,

$$(f_{i\alpha})_2 = [E_i] \odot [J_{i\alpha}] \qquad (C.6)$$

$$(f_{31})_2 = \underset{4x4}{[E_3]} \odot \underset{4x4}{[J_{31}]} \qquad (C.7)$$

$$[E_3] = \sum_{j=1}^{3} \sum_{k=1}^{3} \underset{4x4}{[C_{3jk}]}\, \dot{\theta}_j\, \dot{\theta}_k \qquad (C.8)$$

The right side of the Eqn. (C.7) will have nine terms summed together. One of these terms is taken for which j=1 and k=2, and it is expanded below,

$$\underset{4x4}{[C_{312}]}\, \dot{\theta}_1\, \dot{\theta}_2 = \underset{4x4}{[T_o^3]^T}\, \underset{4x4}{[U_{312}]}\, \dot{\theta}_1\, \dot{\theta}_2 \qquad (C.9)$$

where

$$[T_o^3] = [T_o^1][T_1^2][T_2^3] \qquad (C.10)$$

and

$$[U_{312}] = [T_o^o][Q][T_o^1][Q][T_1^3] \qquad (C.11)$$

In this way all the nine terms in the right hand of the Eqn. (C.8) are calculated and they are added to get $[E_3]$.

Now $[J_{31}]$ in the Eqn. (C.7) is calculated next,

$$[J_{31}] = \sum_{g=1}^{NG(3)} m_{3g} \underset{4x1}{\underline{\Phi}_{3g1}} \underset{1x4}{\underline{b}_{3g}^{T}} \qquad (C.12)$$

where NG(3) is the total number of grid points or nodes in the link 3. For example if the link 3 of the robot is discritized into 5 finite elements, then the NG(3) will be 6. The Eqn. (C.12) is further expanded for one particular value of g = 2. We get $m_{32}\ \underline{\Phi}_{321}\ \underline{b}_{32}^{T}$ where m_{32} is the lumped mass of the second node in the third link. From Eqn. (3.21) we get

$$\underline{\Phi}_{321} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \qquad (C.13)$$

and $\underline{b}_{32}$ is the position vector of the second node in the link 3, which is determined after knowing the input angles θ_1, θ_2 and θ_3 for the robot.

In this way all the quantities in the right hand side of the Eqn. (C.12) are calculated and added to get $[J_{31}]$. Then $[E_3]$ and $[J_{31}]$ are substituted in the Eqn. (C.6) and $(f_{31})_2$ is calculated. Similarly all other terms in Eqn. (C.5) are calculated and the velocity force is obtained.

The calculation of acceleration force is as follows:

From Eqn. (3.47) we have

$$(f_{i\alpha})_3 = [F_i] \odot [J_{i\alpha}] \tag{C.14}$$

where

$$[F_i] = \sum_{j=1}^{NL} [C_{ij}] \ddot{\theta}_j \tag{C.15}$$

$$[C_{ij}] = [T_o^i]^T \odot [U_{ij}] \tag{C.16}$$

$[U_{ij}]$ is given in Eqn. (3.23) and $[J_{i\alpha}]$ was already derived in Eqn. (C.5).

Taking the same node as before we have

$$(f_{31})_3 = [F_3] \odot [J_{31}] \tag{C.17}$$

$$[F_3] = \sum_{j=1}^{3} C_{3j}\ \ddot{\theta}_j \qquad (C.18)$$

$$= C_{31}\ \ddot{\theta}_1 + C_{32}\ \ddot{\theta}_2 + C_{33}\ \ddot{\theta}_3 \qquad (C.19)$$

One of the terms in Eqn. (C.19) is expaned further

$$[C_{32}] = [T_o^{\,3}]^T[U_{32}] \qquad (C.20)$$

$$= [T_o^{\,3}]^T[T_o^{\,1}][Q][T_1^{\,3}] \qquad (C.21)$$

Since all the quantities in the right hand side of the Eqn. (C.20) are known, $[C_{32}]$ is calculated. Similarly all terms in Eqn. (C.19) are calculated. The angular accelerations $\ddot{\theta}_1$, $\ddot{\theta}_2$ and $\ddot{\theta}_3$ are known from Section (2.5) and hence $[F_3]$ in the Eqn. (C.18) is calculated. Since $[J_{31}]$ is already known from Eqn. (C.12), $(F_{31})_3$ in Eqn. (C.17) is calculated. Similarly all other terms in Eqn. (C.14) are calculated. Thus the acceleration force is obtained.

The calculation of gravity force in Eqn. (C.1) is as follows:

Writing Eqn. (3.31) using the dot product notation we have

$$(f_{i\alpha})_4 = [T_o^{\,i}]^T \odot [M_{i\alpha}] \qquad (C.22)$$

where $[M_{i\alpha}]$ is given in Eqn. (3.33). Considering the same node as before, the gravity force is written as

$$(f_{31})_4 = [T_o^{\,3}]^T \odot [M_{31}] \qquad (C.23)$$

where

$$[M_{31}] = \sum_{g=1}^{NG(3)} m_{3g}\, \Phi_{3g1}\, \underline{g}^T \qquad (C.24)$$

m_{3g} and Φ_{3g1} were derived in the Eqn. (C.12) and

$$\underline{g} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -9.81 \end{bmatrix} \text{ m/sec}^2 \qquad (C.25)$$

Thus $[M_{31}]$ in the Eqn. (C.24) is calculated. By substituting

$[M_{31}]$ in Eqn. (C.23) one can get $(f_{31})_4$. Similarly all the terms in the gravity force in the Eqn. (C.22) can be calculated.

Since all the quantities in the right hand side of Eqn. (C.1) are known, the total force vector $(f_{i\alpha})$ can be calculated. As a matter of clarification, it must be noted that the total force (f_{31}) represents the first element in the total force vector for the link 3.

APPENDIX D

DESCRIPTION AND LISTING OF THE COMPUTER PROGRAMS

The required computations were done on the VAX 11/780 digital computer, using a package of programs developed in FORTRAN language. All these programs are given in this Appendix. They are self-sufficient and can be run with the required input data files. Many IMSL subroutines are used for matrix multiplication, inversion, transpose and while finding the eigenvalues and vectors. A brief description about each program is given here.

The program NEWTON solves the inverse kinematics problem. Given the position vector of the end effector, this program calculates the required input angles using modified Newton-Raphson technique. The same program also calculates the angular velocities and accelerations of the links for a specified velocity and acceleration of the end effector.

The program KINEM uses the complex Optimization Technique to solve the inverse kinematics problem. This program is capable of finding the best solution among the possible multiple solutions. If there are any constraints on the motion of the robot links, they can also be easily incorporated.

The program VEL finds the input velocities and accelerations of the links for a given velocity and acceleration of the end effector. This program simultaneously solves three equations to find the three unknown angular velocities and accelerations.

The program FIN is written to find the natural frequency of the robotic manipulator using three-dimensional finite element analysis. The robot was discretized into beam elements with six degrees of freedom at each node. Lumped mass approach was used to find the elemental mass and stiffness matrices. Then they are pre and post multiplied with transformation matrix and then assembled into the global system.

The program STRUDL finds the natural frequency of the robotic manipulator using a standard finite element analysis package GTSTRUDL. The main purpose of this program is to verify the results obtained from the program FIN. In order to run this program, it must be linked with GTSTRUDL package.

The program DYNCON is used to condense the system matrices obtained from the program FIN. Guyan's reduction technique is used to reduce the size of the dynamic matrices.

The program COMCON serves the same purpose as DYNCON except the fact that component mode synthesis technique is used for condensation.

```
C*************************************************************
C*                                                           *
C*      LISTING OF THE PROGRAM ++NEWTON++                    *
C*      PROGRAM TO DO THE DISPLACEMENT ANALYSIS VELOCITY     *
C*      ANALYSIS AND ACCELERATION ANALYSIS USING             *
C*      MODIFIED NEWTON RAPHSON TECHNIQUE                    *
C*                                                           *
C*************************************************************
      REAL T1(4,4),T2(4,4),T3(4,4),L1,TEMP1(4,4),
     1 RSP1(4),RAP1(4),Q1(4,4),Q2(4,4),A(3,3),D1(4,4),
     1 TEMP3(4,4),TEMP4(4,4),TEMP5(4,4),TEMP6(4),
     1 TEMP8(4),TEM1(3,3),TEM2(3,3),DEL(3),WKAREA(1000),
     1 TEMP10(4,4),TEMP11(4,4),TEMP12(4,4),TEM3(3,3),
     1 WWV(4,4),D(3,4,4),ANACCL(3),ALACCL(3),TEM4(3),
     1 TEMP2(4,4),R1(4),D2(4,4),D3(4,4),ANVEL(3),
     1 TEMP9(4,4),INC(3),ALVEL(3),TEMP7(4)
      L1=1.016
      THETA1=.1243549
      THETA2=1.002441
      THETA3=-1.24042
C*************************************************************
C*      R1 IS THE SPECIFIED POINT IN  LOCAL COORDINATE       *
C*      SYSTEM RSP1 IS THE SPECIFIED POINT IN GLOBAL         *
C*      COORDINATE SYSTEM                                    *
C*************************************************************
      R1(1)=1.
      R1(2)=1.5113
      RSP1(1)=1.
      RSP1(2)=2.
      RSP1(3)=0.25
      RSP1(4)=-0.5
C*************************************************************
C     HERE THE FIRST TRANSFORMATION MATRIX IS FORMED          *
C*************************************************************
100   T1(1,1)=1.
      T1(2,2)=COS(THETA1)
      T1(2,3)=-SIN(THETA1)
      T1(3,2)=SIN(THETA1)
      T1(3,3)=COS(THETA1)
      T1(4,4)=1.
C*************************************************************
C     HERE THE SECOND TRANSFORMATION MATRIX IS FORMED         *
C*************************************************************
      T2(1,1)=1.
      T2(2,2)=COS(THETA2)
      T2(2,4)=SIN(THETA2)
      T2(3,3)=1.
      T2(4,2)=-SIN(THETA2)
      T2(4,4)=COS(THETA2)
C*************************************************************
C     HERE THE THIRD TRANSFORMATION MATRIX IS FORMED          *
C*************************************************************
```

```
170      A(I+NI,J+NI)=REIG(I,J)
         CALL VMULFM(A,GKK,NN,NO,NN,160,160,TEMP4,160,IER)
         CALL VMULFF(TEMP4,A,NO,NN,NO,160,160,RK,160,IER)
         CALL VMULFM(A,GMM,NN,NO,NN,160,160,TEMP5,160,IER)
         CALL VMULFF(TEMP5,A,NO,NN,NO,160,160,RM,160,IER)
         IDGT =4
         CALL LINV2F (RK,NO,160,RKINV,IDGT,WKAREA,IER)
         CALL VMULFF (RKINV,RM,NO,NO,NO,160,160,LAMDA,
     1   160,IERR)
         DO 180 I=1,NO
         DO 180 J=1,NO
180      ILAMDA(I,J)=CMPLX(LAMDA(I,J))
C*****************************************************
C        DETERMINATION OF EIGEN VALUES                *
C*****************************************************
         IJOB =1
         CALL EIGCC(ILAMDA,NO,160,IJOB,EIGVAL,EIGVEC,160,
     1   WK,IERROR)
         DO 190 I=1,NO
190      EIGVAL(I)=1./EIGVAL(I)
         PRINT *, 'IERROR=',IERROR,'EIGEN VALUES ARE='
         PRINT 200,(EIGVAL(I),I=1,NO)
200      FORMAT (2X,12E10.4)
         PRINT *,'THE FREQUENCIES ARE'
         DO 210 I=1,NO
         OMEGA(I)=SQRT(REAL(EIGVAL(I)))
         PRINT *,'I=',I,'FREQUENCY=',OMEGA(I)
210      CONTINUE
230      FORMAT (2(E12.4,4X))
         STOP
         END
```

```
      T3(1,1)=1.
      T3(2,1)=L1
      T3(2,2)=COS(THETA3)
      T3(2,4)=SIN(THETA3)
      T3(3,3)=1.
      T3(4,2)=-SIN(THETA3)
      T3(4,4)=COS(THETA3)
C*****************************************************************
C     APPROXIMATE POSITION VECTORS ARE FOUND HERE                *
C*****************************************************************
      CALL VMULFF(T1,T2,4,4,4,4,4,TEMP1,4,IER)
      CALL VMULFF(TEMP1,T3,4,4,4,4,4,TEMP2,4,IER)
      CALL VMULFF(TEMP2,R1,4,4,1,4,4,RAP1,4,IER)
      PRINT*,'APPROXIMATE VECTORS ARE',(RAP1(I),I=1,4)
C*****************************************************************
C     D MATRICES ARE FORMED HERE.                                *
C*****************************************************************
      Q1(2,3)=-1.
      Q1(3,2)=1.
      Q2(2,4)=1.
      Q2(4,2)=-1.
      CALL VMULFF(T1,Q1,4,4,4,4,4,TEMP3,4,IER)
      CALL LINV2F(T1,4,4,TEMP9,4,WKAREA,IER)
      CALL VMULFF(TEMP3,TEMP9,4,4,4,4,4,D1,4,IER)
      CALL VMULFF(TEMP1,Q2,4,4,4,4,4,TEMP5,4,IER)
      CALL LINV2F(TEMP1,4,4,TEMP10,4,WKAREA,IER)
      CALL VMULFF(TEMP5,TEMP10,4,4,4,4,4,D2,4,IER)
      CALL VMULFF(TEMP2,Q2,4,4,4,4,4,TEMP11,4,IER)
      CALL LINV2F(TEMP2,4,4,TEMP12,4,WKAREA,IER)
      CALL VMULFF(TEMP11,TEMP12,4,4,4,4,4,D3,4,IER)
C*****************************************************************
C     A MATRICES ARE FORMED HERE.                                *
C*****************************************************************
      CALL VMULFF(D1,RAP1,4,4,1,4,4,TEMP6,4,IER)
      CALL VMULFF(D2,RAP1,4,4,1,4,4,TEMP7,4,IER)
      CALL VMULFF(D3,RAP1,4,4,1,4,4,TEMP8,4,IER)
      DO 10 I=1,3
      A(I,1)=TEMP6(I+1)
      A(I,2)=TEMP7(I+1)
10    A(I,3)=TEMP8(I+1)
C*****************************************************************
C     THE INCREMENT IN ANGLES ARE CALCULATED HERE                *
C*****************************************************************
      CALL VMULFM(A,A,3,3,3,3,3,TEM1,3,IER)
      CALL LINV2F(TEM1,3,3,TEM2,4,WKAREA,IER)
      CALL VMULFP(TEM2,A,3,3,3,3,3,TEM3,3,IER)
      DO 20 I=1,3
20    DEL(I)=RSP1(I+1)-RAP1(I+1)
      CALL VMULFF(TEM3,DEL,3,3,1,3,3,INC,3,IER)
      THETA1=THETA1+INC(1)
      THETA2=THETA2+INC(2)
      THETA3=THETA3+INC(3)
```

```
      DO 30 I =2,4
      IF((ABS(RSP1(I)-RAP1(I))).GT.(1.E-5)) GO TO 100
30    CONTINUE
      PRINT*,' FINAL THETA VALUES ARE'
      PRINT*,THETA1,THETA2,THETA3
      PRINT*,' THE SPECIFIED POINTS ARE'
      PRINT*,(RAP1(I),I=1,4)
C*****************************************************************
C     HERE THE VELOCITY ANALYSIS STARTS
C*****************************************************************
      ALVEL(1)=0.733235
      ALVEL(2)=0.733235
      ALVEL(3)=-0.733235
      CALL VMULFF(TEM3,ALVEL,3,3,1,3,3,ANVEL,3,IER)
      PRINT*,'THE ANGULAR VELOCITIES',ANVEL
C*****************************************************************
C     HERE THE ACCELERATION ANALYSIS STARTS
C*****************************************************************
      ALACCL(1)=2.93294
      ALACCL(2)=2.93294
      ALACCL(3)=-2.93294
      DO 40 I=1,4
      DO 40 J=1,4
      D(1,I,J)=D1(I,J)
      D(2,I,J)=D2(I,J)
40    D(3,I,J)=D3(I,J)
      DO 50 M=1,3
      DO 50 J=1,3
      DO 60 I=1,4
      DO 60 K=1,4
      TEMP1(I,K)=D(J,I,K)
60    TEMP2(I,K)=D(M,I,K)
      IF (J.GT.M) CALL VMULFF(TEMP2,TEMP1,4,4,4,4,4,TEMP3,
     1 4,IER)
      IF (J.LE.M) CALL VMULFF(TEMP1,TEMP2,4,4,4,4,4,TEMP3,
     1 4,IER)
      DO 70 I=1,4
      DO 70 K=1,4
70    WWV(I,K)=WWV(I,K)+TEMP3(I,K)*ANVEL(J)*ANVEL(M)
50    CONTINUE
C     RSP1(4)=-RSP1(4)
      CALL VMULFF(WWV,RSP1,4,4,1,4,4,TEMP6,4,IER)
      PRINT*,'IER',IER,'WWV*RV',TEMP6
      DO 80 I=1,3
80    TEM4(I)=ALACCL(I)-TEMP6(I+1)
      CALL VMULFF(TEM3,TEM4,3,3,1,3,3,ANACCL,3,IER)
      PRINT*,'ANGULAR ACCELERATIONS',ANACCL
      STOP
      END
```

```
C*****************************************************************
C*                                                               *
C*     LISTING OF THE PROGRAM ++KINEM**                          *
C*     PROGRAM TO DO THE DISPLACEMENT ANALYSIS USING             *
C*     COMPLEX OPTIMISATION FOR ROBOTIC MANIPULATORS.            *
C*                                                               *
C*****************************************************************
      DIMENSION X(6,3),R(6,3),F(6),G(3),H(3),XC(3)
      INTEGER GAMMA
      DOUBLE PRECISION DSEED
      DSEED=123.D0
      NI=50
      NO=66
      N=3
      M=3
      K=6
      ITMAX=500
      IC=0
      IPRINT=0
      ALPHA=1.3
      BETA=.1
      GAMMA=5
      DELTA=.0001
      X(1,1)=0.12435
      X(1,2)=.92378
      X(1,3)=-1.30242
999   DSEED=DSEED+39.D0
      DO 10 II=2,K
      DO 10 J=1,N
10    R(II,J)=GGUBFS(DSEED)
      CALL CONSX(N,M,K,ITMAX,ALPHA,BETA,GAMMA,DELTA,X,
     1 R,F,IT,IEV2,NO,G,H,XC,IPRINT)
      IF(F(IEV2).EQ.TEMP) GO TO 13
      IF(IT-ITMAX) 20,20,30
30    PRINT *,' MAX ITERATIONS EXCEEDED',IT
20    PRINT *,' FINAL VALUE OF FUNCTION',F(IEV2)
      PRINT *,' ITERATION NO. IS',IT
      PRINT *,' THE FINAL X VALUES ARE'
      PRINT *, (X(IEV2,J),J=1,N)
13    TEMP=F(IEV2)
      IF (F(IEV2).LT.-.000005) THEN
      DO 11 I=1,N
11    X(1,I)=X(IEV2,I)
      GO TO 999
      END IF
      STOP
      END


      SUBROUTINE CONSX(N,M,K,ITMAX,ALPHA,BETA,GAMMA,
     1 DELTA,X,R,F,IT,IEV2,NO,G,H,XC,IPRINT)
```

```
      DIMENSION X(K,M),R(K,N),F(K),G(M),H(M),XC(N)
      INTEGER GAMMA
      IT=1
      KODE=0
      IF (M-N) 20,20,10
10    KODE=1
20    CONTINUE
      DO 40 II=2,K
      DO 30 J=1,N
30    X(II,J)=0.
40    CONTINUE
      DO 65 II=2,K
      DO 50 J=1,N
      I=II
      CALL CONST(N,M,K,X,G,H,I)
      X(II,J)=G(J)+R(II,J)*(H(J)-G(J))
50    CONTINUE
      K1=II
      CALL CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
      IF (II-2) 51,51,55
51    IF(IPRINT)52,65,52
52    PRINT*,' THE COORDINATES OF INITIAL COMPLEX ARE'
      IO=1
      PRINT*,(IO,J,X(IO,J),J=1,N)
55    IF (IPRINT) 56,65,56
56    PRINT*,(II,J,X(II,J),J=1,N)
65    CONTINUE
      K1=K
      DO 70 I=1,K
      CALL FUNC(N,M,K,X,F,I)
70    CONTINUE
      KOUNT=1
      IA=0
      IF (IPRINT) 72,80,72
72    PRINT*,' THE VALUES OF FUNCTION',(J,F(J),J=1,K)
80    IEV1=1
      DO 100 ICM=2,K
      IF(F(IEV1)-F(ICM))100,100,90
90    IEV1=ICM
100   CONTINUE
      IEV2=1
      DO 120 ICM=2,K
      IF (F(IEV2)-F(ICM)) 110,110,120
110   IEV2=ICM
120   CONTINUE
      IF (F(IEV2)-(F(IEV1)+BETA)) 140,130,130
130   KOUNT=1
      GO TO 150
140   KOUNT=KOUNT+1
      IF(KOUNT-GAMMA) 150,240,240
150   CALL CENTR(N,M,K,IEV1,I,XC,X,K1)
      DO 160 JJ=1,N
```

```
160     X(IEV1,JJ)=(1.+ALPHA)*(XC(JJ))-ALPHA*(X(IEV1,JJ))
        I=IEV1
        CALL CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
        CALL FUNC(N,M,K,X,F,I)
170     IEV2=1
        DO 190 ICM=2,K
        IF (F(IEV2)-F(ICM)) 190,190,180
180     IEV2=ICM
190     CONTINUE
        IF (IEV2-IEV1)220,200,220
200     DO 210 JJ=1,N
        X(IEV1,JJ)=(X(IEV1,JJ)+XC(JJ))/2.
210     CONTINUE
        I=IEV1
        CALL CHECK (N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
        CALL FUNC (N,M,K,X,F,I)
        GO TO 170
220     CONTINUE
        EIT=IT
        IF (IPRINT) 230,228,230
230     PRINT*,' ITERATION NO.',IT
        PRINT*,' THE COORDINATES OF CORRECTED POINT'
        PRINT*,(IEV1,JC,X(IEV1,JC),JC=1,N)
        PRINT *,' THE VALUES OF THE FUNCTION'
        PRINT*,(I,F(I),I=1,K)
        PRINT*,(JC,XC(JC),JC=1,N)
228     IT=IT+1
        IF(IT-ITMAX) 80,80,240
240     RETURN
        END


        SUBROUTINE CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
        DIMENSION X(K,M),G(M),H(M),XC(N)
10      KT=0
        CALL CONST(N,M,K,X,G,H,I)
        DO 50 J=1,N
        IF(X(I,J)-G(J))20,20,30
20      X(I,J)=G(J)+DELTA
        GO TO 50
30      IF (H(J)-X(I,J)) 40,40,50
40      X(I,J)=H(J)-DELTA
50      CONTINUE
        IF (KODE) 110,110,60
60      NN=N+1
        DO 100 J=NN,M
        CALL CONST (N,M,K,X,G,H,I)
        IF (X(I,J)-G(J)) 80,70,70
70      IF(H(J)-X(I,J))80,100,100
80      IEV1=I
        KT=1
        CALL CENTR(N,M,K,IEV1,I,XC,X,K1)
```

```
      DO 90 JJ=1,N
      X(I,JJ)=(X(I,JJ)+XC(JJ))/2.
90    CONTINUE
100   CONTINUE
      IF(KT)110,110,10
110   RETURN
      END


      SUBROUTINE CENTR(N,M,K,IEV1,I,XC,X,K1)
      DIMENSION X(K,M),XC(N)
      DO 20 J=1,N
      XC(J)=0.
      DO 10 IL=1,K1
10    XC(J)=XC(J)+X(IL,J)
      RK=K1
20    XC(J)=(XC(J)-X(IEV1,J))/(RK-1.)
      RETURN
      END


      SUBROUTINE FUNC(N,M,K,X,F,I)
      DIMENSION X(K,M),F(K),QX(9),QY(9),PX(9),PY(9)
     1 Y(2),RR1(4),RR2(4),RMAX(3),RMIN(3),RS1(4),RS2(4)
      RS1(2)=2.
      RS1(3)=0.25
      RS1(4)=0.5
      F(I)=0.
      CALL MAIN(I,K,M,X,RR1)
      F(I)=(RS1(2)-RR1(2))**2+(RS1(3)-RR1(3))**2+
     1 (RS1(4)-RR1(4))**2
      F(I)=-F(I)*1.E10+(X(I,1)-.124355)**2
     1 +(X(I,2)-.9424094)**2+(X(I,3)+1.295016)**2
      RETURN
      END


      SUBROUTINE CONST(N,M,K,X,G,H,I)
      DIMENSION X(K,M),G(M),H(M)
      G(1)=-1.
      H(1)=1.
      G(2)=0.0
      H(2)=1.5
      G(3)=-3.
      H(3)=0.0
      RETURN
      END


      SUBROUTINE MAIN(I,K,M,X,RR1)
```

```
      REAL L(3),THETA(3),TO1(4,4),TO2(4,4),TO3(4,4),
     1 R1(4),R2(4),RR1(4),RR2(4),D1(4,4),D2(4,4),D3(4,4),
     1 TEMP1(4,4),TEMP2(4,4),RS1(4),RS2(4),TEMP3(4,4),
     1 TEMP5(4,4),TEMP6(4,4),RES(8),DEL(3),TEM1(3,3),
     1 TEM3(3,8),A(8,3),Q(4,4),WKAREA(1000),X(K,M),
     1 T12(4,4),TEMP(4,4),TEMP4(4,4),TEM2(3,3),T23(4,4)
      L(1)=0.
      L(2)=1.016
      L(3)=1.5113
      ALFA=3.141592/2.
      THETA(1)=X(I,1)
      THETA(2)=X(I,2)
      THETA(3)=X(I,3)
100   CALL TRANS(L(1),THETA(1),ALFA,TO1)
      CALL TRANS(L(2),THETA(2),0.,T12)
      CALL TRANS(L(3),THETA(3),0.,T23)
      CALL VMULFF(TO1,T12,4,4,4,4,4,TO2,4,IER)
      CALL VMULFF(TO2,T23,4,4,4,4,4,TO3,4,IER)
      R1(1)=1.
      R1(2)=0.
      CALL VMULFF(TO3,R1,4,4,1,4,4,RR1,4,IER)
5     FORMAT(5X,4E10.3)
      RETURN
      END


      SUBROUTINE TRANS(L,THETA,ALFA,A)
      REAL L,THETA,A(4,4)
      CT=COS(THETA)
      ST=SIN(THETA)
      CA=COS(ALFA)
      SA=SIN(ALFA)
      A(1,1)=1.
      A(2,1)=L*CT
      A(2,2)=CT
      A(2,3)=-ST*CA
      A(2,4)=ST*SA
      A(3,1)=L*ST
      A(3,2)=ST
      A(3,3)=CT*CA
      A(3,4)=-CT*SA
      A(4,3)=SA
      A(4,4)=CA
      RETURN
      END
```

```
C*****************************************************************
C*                                                               *
C*      LISTING OF THE PROGRAM ++VEL++                           *
C*      PROGRAM TO CALCULATE VELOCITY AND ACCELERATION           *
C*      OF OPEN LOOP SPATIAL MECHANISMS BY SOLVING               *
C*      SYMULTANEOUS EQUATIONS.                                  *
C*                                                               *
C*****************************************************************

C*****************************************************************
C*      NOTATION FOLLOWED                                        *
C*      T1=THETA1, T2=THETA2, T3=THETA3, SI=T2+T3                *
C*****************************************************************
      DIMENSION AJACOB(4,4),ALVEL(4),WKAREA(100),
     1, TEMP(4,4),ANVEL(4),DERJAC(4,4),ANACCL(4),ALACCL(4)
      A2=1.016
      A3=1.5113
      T1=0.
      T2=0.8281533
      T3=-1.346276
      SI=T2+T3
      A1X=-(A2*COS(T2)*SIN(T1)+A3*COS(SI)*SIN(T1))
      A2X=-(A2*SIN(T2)*COS(T1)+A3*SIN(SI)*COS(T1))
      A3X=-(A3*SIN(SI)*COS(T1))
      A1Y=(A2*COS(T2)*COS(T1)+A3*COS(SI)*COS(T1))
      A2Y=-(A2*SIN(T2)*SIN(T1)+A3*SIN(SI)*SIN(T1))
      A3Y=-(A3*SIN(SI)*SIN(T1))
      A1Z=0.
      A2Z=A2*COS(T2)+A3*COS(SI)
      A3Z=A3*COS(SI)
      AJACOB(1,1)=1.
      AJACOB(2,2)=A1X
      AJACOB(2,3)=A2X
      AJACOB(2,4)=A3X
      AJACOB(3,2)=A1Y
      AJACOB(3,3)=A2Y
      AJACOB(3,4)=A3Y
      AJACOB(4,2)=A1Z
      AJACOB(4,3)=A2Z
      AJACOB(4,4)=A3Z
      ALVEL(1)=1.
      ALVEL(2)=0.733235
      ALVEL(3)=0.733235
      ALVEL(4)=0.733235
      CALL LINV2F(AJACOB,4,4,TEMP,4,WKAREA,IER)
      CALL VMULFF(TEMP,ALVEL,4,4,1,4,4,ANVEL,4,IER)
      PRINT*,'THE VELOCITIES',ANVEL
C*****************************************************************
C     HERE THE CALCULATIONS FOR ACCELERATIONS STARTS             *
C*****************************************************************
      VEL1=ANVEL(2)
      VEL2=ANVEL(3)
```

```
      VEL3=ANVEL(4)
      A4X=2.*(A2*SIN(T1)*SIN(T2)+A3*SIN(SI)*SIN(T1))*
     1  VEL1*VEL2-2*(A3*COS(SI)*COS(T1))*VEL2*VEL3
     1  +2*(A3*SIN(T1)*SIN(SI))*VEL1*VEL3
     1  -(A2*COS(T1)*COS(T2)+A3*COS(SI)*COS(T1))*VEL1**2
     1  -(A2*COS(T1)*COS(T2)+A3*COS(SI)*COS(T1))*VEL2**2
     1  -(A3*COS(SI)*COS(T1))*VEL3**2
      A4Y=-(A2*COS(T2)*SIN(T1)+A3*COS(SI)*SIN(T1))*
     1  VEL1**2-(A2*COS(T2)*SIN(T1)+A3*COS(SI)*SIN(T1))
     1  *VEL2**2-(A3*COS(SI)*SIN(T1))*VEL3**2
     1  -2.*(A2*SIN(T2)*COS(T1)+A3*SIN(SI)*COS(T1))*VEL1
     1  *VEL2-2.*(A3*COS(SI)*SIN(T1))*VEL2*VEL3
     1  -2.*(A3*SIN(SI)*COS(T1))*VEL1*VEL3
      A4Z=-(A2*SIN(T2)+A3*SIN(SI))*VEL2**2
     1  -2.*(A3*SIN(SI))*VEL2*VEL3
     1  -A3*SIN(SI)*VEL3**2
      DO 10 I=2,4
      DO 10 J=2,4
10    DERJAC(I,J)=AJACOB(I,J)
      DERJAC(1,1)=1.
      DERJAC(2,1)=A4X
      DERJAC(3,1)=A4Y
      DERJAC(4,1)=A4Z
      ALACCL(1)=1.
      ALACCL(2)=2.93294
      ALACCL(3)=2.93294
      ALACCL(4)=2.93294
      CALL LINV2F(DERJAC,4,4,TEMP,4,WKAREA,IER)
      CALL VMULFF(TEMP,ALACCL,4,4,1,4,4,ANACCL,4,IER)
      PRINT*,'ANGULAR ACCL ARE',ANACCL
      STOP
      END
```

```
C*****************************************************************
C*                                                               *
C*      LISTING OF THE PROGRAM ++FIN++                           *
C*      PROGRAM TO CALCULATE THE NATURAL FREQUENCIES OF          *
C*      ROBOTIC MANIPULATORS USING FINITE ELEMENT                *
C*      ANALYSIS                                                 *
C*                                                               *
C*****************************************************************
      DIMENSION A1(3),B1(3),C1(3),D1(3),GM(200,200),
     1 GM1(200,200),GK1(200,200),GM2(200,200),
     1 GM3(200,200),GK3(200,200),GM23(200,200),
     1 THETA(3),GK(200,200),GK2(200,200),GK23(200,200)
      COMMON/BLK1/OMEGA(200)
      OPEN (UNIT=14,FILE='MAS.DAT',TYPE='NEW')
      OPEN (UNIT=15,FILE='STIF.DAT',TYPE='NEW')
      THETA(1)=0.1243549
      THETA(2)=1.002441
      THETA(3)=-1.24042
      CALL DIMEN(THETA,B1,C1,D1)
      N1=10
      N2=1
      N3=15
      CALL BEAM(A1,B1,N1,2.E11,.01016,7800.,6.14591E-5,
     1 3.07295E-5,3.07295E-5,GM1,GK1)
      CALL BEAM(C1,B1,N2,6.894E10,0.00798,2700.,3.62E-5,
     1 1.81E-5,1.81E-5,GM2,GK2)
      CALL BEAM(B1,D1,N3,6.894E10,0.00798,2700.,3.62E-5,
     1 1.81E-5,1.81E-5,GM3,GK3)
      CALL ASSEM1(N2,GM2,GK2,N3,GM3,GK3,GM23,GK23)
      N23=(N2+N3)*6+6
      CALL ASSEM(N1,GM1,GK1,N23,GM23,GK23,GM,GK)
      N=(N1+N2+N3)*6+6
      CALL FREQ(GM,GK,N)
      PRINT*,' THE NATURAL FREQUENCIES ARE'
      DO 14 I=1,6
      PRINT*, 'I=',I,'FREQUENCY=',OMEGA(I)/2./3.14159
14    CONTINUE
      STOP
      END


      SUBROUTINE BEAM(A1,B1,N,E,A,DEN,AIX,AIY,AIZ,GM,GK)
      DIMENSION AK(12,12),AM(12,12),T(12,12),A1(3),B1(3),
     1 TEMP1(12,12),GK(200,200),GM(200,200),EK(12,12),
     1 AMM(25,12,12),AKK(25,12,12),EM(12,12)
      DO 5 I=1,200
      DO 5 J=1,200
      GM(I,J)=0.
5     GK(I,J)=0.
      AMAS=DEN*A
      ANEW=0.3
```

```
      G=E/(2.*(1.+ANEW))
      TOTLEN=SQRT((B1(1)-A1(1))**2+(B1(2)-A1(2))**2+
     1  (B1(3)-A1(3))**2)
      ALEN=TOTLEN/N
      CALL STIFF(ALEN,E,G,A,AIX,AIY,AIZ,AK)
      CALL LUMPMAS(ALEN,A,DEN,AM)
      CALL TRAN(A1,B1,T)
C*****************************************************************
C     ELEMENT MASS AND STIFFNESS MATRICES ARE                    *
C     MULTIPLIED BY TRANSFORMATION MATRIX                        *
C*****************************************************************
      CALL VMULFM(T,AK,12,12,12,12,12,TEMP1,12,IER)
      CALL VMULFF(TEMP1,T,12,12,12,12,12,EK,12,IER)
      CALL VMULFM(T,AM,12,12,12,12,12,TEMP1,12,IER)
      CALL VMULFF(TEMP1,T,12,12,12,12,12,EM,12,IER)
      DO 10 I=1,N
      DO 10 J=1,12
      DO 10 K=1,12
      AMM(I,J,K)=EM(J,K)
10    AKK(I,J,K)=EK(J,K)
C*****************************************************************
C     HERE THE ELEMENT MATRICES ARE ASSEMBLED                    *
C*****************************************************************
      I3=0
      DO 20 I=1,N*6-5,6
      I3=I3+1
      I1=I-1
      DO 20 I2=1,12
      DO 20 J2=1,12
      GM(I1+I2,I1+J2)=GM(I1+I2,I1+J2)+AMM(I3,I2,J2)
20    GK(I1+I2,I1+J2)=GK(I1+I2,I1+J2)+AKK(I3,I2,J2)
      RETURN
      END


      SUBROUTINE FREQ(GM,GK,N)
      DIMENSION GMM(200,200),GKK(200,200),GM(200,200),
     1  WKAREA(2000),ALAMDA(200,200),GKKINV(200,200),
     1  GK(200,200)
      COMPLEX  AILAMDA(200,200),EIGVAL(200),
     1  EIGVEC(200,200)
      COMMON/BLK1/OMEGA(200)
C*****************************************************************
C     HERE THE  BOUNDARY CONDITIONS ARE APPLIED                  *
C*****************************************************************
      DO 30 I=1,N-6
      DO 30 J=1,N-6
      GMM(I,J)=GM(I+6,J+6)
30    GKK(I,J)=GK(I+6,J+6)
      WRITE(14,*)((GMM(I,J),J=1,N-6),I=1,N-6)
      WRITE(15,*)((GKK(I,J),J=1,N-6),I=1,N-6)
```

```
C*************************************************************
C       HERE THE NATURAL FREQUENCIES ARE FOUND                *
C*************************************************************
        NN=N-6
        IDGT=4
        CALL LINV2F(GKK,NN,200,GKKINV,IDGT,WKAREA,IER)
        CALL VMULFF(GKKINV,GMM,NN,NN,NN,200,200,ALAMDA,200,
     1  IER)
        DO 40 I=1,NN
        DO 40 J=1,NN
40      AILAMDA(I,J)=CMPLX(ALAMDA(I,J))
C*************************************************************
C       DETERMINATION OF EIGEN VALUES                         *
C*************************************************************
        IJOB=1
        CALL EIGCC(AILAMDA,NN,200,IJOB,EIGVAL,EIGVEC,200,
     1  WKAREA,IER)
        DO 50 I=1,NN
50      EIGVAL(I)=1./EIGVAL(I)
        DO 70 I=1,NN
        OMEGA(I)=SQRT(REAL(EIGVAL(I)))
70      CONTINUE
        DO 80 I=1,NN
        DO 80 J=I+1,NN
        IF (OMEGA(I) .GT.OMEGA(J)) THEN
        TEMP=OMEGA(I)
        OMEGA(I)=OMEGA(J)
        OMEGA(J)=TEMP
        END IF
80      CONTINUE
        RETURN
        END


        SUBROUTINE STIFF(AL,E,G,A,AIX,AIY,AIZ,AK)
        DIMENSION AK(12,12)
        AK(1,1)=E*A/AL
        AK(2,2)=12.*E*AIZ/AL**3
        AK(3,3)=12.*E*AIY/AL**3
        AK(4,4)=G*AIX/AL
        AK(5,3)=-6.*E*AIY/AL**2
        AK(5,5)=4.*E*AIY/AL
        AK(6,2)=6.*E*AIZ/AL**2
        AK(6,6)=4.*E*AIZ/AL
        AK(7,1)=-E*A/AL
        AK(7,7)=A*E/AL
        AK(8,2)=-12.*E*AIZ/AL**3
        AK(8,6)=-6.*E*AIZ/AL**2
        AK(8,8)=12.*E*AIZ/AL**3
        AK(9,3)=-12.*E*AIY/AL**3
        AK(9,5)=6.*E*AIY/AL**2
```

```
      AK(9,9)=12.*E*AIY/AL**3
      AK(10,4)=-G*AIX/AL
      AK(10,10)=G*AIX/AL
      AK(11,3)=-6.*E*AIY/AL**2
      AK(11,5)=2.*E*AIY/AL
      AK(11,9)=6.*E*AIY/AL**2
      AK(11,11)=4.*E*AIY/AL
      AK(12,2)=6.*E*AIZ/AL**2
      AK(12,6)=2.*E*AIZ/AL
      AK(12,8)=-6.*E*AIZ/AL**2
      AK(12,12)=4.*E*AIZ/AL
      DO 10 I=1,11
      DO 10 J=I+1,12
10    AK(I,J)=AK(J,I)
      RETURN
      END


      SUBROUTINE LUMPMAS(AL,A,DEN,AM)
      DIMENSION AM(12,12)
      AMASS=AL*A*DEN/2.
      DO 10 I=1,3
      AM(I,I)=AMASS
10    AM(I+6,I+6)=AMASS
      DO 20 I=4,6
      AM(I,I)=AMASS/10000.
20    AM(I+6,I+6)=AMASS/10000.
      RETURN
      END


      SUBROUTINE TRAN(A,B,FT)
      DIMENSION A(3),B(3),FT(12,12)
      AX=A(1)
      AY=A(2)
      AZ=A(3)
      BX=B(1)
      BY=B(2)
      BZ=B(3)
      PX=0.
      PY=10.
      PZ=0.
      ABX=BX-AX
      ABY=BY-AY
      ABZ=BZ-AZ
      ABLEN=SQRT(ABX*ABX + ABY*ABY + ABZ*ABZ)
      CX=ABX/ABLEN
      CY=ABY/ABLEN
      CZ=ABZ/ABLEN
      CONS=SQRT(CX*CX+CZ*CZ)
```

```
      XP=CX*PX + CY*PY + CZ*PZ
      YP=-CX*CY*PX/CONS + CONS*PY - CY*CZ*PZ/CONS
      ZP=-CZ*PX/CONS + CX*PZ/CONS
      ALFA = ATAN(ZP/YP)
      FT(1,1)=CX
      FT(1,2)=CY
      FT(1,3)=CZ
      FT(2,1)=(-CX*CY*COS(ALFA) -CZ*SIN(ALFA) ) /CONS
      FT(2,2)=CONS*COS(ALFA)
      FT(2,3)=(-CY*CZ*COS(ALFA) +CX*SIN(ALFA) ) /CONS
      FT(3,1)=(CX*CY*SIN(ALFA) -CZ*COS(ALFA) ) /CONS
      FT(3,2)=-CONS*SIN(ALFA)
      FT(3,3)=(CY*CZ*SIN(ALFA) +CX*COS(ALFA) ) /CONS
      N=0
      DO 10 I=1,3
      N=N+3
      DO 10 J=1,3
      DO 10 KK=1,3
10    FT(J+N,KK+N)=FT(J,KK)
100   FORMAT (12(1X,F6.3))
      RETURN
      END


      SUBROUTINE ASSEM1(N2,GM2,GK2,N3,GM3,GK3,GM23,GK23)
      DIMENSION GM2(200,200),GK2(200,200),GM3(200,200),
     1 GM23(200,200),GK23(200,200),TEMP1(200),TEMP2(200),
     1 GK3(200,200)
      DO 5 I=1,200
      DO 5 J=1,200
      GM23(I,J)=0.
5     GK23(I,J)=0.
      DMAS=50./(N2+1)
      GMAS=25.
      DO 7 I=1,(N2+1)*6,6
      DO 7 II=1,3
7     GM2(II+I-1,II+I-1)=GM2(II+I-1,II+I-1)+DMAS
C*****************************************************
C*    HERE THE GRIPPER MASES ARE ADDED               *
C*****************************************************
      GM3(N3*6+1,N3*6+1)=GM3(N3*6+1,N3*6+1)+GMAS
      GM3(N3*6+2,N3*6+2)=GM3(N3*6+2,N3*6+2)+GMAS
      GM3(N3*6+3,N3*6+3)=GM3(N3*6+3,N3*6+3)+GMAS
C*****************************************************
C     HERE THE ASSEMBLING STARTS                     *
C*****************************************************
      DO 10 I=1,N2*6+6
      DO 10 J=1,N2*6+6
      GM23(I,J)=GM2(I,J)
10    GK23(I,J)=GK2(I,J)
```

```
      DO 20 I=1,N3*6+6
      DO 20 J=1,N3*6+6
      GM23(N2*6+I,N2*6+J)=GM23(N2*6+I,N2*6+J)+GM3(I,J)
20    GK23(N2*6+I,N2*6+J)=GK23(N2*6+I,N2*6+J)+GK3(I,J)
      N=(N2+N3)*6+6
      DO 30 I=1,6
      DO 30 J=1,N
      TEMP1(J)=GM23(I,J)
      TEMP2(J)=GK23(I,J)
      GM23(I,J)=GM23(I+6,J)
      GK23(I,J)=GK23(I+6,J)
      GM23(I+6,J)=TEMP1(J)
30    GK23(I+6,J)=TEMP2(J)
      DO 40 J=1,6
      DO 40 I=1,N
      TEMP1(I)=GM23(I,J)
      TEMP2(I)=GK23(I,J)
      GM23(I,J)=GM23(I,J+6)
      GK23(I,J)=GK23(I,J+6)
      GM23(I,J+6)=TEMP1(I)
40    GK23(I,J+6)=TEMP2(I)
      RETURN
      END


      SUBROUTINE ASSEM(N1,GM1,GK1,N,GM23,GK23,GM,GK)
      DIMENSION GM1(200,200),GK1(200,200),GM23(200,200),
     1 GK23(200,200),GM(200,200),GK(200,200)
      DO 5 I=1,200
      DO 5 J=1,200
      GM(I,J)=0.
5     GK(I,J)=0.
C*****************************************************************
C*    ASSEMBLING STARTS HERE                                     *
C*****************************************************************
      DO 20 I=1,N1*6+6
      DO 20 J=1,N1*6+6
      GM(I,J)=GM1(I,J)
20    GK(I,J)=GK1(I,J)
      DO 30 I=1,N
      DO 30 J=1,N
      GM(N1*6+I,N1*6+J)=GM(N1*6+I,N1*6+J)+GM23(I,J)
30    GK(N1*6+I,N1*6+J)=GK(N1*6+I,N1*6+J)+GK23(I,J)
      RETURN
      END


      SUBROUTINE DIMEN(THETA,B1,C1,D1)
```

```
      DIMENSION AL(3),THETA(3),TO1(4,4),TO2(4,4),R1(4),
     1 R2(4),RR1(4),RR2(4),A(8,3),Q(4,4),R3(4),RR3(4),
     1 T12(4,4),T23(4,4),B1(3),C1(3),D1(3),TO3(4,4)
      AL(1)=0.
      AL(2)=1.016
      AL(3)=1.5113
      ALFA=3.141592/2.
100   CALL TRANS(AL(1),THETA(1),ALFA,TO1)
      CALL TRANS(AL(2),THETA(2),0.,T12)
      CALL TRANS(AL(3),THETA(3),0.,T23)
      CALL VMULFF(TO1,T12,4,4,4,4,4,TO2,4,IER)
      CALL VMULFF(TO2,T23,4,4,4,4,4,TO3,4,IER)
5     FORMAT(5X,4E10.3)
      R1(1)=1.
      R1(2)=0.
      R2(1)=1.
      R2(2)=-1.5113
      R3(1)=1.
      R3(2)=-1.6613
      CALL VMULFF(TO3,R1,4,4,1,4,4,RR1,4,IER)
      CALL VMULFF(TO3,R2,4,4,1,4,4,RR2,4,IER)
      CALL VMULFF(TO3,R3,4,4,1,4,4,RR3,4,IER)
      DO 10 I=1,3
      D1(I)=RR1(I+1)
      C1(I)=RR3(I+1)
10    B1(I)=RR2(I+1)
      RETURN
      END


      SUBROUTINE TRANS(AL,THETA,ALFA,A)
      DIMENSION A(4,4)
      CT=COS(THETA)
      ST=SIN(THETA)
      CA=COS(ALFA)
      SA=SIN(ALFA)
      A(1,1)=1.
      A(2,1)=AL*CT
      A(2,2)=CT
      A(2,3)=-ST*CA
      A(2,4)=ST*SA
      A(3,1)=AL*ST
      A(3,2)=ST
      A(3,3)=CT*CA
      A(3,4)=-CT*SA
      A(4,3)=SA
      A(4,4)=CA
      RETURN
      END
```

```
$*****************************************************************
$*                                                               *
$*      LISTING OF THE PROGRAM ++STRUDL++                        *
$*      PROGRAM TO CALCULATE THE NATURAL FREQUENCY OF            *
$*      ROBOT USING STANDARD PACKAGE GTSTRUDL                    *
$*                                                               *
$*****************************************************************

*TITLE 'ANALYSIS OF ROBOT'
STRUDL 'PROJECT' 'FREE VIBRATION OF ROBOT'
TYPE SPACE FRAME
UNITS NEWTON METER
GENERATE 39 JOINTS ID 1,1  X 0.0 .0180804 Y 0.0 0.0 Z 0.0
0.0196966
GENERATE 6 JOINTS ID 40,1 X 0.6731571 .00231665 Y 0.0 0.0
Z .7563947 -.00132065
GENERATE 56 JOINTS ID 46,1 X 0.7105024 .0234454 Y 0.0 0.0
Z .7351052 -.0133655
GENERATE 38 MEMBERS ID 1,1 FROM 1,1 TO 2,1
GENERATE 55 MEMBERS ID 46,1 FROM 46,1 TO 47,1
MEMBER INCIDENCES
39      40      41
40      41      42
41      42      43
42      43      44
43      44      45
44      45      39
45      39      46
STATUS SUPPORT 1
MEMBER PROPERTIES PRISMATIC
1 TO 38 AX 0.01016  IZ 3.07E-05  IY 3.07E-05  IX 6.14E-05
39  TO 100 AX 0.00798 IZ 1.81E-05 IY 1.81E-05 IX 3.62E-05
UNITS METERS NEWTONS CYCLES
CONSTANTS E 2000000. MEMBERS 1 TO 38
E 1379310.3 MEMBERS 39 TO 44
E 689655.17 MEMBERS 45 TO 100

DENSITY 76518. MEMBERS 1 TO 38
DENSITY 51502. MEMBERS 39 TO 44
DENSITY 26487. MEMBERS 45 TO 100

INERTIA OF JOINTS LUMPED
INERTIA OF JOINTS ADD 39 TO 45 LINEAR X 7.1428571
Y 7.1428571 Z 7.1428571
INERTIA OF JOINTS ADD 101    LINEAR X 25. Y 25. Z 25.0
PLOT FORMAT ORIENTATION NON STANDARD
PLOT PLANE
EIGENPROBLEM PARAMETERS
SOLVE USING SUBSPACE
NUMBER ON MODES 6
END
DYNAMIC ANALYSIS MODAL
```

```
PRINT DYNAMIC DEGREES OF FREEDOM ALL
PRINT DATA
FINISH
```

```
C*****************************************************************
C*                                                               *
C*      LISTING OF THE PROGRAM ++DYNCON++                        *
C*      PROGRAM TO CONDENSE DYNAMIC MATRICES USING               *
C*      GUYAN'S REDUCTION TECHNIQUE                              *
C*                                                               *
C*****************************************************************
      REAL GMM(160,160),GKK(160,160),KSSINV(160,160),
     1 GKKINV(160,160),WKAREA(10000),LAMDA(160,160),
     1 OMEGA(160),TEMP3(160,160),TEMP6(160),TEMP8(160,160),
     1 GCC(160,160),GMMM(160,160),GKKK(160,160),GC(160,160),
     1 F(160),KSS(160,160),TEMP7(160),GCR(160,160),MAX,
     1 R(160),TEMP1(160),TEMP2(160),KMS(160,160),T(160,160),
     1 TEMP4(160,160),GKR(160,160),GMR(160,160),
     1 WK(10000),TEMP5(160,160)
      COMPLEX ILAMDA(160,160),EIGVAL(160),BETVAL(160),
     1 EIGVEC(160,160)
      OPEN (UNIT=14,FILE='STIF.DAT',TYPE='OLD')
      OPEN (UNIT=23,FILE='MAS.DAT',TYPE='OLD')
C*****************************************************************
C     N = TOTAL NO. OF D.O.F.                                   *
C     NSD=NO. OF SLAVE D.O.F                                    *
C     NR =NO. OF RETAINED MASTER D.O.F.                         *
C*****************************************************************
      N=156
      NSD=136
      NR=N-NSD
      NN=N
      READ(14,*)((GKK(I,J),J=1,N),I=1,N)
      READ(23,*)((GMM(I,J),J=1,N),I=1,N)
C*****************************************************************
C     THE CONDENSATION STARTS HERE                              *
C*****************************************************************
      DO 200 II=1,NSD
      MAX=0.
      DO 60 I=1,N
60    R(I)=GKK(I,I)/GMM(I,I)
      DO 70 I=1,N
      IF(R(I).GT.MAX) THEN
              MAX=R(I)
              IMAX=I
      ELSE
              GO TO 70
      ENDIF
70    CONTINUE
      DO 80 J=1,NN
      TEMP1(J)=GMM(IMAX,J)
      TEMP2(J)=GKK(IMAX,J)
      TEMP6(J)=GCC(IMAX,J)
80    TEMP7(J)=F(IMAX)
      DO 90 I=IMAX,N-1
      DO 90 J=1,NN
```

```
      GMM(I,J)=GMM(I+1,J)
      GKK(I,J)=GKK(I+1,J)
      GCC(I,J)=GCC(I+1,J)
90    F(I)=F(I+1)
      DO 100 J=1,NN
      GMM(N,J)=TEMP1(J)
      GKK(N,J)=TEMP2(J)
      GCC(N,J)=TEMP6(J)
100   F(N)=TEMP7(J)
      DO 110 I=1,NN
      TEMP1(I)=GMM(I,IMAX)
      TEMP2(I)=GKK(I,IMAX)
110   TEMP6(I)=GCC(I,IMAX)
      DO 120 J=IMAX,N-1
      DO 120 I=1,NN
      GMM(I,J)=GMM(I,J+1)
      GKK(I,J)=GKK(I,J+1)
120   GCC(I,J)=GCC(I,J+1)
      DO 130 I=1,NN
      GMM(I,N)=TEMP1(I)
      GKK(I,N)=TEMP2(I)
130   GCC(I,N)=TEMP6(I)
      N=N-1
200   CONTINUE
      N=NN
      DO 140 I=1,NSD
      DO 140 J=1,NSD
140   KSS(I,J)=-1.*GKK(I+NR,J+NR)
      DO 150 I=1,NR
      DO 150 J=1,NSD
150   KMS(I,J)=GKK(I,J+NR)
      IDGT=4
      CALL LINV2F(KSS,NSD,160,KSSINV,IDGT,WKAREA,IER)
      CALL VMULFP(KSSINV,KMS,NSD,NSD,NR,160,160,TEMP8,
     1 160,IER)
      DO 160 I=1,NR
160   T(I,I)=1.
      DO 170 I=1,NSD
      DO 170 J=1,NR
170   T(I+NR,J)=TEMP8(I,J)
      CALL VMULFM(T,GKK,N,NR,N,160,160,TEMP3,160,IER)
      CALL VMULFF(TEMP3,T,NR,N,NR,160,160,GKR,160,IER)
      CALL VMULFM(T,GMM,N,NR,N,160,160,TEMP4,160,IER)
      CALL VMULFF(TEMP4,T,NR,N,NR,160,160,GMR,160,IER)
      CALL VMULFM(T,GCC,N,NR,N,160,160,TEMP5,160,IER)
      CALL VMULFF(TEMP5,T,NR,N,NR,160,160,GCR,160,IER)
      N=NR
C*****************************************************************
C     INCLUSION OF DAMPING MATRIX                                *
C*****************************************************************
      DO 177 I=1,N
      DO 177 J=1,N
```

```
      GMMM(I,J+N)=GMR(I,J)
      GMMM(I+N,J)=GMR(I,J)
      GMMM(I+N,J+N)=GCR(I,J)
      GKKK(I,J)=-1.*GMR(I,J)
177   GKKK(I+N,J+N)=GKR(I,J)
      N=2*N
      IDGT =4
      CALL LINV2F (GKKK,N,160,GKKINV,IDGT,WKAREA,IER)
      CALL VMULFF (GKKINV,GMMM,N,N,N,160,160,LAMDA,
     1 160,IERR)
      DO 220 I=1,N
      DO 220 J=1,N
220   ILAMDA(I,J)=CMPLX(LAMDA(I,J))
C*************************************************************
C     DETERMINATION OF EIGEN VALUES                          *
C*************************************************************
      IJOB =1
      CALL EIGCC(ILAMDA,N,160,IJOB,EIGVAL,EIGVEC,160,
     1 WK,IERROR)
      DO 133 I=1,N
      EIGVAL(I)=1./EIGVAL(I)
      EIGVAL(I)=EIGVAL(I)/2./3.1415927
133   CONTINUE
      PRINT *, 'IERROR=',IERROR,'EIGEN VALUES ARE='
      PRINT 16,(EIGVAL(I),I=1,N)
16    FORMAT (2X,2E13.7,5X,2E13.7)
      STOP
      END
```

```
C*****************************************************************
C*                                                               *
C*     LISTING OF THE PROGRAM ++COMCON++                         *
C*     PROGRAM TO CONDENSE THE DYNAMIC MATRICES USING            *
C*     COMPONENT MODE SYNTHESIS TECHNIQUE                        *
C*                                                               *
C*****************************************************************

C*****************************************************************
C       FR=FROM ROW WHICH HAS TO BE SHIFTED                       *
C       TR=TO ROW WHICH RECEIVES THE SHIFTED FROM ROW             *
C       NN=INITIAL SIZE OF STIFFNESS MATRIX AFTER                 *
C          APPLYING BOUNDARY CONDITION                            *
C       NO= EIGENVECTORS TO BE RETAINED INCLUDING                 *
C          INTERFACE CORDINATE                                    *
C       NI=NUMBER OF INTERFACE COORDINATES                        *
C       N=FR-TR=NN-NI=SIZE OF THE KFF MATRIX                      *
C*****************************************************************

      REAL GM(160,160),GK(160,160),M(4,4),K(4,4),
     1 GKK(160,160),GKKINV(160,160),WKAREA(16000),
     1 WK(16000),OMEGA(160),MASS,L,INER,TEMP1(160),
     1 GMMM(160,160),GKKK(160,160),RKINV(160,160),
     1 KFI(160,160),TEMP3(160,160),REIG(160,160),
     1 TEMP4(160,160),RK(160,160),TEMP5(160,160),
     1 GMM(160,160),LAMDA(160,160),TEMP2(160),A(160,160),
     1 RM(160,160)
      COMPLEX ILAMDA(160,160),EIGVAL(160),BETVAL(160),
     1 EIGVEC(160,160)
      INTEGER FR,TR
      OPEN (UNIT=14,FILE='STIF.DAT',TYPE='OLD')
      OPEN (UNIT=23,FILE='MAS.DAT',TYPE='OLD')
      NN=156
      NI=6
      NO=7
      N=NN-NI
      READ (14,*) ((GKK(I,J),J=1,NN),I=1,NN)
      READ (23,*) ((GMM(I,J),J=1,NN),I=1,NN)
C*****************************************************************
C       CONDENSATION STARTS HERE.                                 *
C*****************************************************************
      TR=0
      DO 100 FR=NN-5,NN
      TR=TR+1
      DO 35 J=1,NN
      TEMP1(J)=GMM(FR,J)
35    TEMP2(J)=GKK(FR,J)
      DO 40 I=1,N
      II=FR+1-I
         DO 40 J=1,NN
         GMM(II,J)=GMM(II-1,J)
40       GKK(II,J)=GKK(II-1,J)
```

```
      DO 50 J=1,NN
      GMM(TR,J)=TEMP1(J)
50    GKK(TR,J)=TEMP2(J)
      DO 60 I=1,NN
      TEMP1(I)=GMM(I,FR)
60    TEMP2(I)=GKK(I,FR)
      DO 70 J=1,N
      JJ=FR+1-J
        DO 70 I=1,NN
        GMM(I,JJ)=GMM(I,JJ-1)
70      GKK(I,JJ)=GKK(I,JJ-1)
      DO 80 I=1,NN
      GMM(I,TR)=TEMP1(I)
80    GKK(I,TR)=TEMP2(I)
100   CONTINUE
      DO 110 I=1,N
      DO 110 J=1,NI
110   KFI(I,J)=GKK(I+NI,J)
      DO 120 I=1,N
      DO 120 J=1,N
      GKKK(I,J)=GKK(I+NI,J+NI)
120   GMMM(I,J)=GMM(I+NI,J+NI)
      IDGT =4
      CALL LINV2F (GKKK,N,160,GKKINV,IDGT,WKAREA,IER)
      CALL VMULFF (GKKINV,GMMM,N,N,N,160,160,LAMDA,
     1 160,IERR)
      DO 130 I=1,N
      DO 130 J=1,N
130   ILAMDA(I,J)=CMPLX(LAMDA(I,J))
C*************************************************************
C     DETERMINATION OF EIGEN VALUES                          *
C*************************************************************
      IJOB =1
      CALL EIGCC(ILAMDA,N,160,IJOB,EIGVAL,EIGVEC,160,
     1 WK,IERROR)
      DO 140 I=1,N
140   EIGVAL(I)=1./EIGVAL(I)
      DO 150 I=1,N
      DO 150 J=1,NO-NI
150   REIG(I,J)=REAL(EIGVEC(I,J))
      CALL VMULFF(GKKINV,KFI,N,N,NI,160,160,TEMP3,
     1 160,IER)
C*************************************************************
C     FORMATION OF TRANSFORMATION MATRIX                     *
C*************************************************************
      DO 155 I=1,NI
155   A(I,I)=1.
      DO 160 I=1,N
      DO 160 J=1,NI
160   A(I+NI,J)=-TEMP3(I,J)
      DO 170 I=1,N
      DO 170 J=1,NO-NI
```

```
170      A(I+NI,J+NI)=REIG(I,J)
         CALL VMULFM(A,GKK,NN,NO,NN,160,160,TEMP4,160,IER)
         CALL VMULFF(TEMP4,A,NO,NN,NO,160,160,RK,160,IER)
         CALL VMULFM(A,GMM,NN,NO,NN,160,160,TEMP5,160,IER)
         CALL VMULFF(TEMP5,A,NO,NN,NO,160,160,RM,160,IER)
         IDGT =4
         CALL LINV2F (RK,NO,160,RKINV,IDGT,WKAREA,IER)
         CALL VMULFF (RKINV,RM,NO,NO,NO,160,160,LAMDA,
     1   160,IERR)
         DO 180 I=1,NO
         DO 180 J=1,NO
180      ILAMDA(I,J)=CMPLX(LAMDA(I,J))
C*****************************************************
C        DETERMINATION OF EIGEN VALUES                *
C*****************************************************
         IJOB =1
         CALL EIGCC(ILAMDA,NO,160,IJOB,EIGVAL,EIGVEC,160,
     1   WK,IERROR)
         DO 190 I=1,NO
190      EIGVAL(I)=1./EIGVAL(I)
         PRINT *, 'IERROR=',IERROR,'EIGEN VALUES ARE='
         PRINT 200,(EIGVAL(I),I=1,NO)
200      FORMAT (2X,12E10.4)
         PRINT *,'THE FREQUENCIES ARE'
         DO 210 I=1,NO
         OMEGA(I)=SQRT(REAL(EIGVAL(I)))
         PRINT *,'I=',I,'FREQUENCY=',OMEGA(I)
210      CONTINUE
230      FORMAT (2(E12.4,4X))
         STOP
         END
```