GENERALIZED FINITE ELEMENT DYNAMIC MODELLING AND SIMULATION FOR FLEXIBLE ROBOT MANIPULATORS

Zhou Feng

A THESIS

in the

DEPARTMENT OF MECHANICAL ENGINEERING

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science

at Concordia University

Montréal, Quebéc, Canada

January, 1993

ISBN 0-315-84623-2

Canada

ABSTRACT

Generalized Finite Element Dynamic Modelling And Simulation For Flexible Robot Manipulators

Zhou Feng

Generalized system modelling using finite element technique is examined in order that the dynamic behaviour of flexible robot manipulators can be studied. Most of the dynamic models currently used for robot motion simulation are restricted to open kinematic chains of rigid body elements. However, to improve dynamic performance, it is essential to take into account the structural mechanic characteristics of robot components. This thesis presents a general finite element model for flexible robots and validates them by simulation.

The kinematic equations are set up and the system kinetic and potential energies as well as their derivatives are obtained. Lagrange's formulation is employed and the generalized system dynamic equations which represent spatial performance are solved based on finite element analysis.

To validate the generalized equations developed for flexible robot manipulator, two sample simulations are carried out for 2-link and single link manipulators without and with torque. The responses are obtained, compared and discussed.

ACKNOWLEDGEMENTS

The author is sincerely grateful to his thesis supervisor Dr. T.S.Sankar, under whose research guidance this work was completed, for his ideas, help, financial assistance and continuous encouragement during the course of this investigation.

The help and assistance of many friends and colleagues, especially those of Dr. Chang-jin Li, are gratefully acknowledged.

Finally, the author is grateful to his parents for their abundant moral support and understanding throughout the course of this Masters program.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A: Mass Matrix

A : joint transformation matrix

B : transformation matrix from origin of base coordinate to the origin of the local coordinate

C_{ij}: undeformed position vector

D_i: the link transformation matrix for link i

EA: structural rigidities of link due to extension

EI_z, EI : bending stiffness of link

F_1 to $F_{1,1}$, $F_{1,1}$ to $F_{1,1,1}$: coefficients

GI_x: structure rigidities of link due to torsion

H: coefficient vector

K: kinetic energy

K':stiffness matrix

l_{ij}: jth element length of ith link

m: total number of links

n: total number of elements in one link

N_{ij}^T : shape function matrix

$N_{ijxk}, N_{ijyk}, N_{ijzk}, N_{ij\beta k}$: shape functions

$O\,X\,Y_0Z_0$, $O\,X_i\,Y_iZ_i$: local coordinate system

P : elastic deformation matrix for link i

q: system's variable vector

$r_{ij}^1(x)$, $r_{ij}(x)$: deflection

T_1^0, T_i : the transformation matrix from base coordinate system to local coordinate system

$u_{ijxk}, u_{ijyk}, u_{ijzk}, u_{ij\beta k}$: finite elements nodal displacement

$\boldsymbol{U}_{ij}^{i}$: the nodal variable vector

u_i: elastic variables for link 1 in the simulations

U'_1, U'_2, U'_3, U'_4: nodal deflection in direction U'

$(V_{ij})_e$, $(V_{ij})_g$: potential energy

V'_1, V'_2, V'_3, V'_4: nodal deflection in direction V'

w_i: elastic variables for link 2 in the simulations

W'_1, W'_2, W'_3, W'_4: nodal deflection in direction W'

x_{ij}: position of the point in the jth element measured from O_i.

$\phi_{ijx}^{i}, \phi_{ijy}^{i}, \phi_{ijz}^{i}, \phi_{ij\beta}^{i}$: deflections

θ_i : joint variable

μ : density of the link

τ : driving torque

CHAPTER 1

INTRODUCTION

1.1 PRELIMINARIES

Most of present analysis and design for industrial robots are based on the assumption that robot arms are composed of rigid bodies. This assumption brings a lot of convenience in both analysis and study of these robot manipulators. Control systems for robot manipulators based on this assumption are designed so that the end-effector of the robot manipulator will be in a certain position after joints are driven to a given set of precomputed angular positions. But in fact, the manipulator links are not absolutely rigid. The rigidity assumption is, in fact, presumes that if the stiffness of the object is so large compared to its geometry that the influence of its flexibility may be neglected and the object could be considered as rigid. In order to keep the robot arms rigid as per design when they are performing at high speed and/or carrying heavy payloads, robot manipulators have to be built massive, hence will encounter high material cost and large energy consumption.

Noticing these drawbacks of rigid robot arm design, the concept of flexible robot manipulator is introduced to replace the analytical premise of rigid robot arm. Compared with the traditional design of rigid robot manipulators, the flexible

robot manipulators not only are able to carry equally heavy payloads and can be fast moving without having to design each link thick and strong, but also possess many advantages: they need less material, are more maneuverable, are lighter and hence consume less power, possess smaller volume and are more easily transportable [1-2].

The main obstacle preventing wide use of flexible robot manipulator is due in part to the fact that it is often difficult to obtain a reasonable accuracy of response corresponding to the input commands from the control system [3]. At the same time, the robot model used in the design should be able to describe the identifiable behaviour of the manipulators with adequate accuracy. An improved dynamic model (i.e., the dynamic equations of motion describing the dynamic behaviour) of such manipulators with option to include flexibility is therefore needed for adoption in the design and simulation of the control scheme to provide appropriate control of lightweight manipulators at increased speeds and heavier payloads. This can be achieved through the description of link flexibility by the finite element with an appropriate selection of the number of elements.

It is evident that the more finite elements adopted, the more accurate the model response can be [4]. However, the fact of reality is, one can not simply take more finite elements. The reason is that the computation of that model will rise sharply with the growth of number of finite elements.

Therefore, as a compromise, a suitable number of finite elements are employed to yield the necessary design tolerance. In many cases, just a few finite elements are found to be sufficient.

1.2 PROBLEM DEFINITION

The dynamic model essentially describes the dynamic characteristics of flexible manipulators. Based on this model, the positions, velocities and accelerations of the manipulator links could be determined by the input parameters dictated by the control system. In other words the dynamic models are to be prescribed in the form [5]

$$\tau = A(q) * \ddot{q} + H(q, \dot{q}) \qquad (1.1)$$

where τ is a vector of generalized forces or torques,

q is a vector of variables of the system,

A is a square mass matrix,

H is a coefficient vector including gravity effect etc..

In order to achieve this, various approaches have been developed to conform to such a model. The two main approaches to the dynamic modelling of flexible robot arms are the assumed modes method and the finite element method. The assumed modes method is quite general and contains less

variables and can lead to efficient computation. However, this method requires the precise assumed mode shape for the links and hence limits the use of this method only for manipulators with simple shaped links [6]. The finite element method, however, could be applied to manipulators when links are of non-regular cross-sections and include better accuracy as well [7].

A number of investigations have been carried out employing finite element approach to derive the dynamic equations. However, these equations are restricted to only certain cases. In this thesis, the main contribution involves derivation of the generalized model equations in a finite element setting. This approach is found more suitable for design of robot control.

1.3 PRESENT STATE OF THE DYNAMIC MODELLING THROUGH FINITE ELEMENT

The methods for dynamic modelling of manipulators by means of finite element concept were presented by Geradin [8] and Bricont [9]. Each flexible link was considered as one element so that the dynamic equations could be derived by the Lagrangian method. Since these methods did not discuss the case of a link with several elements, the accuracy could only be improved by choosing an appropriate mode shape of elements.

Usoro, Nadira and Mahil [10] developed another method to advance the dynamic model based on the finite element approach. With each flexible link of the manipulator being divided into elements, the accuracy of the response was achieved by increasing the number of elements of each flexible link. However, this investigation has application limitations since the dimension of mass matrix of the equations is dependent on the number of elements chosen for each link and the number of links. With the number of links increased and more elements chosen, the mass matrix becomes too bulky to be handled. Therefore, these authors discussed only the case of a two-link manipulator with two elements for each link instead of deriving more generalized equations for design purpose.

Book [11] proposed a method of recursive Lagrangian dynamics for flexible manipulator arms. Here [4x4] matrices are utilized to represent both the joint and deflection motions. The computational approach resulting from the Lagrangian formulation of the system dynamics, that is reduced to recursive form, has already been proven as an efficient method for manipulators with rigid links. To consider the flexibility of robot manipulator, the so called assumed modes method, which assumes the link's deformed shape by a certain describing function, is employed. Since the assumed modes method, where a certain mode shape is presumed for each link, is adopted in this work, this may not be suitable to describe

the manipulators with complex shaped links.

A computer simulation of the control of a flexible robot arm was carried out by Lee and Wang [12]. The dynamic equations for a single-link flexible robot arm were derived in a rigorous manner. The flexible beam is modeled as *n* beam elements and *n+1* nodes. As the model employs the Newton and not Lagrangian formulation, it could only be fit for simple manipulator examples. In addition, the calculation of some coefficients grows very rapidly with an increase in the number of links. Here, the moment acting on the beam could not always be considered as zero particularly in case of a multi link arm. If the moment could not be ignored, it will become difficult to obtain the governing equations using this approach for a flexible manipulator arm.

Based on the above work, Lee and Wang [13] developed the dynamic equations for a two-link flexible robot arm. The main success in this work centres around the determination of the inverse of matrix A^* which links the acceleration vector with the vector combining velocity variables and position variables. However, the inverse of A^* can not be easily obtained as well as it is impossible to obtain this quantity for a more general case. With an increase in the number of links, calculations of velocity and acceleration also become extremely complicated.

Chang and Hamilton [14] presented a dynamic model for robot manipulators with flexible links by means of the finite

element method and Lagrange's formulation. In this work, an Equivalent Rigid Link System (ERLS) model was used which describes the kinematics of robot manipulators with flexible links by separating the global motion of the flexible manipulator into large and small motions. This algorithm was suggested to simplify the analysis of the flexible robot arms. However the authors [14] did not consider the case of manipulator links having more than one element. They just considered the equations of links with single element.

Gaultier and Cleghorn [15] use the Hamilton's principle to implement the modelling of both planar and spatial flexible manipulators by creating a beam finite element. The element employs torsional, axial, and lateral shape functions to approximate the link deformation vector. The governing differential equations for an arbitrary spatially translating and rotating flexible link were finally derived. The derivation was restricted only to links which can adopt to Euler-Bernoulli beam theory of constant cross sectional area.

Naganathan and Soni [16] presented a finite, element-based nonlinear model of flexibility effects in certain class of manipulators. The governing equations of motion are derived including the effects of rotary inertia, shear deformation, and effects of nonlinear motion of each of the links. However, a uniform cross section of manipulator links is assumed which limits the application of this method.

It can be realized from the above literature survey that

the various approaches for flexible manipulator modelling by the previous researchers can be classified under 2 major groups:

1. Use of assumed modes: This approach is recommended by Book [11], Gordaninejad and Azhdari [17]. However, these researchers offer discussions and applications that are limited only to manipulator links with uniform cross section.

2. Use of finite element: This approach is recommended by Geradin [8], Usoro [10], Naganathan and Soni [16], Lee and Wang [12] [13], Bricout [9], Jonker [18], Chang and Hamilton [14] [19]. As mentioned previously, these investigations do not give or lead to a generalized dynamic equations for any type of flexible robot manipulators. References [10] [12] [13] discuss only specific cases with one or, at best, two links. Investigations [8] [14] derived the dynamic equations for flexible manipulators with one element in each link. This certainly reduces the ability of these approaches to obtain a better accuracy of the manipulator response and behaviour which may be needed in certain robotic designs and control. As all the above works are limited to specific manipulators, they can not be employed for any general type of robot manipulators making use of finite element method. This shortcoming is addressed in this investigation.

1.4 PROFILE OF THE THESIS

The objective of this thesis is to derive a generalized dynamic equation, which can be employed for any flexible robot manipulator with revolute joints. The finite element approach is employed to model any multi-link flexible manipulators. The whole work presented in the following chapters of this thesis may be divided into 3 stages:

The first stage is to analyze the kinematic character of the flexible manipulators by the concept of finite element. Two types of coordinates (body-fixed system of coordinates attached to links and coordinates attached to elements of links) are utilized so that the points on a beam could be expressed in terms of these coordinates systems. Small deflection characteristic is assumed in order to simplify the analysis to linear relationships and so that the general transformation matrix can be easily set up. The adoption of finite element is introduced in the analysis at this stage.

The second stage is to derive the dynamic equations for flexible robot manipulators. As the position of each point on the link could be expressed by means of the transformation matrix, the kinetic energy for each element is computed and summed over all the elements. The potential energy is computed in the same manner as well. Then the Lagrangian formulation is applied to set up the dynamic equations of the system.

In the third stage, the test, by means of computational

simulations, is performed in order to check the validity and correctness of the derived dynamic equations. This is important for algorithm development. Certain initial conditions are applied to the simulation in this test procedure. Since the different manipulators deployed in practice may possess varying characteristics, the choice of shape functions may greatly influence the results. In order to compare the present results with those obtained earlier by other methods, the parameter values of the model used by Usoro[10] are also employed here using the same initial conditions and other additional conditions specified in [10]. The validity of the present approach is thus ascertained by the above process of comparison. However, it may be necessary to validate further the present method with additional examples that involve more links and more finite elements. But, there are no published literature giving details of such examples.

In the second simulation, a one-link manipulator with its link divided into two finite elements is also discussed using the same linkage parameters as those given by Usoro[10]. The Runge-Kutta method is employed to solve the dynamic equations of which all the joints of flexible manipulators are revolute as shown in Fig 1.1. A torque is introduced in such a manner that the application potential of the procedures developed in this thesis could be demonstrated.

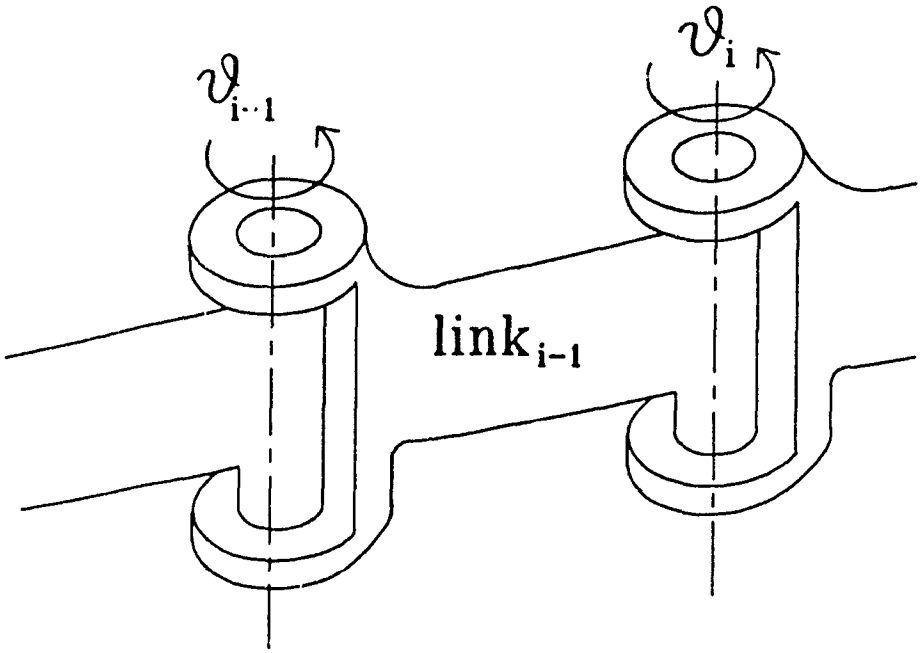


Fig. 1.1 Revolute Joints [20]

CHAPTER 2

KINEMATIC ANALYSIS FOR A ROBOT WITH REVOLUTE TYPE JOINT CONFIGURATION

2.1 INTRODUCTION

Kinematic analysis has been established as the tool to study the spatial configuration of robot as a function of joint angles and other linkage variables. This yields the relationship between the base coordinate system and specified local systems. The transformation from the local coordinate systems to the base system is very important for the proposed analysis as the positions, velocities and accelerations in different systems are to be converted into the same coordinate system which will bring some elegance as well as convenience to the analysis such as computation of system potential energy etc.

2.2 COORDINATE SYSTEM ASSIGNMENT

A manipulator consists of several links connected sequentially by actual joints. Consider the manipulator as described in Fig 2.1, A_i is the joint transformation matrix for joint i, D_{i-1} is the link transformation matrix for link $i-1$ between joints $i-1$ and i, B_i is the cumulative transformation matrix from origin of base coordinate to the

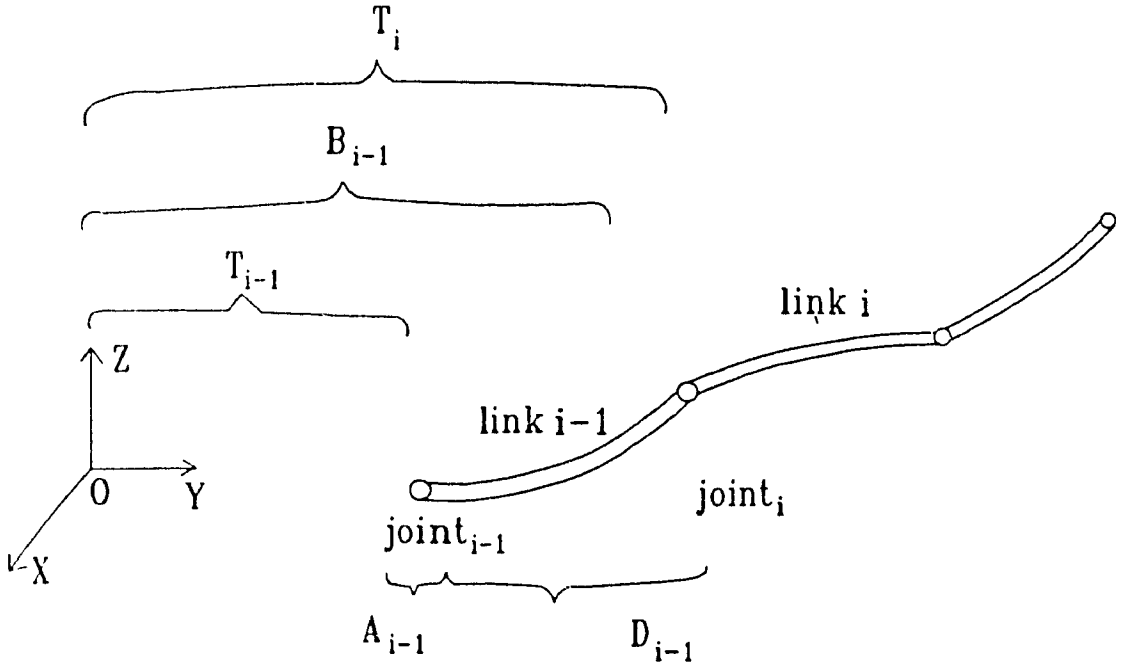


Fig. 2.1 Link Relations for Flexible Manipulator

origin of the local coordinate for link i. The following investigations are made under the assumption of straight link and small deflections.

Let the coordinate system $O_iX_iY_iZ_i$ be defined on link i with the origin of the coordinate system coinciding with joint i so that the x axis is coincident with the neutral axis of the link in its undeformed condition. Thus, the position of the points on link i can be expressed in terms of the coordinate system $O_iX_iY_iZ_i$, which could be understood from Fig.2.2.

Suppose all the manipulator links are made up of many finite elements, the deflection of a spatially defined element can be described in the three directions U', V' and W', as shown in Fig.2.3. For each node of finite elements, there are a total of 6 deflections composed of 3 angular deflections (U'_1, V'_1, W'_1; U'_4, V'_4, W'_4) and 3 linear deflections(U'_1, V'_1, W'_1; U'_3, V'_3, W''_3). These are indicated in the figure for a given element.

2.3 THE KINEMATIC EQUATIONS

Let the following quantities be defined for setting up the kinematic equations of the manipulator links:

$\boldsymbol{r}^i_{ij}(x_{ij})$: the deflection in terms of position variable x_{ij} of the jth element of link i in terms of coordinate $O_iX_iY_iZ_i$, where superscript i denotes the reference to the

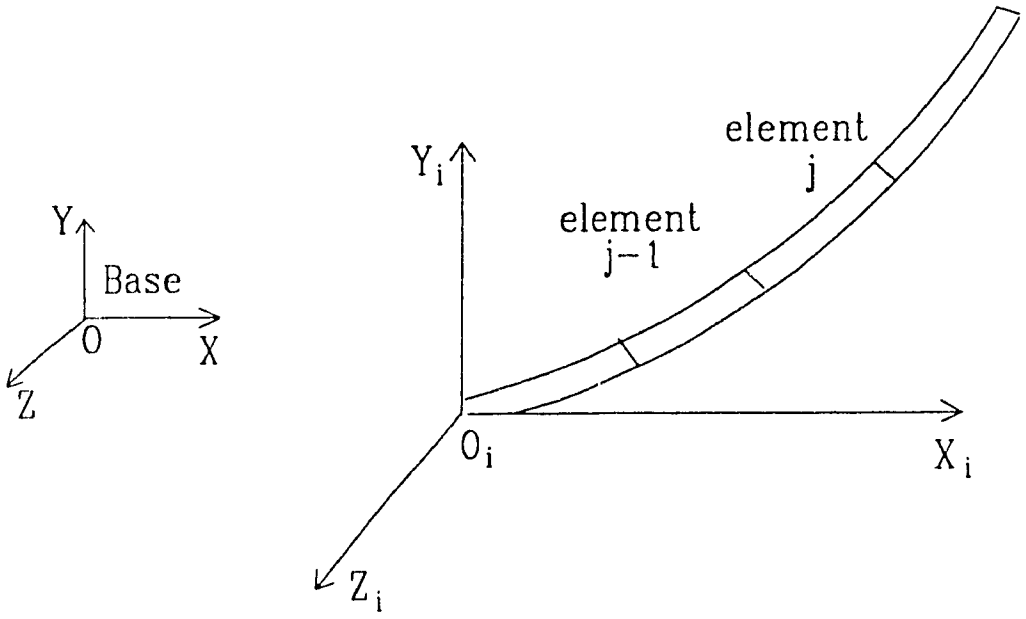


Fig. 2.2 Configuration of Local Coordinate System

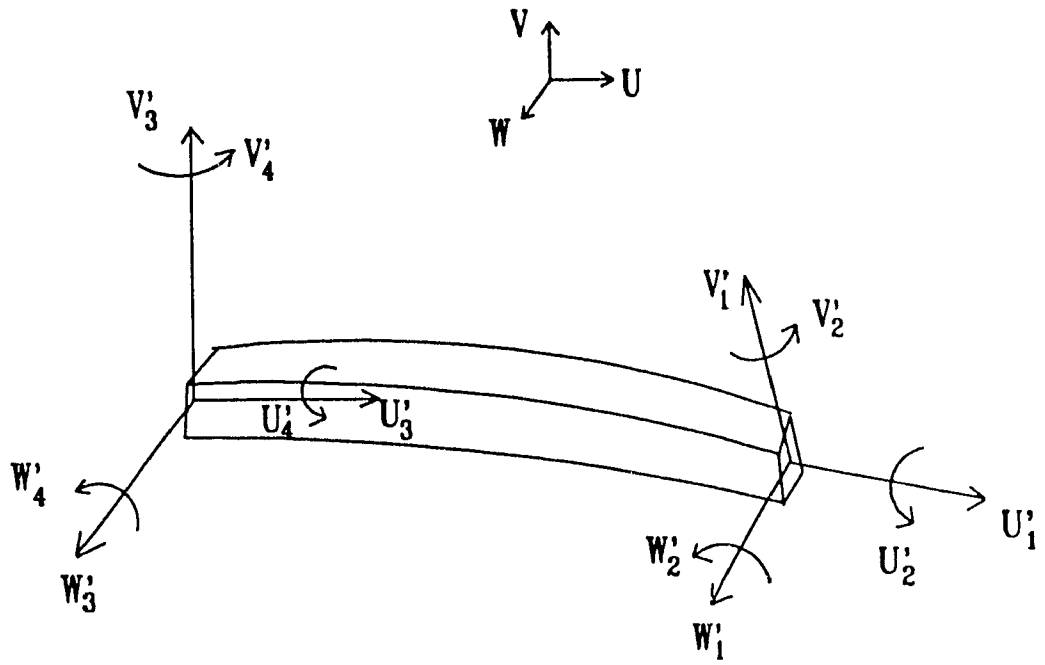


Fig. 2.3 Nodal Displacements

coordinate system $O_iX_iY_iZ_i$.

$\boldsymbol{r}_{ij}(x_{ij})$: the deflection in position variable x_{ij} of the jth element of link i in terms of the base coordinate. The neglection of superscript denotes reference to the base coordinate system.

T_i^o : the transformation matrix for operation from base coordinate system to local system $O_iX_iY_iZ_i$ in terms of base coordinate.

T_i: the simplified form of T_i^o .

Thus, the instantaneous position of a point in jth element on link i in terms of the base coordinate system is given as

$$\boldsymbol{r}_{ij}(x_{ij}) = T_i^o * \boldsymbol{r}_{ij}^{i}(x_{ij}) = T_i * \boldsymbol{r}_{ij}^{i}(x_{ij}) \qquad (2.1)$$

The relationship between $\boldsymbol{r}_{ij}$ and $\boldsymbol{r}_{ij}^{i}$ is also shown in Fig.2.4. The point expressed in terms of ith coordinate system has to be transformed by matrix T_i into the base coordinate system.

The transformation matrix from the base coordinate system to the local system $O_iX_iY_iZ_i$ is expressed as:

$$T_i = T_{i-1} * D_{i-1} * A_i = B_{i-1} * A_i \qquad (2.2)$$

Therefore, the position of the point in terms of

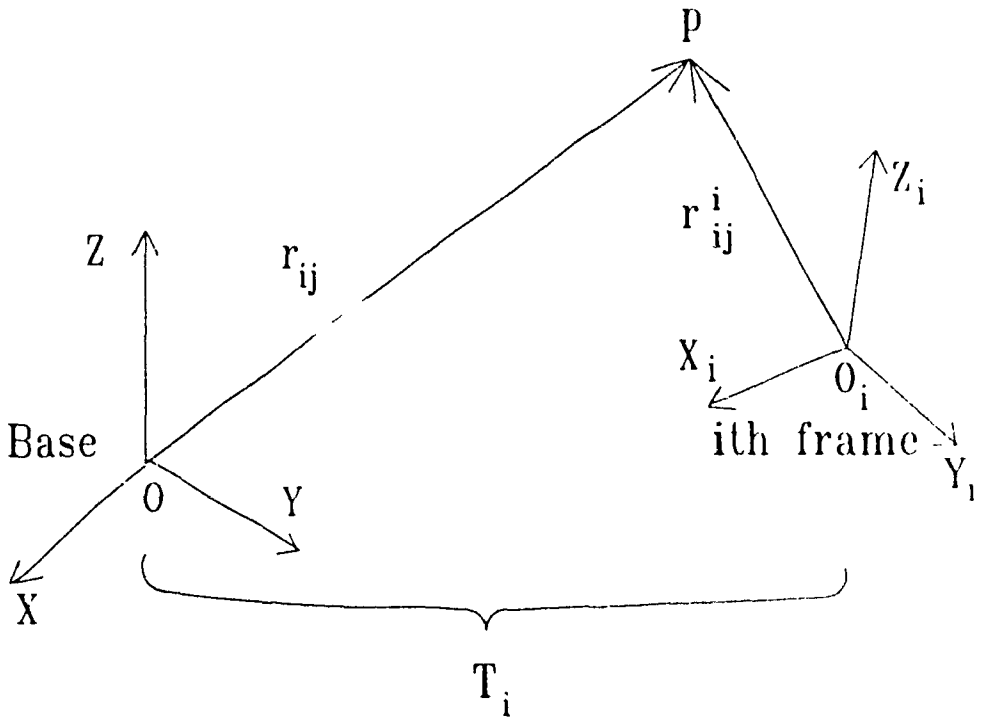


Fig. 2.4 Basic Nomenclature for Transformation

coordinate $O_iX_iY_iZ_i$ is given as:

$$\boldsymbol{r}_{ij}^{i}(x_{ij}) = \begin{bmatrix} (j-1)\,l_{ij} + x_{ij} \\ 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \phi_{ijx}^{i}(x_{ij}, t) \\ \phi_{ijy}^{i}(x_{ij}, t) \\ \phi_{ijz}^{i}(x_{ij}, t) \\ 0 \end{bmatrix} \qquad (2.3)$$

where

l_{ij}: jth element length of ith link

x_{ij}: position of the point in the jth element measured from O_{ij}, and

$O_{ij}X_{ij}Y_{ij}$: Body-fixed coordinates attached to the jth element of link i with O_{ij} fixed to the junction between the jth and $(j-1)$th elements.

In the expression for $\boldsymbol{r}_{ij}^{i}(x_{ij})$, the first part describes the position of the point when the link is undeformed and the second part shows the deformation position of the point. The latter part has to be expanded using element property.

The second part of $\boldsymbol{r}_{ij}^{i}(x_{ij})$ can then be described in terms of shape function in the following manner.

$$\phi_{ijx}^{i}(x_{ij}, t) = \sum_{k=1}^{4} N_{ijxk}(x_{ij}) * u_{ijxk}(t) \qquad (2.4a)$$

$$\phi_{ijy}^{i}(x_{ij}, t) = \sum_{k=1}^{4} N_{ijyk}(x_{ij}) * u_{ijyk}(t) \qquad (2.4b)$$

$$\phi^i_{ijz}(x_{ij},t) = \sum_{k=1}^{4} N_{ijzk}(x_{ij}) * u_{ijzk}(t) \qquad (2.4c)$$

and

$$\phi_{ij\beta}(x_{ij},t) = \sum_{k=1}^{4} N_{ij\beta}(x_{ij}) * u_{ij\beta k}(t) \qquad (2.4d)$$

where

ϕ^i_{ijx}, ϕ^i_{ijy} and ϕ^i_{ijz} are the link deflections along x, y and z axes respectively; $\phi^i_{ij\beta}$ is the torsional angular displacement; N_{ijxk}, N_{ijyk} and N_{ijzk} respectively are the shape functions about the x, y and z axes and u_{ijxk}, u_{ijyk} and u_{ijzk} respectively are finite elements nodal variables about the x, y and z axes.

Since the deformation of link causes both translation and rotation of the coordinate system, the link transformation matrix D_i from joint i-1 to i can be obtained in terms of the nodal vector of the last finite element on the link i and the total number n of the elements in this link.

$$D_i = \begin{bmatrix} 1 & 0 & 0 & nl_{ij} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + P_i \qquad (2.5)$$

The former part of D_i describes the transformation matrix

along the x axis when link is in its undeformed situation. P_i is the additional matrix indicating how much elastic deformation the link takes. Since small deflections have been assumed, it can be shown

$$P_i = \begin{bmatrix} 0 & -u_{iy3} & u_{ix3} & u_{ix4} \\ u_{iy3} & 0 & -u_{iz3} & u_{iy4} \\ -u_{ix3} & u_{iz3} & 0 & u_{iz4} \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (2.6)$$

Here u_{ix3}, u_{iy3} and u_{iz3} respectively are the nodal deformation variables which describe the flexural slopes at tip of link i about axes y, z and x. u_{ix4}, u_{iy4} and u_{iz4} respectively are the nodal deformation variables which describe the flexural deflections at the tip of link i along axes x, y and z.

Taking the time derivative of the expression for the position given by equation(2.1),

$$\frac{d\boldsymbol{r}_{ij}(x_{ij})}{dt} = \dot{T}_i * \boldsymbol{r}^{i}_{ij}(x_{ij}) + T_i * \dot{\boldsymbol{r}}^{i}_{ij}(x_{ij}) \qquad (2.7)$$

The expression for $\dot{r}_{ij}^{i}$ can be obtained by differentiating equation (2.3). Since the first part of r_{ij}^{i} is not dependent on time, the differentiation of this part gives zero. Therefore, the result of the differentiation becomes,

$$\dot{r}_{ij}^{i}(x_{ij}) = \begin{bmatrix} \sum_{k=1}^{4} N_{ijxk}(x_{ij}) * \dot{u}_{ijxk}^{i}(t) \\ \sum_{k=1}^{4} N_{ijyk}(x_{ij}) * \dot{u}_{ijyk}^{i}(t) \\ \sum_{k=1}^{4} N_{ijzk}(x_{ij}) * \dot{u}_{ijzk}^{i}(t) \\ 0 \end{bmatrix} \tag{2.8}$$

By differentiating (2.2), one can obtain

$$\dot{T}_i = \dot{B}_{i-1} * A_i + B_{i-1} * \dot{A}_i \tag{2.9}$$

and

$$\ddot{T}_i = \ddot{B}_{i-1} * A_i + 2\dot{B}_{i-1} * \dot{A}_i + B_{i-1} * \ddot{A}_i \tag{2.10}$$

where

$$\dot{A}_i = \frac{\partial A_i}{\partial \theta_i} * \dot{\theta}_i \tag{2.11a}$$

and

$$\ddot{A}_i = \frac{\partial^2 A_i}{\partial \theta_i^2} * \dot{\theta}_i^2 + \frac{\partial A_i}{\partial \theta_i} * \ddot{\theta}_i \qquad (2.11b)$$

Here θ_i is the joint variable.

B_i can be expressed as:

$$B_i = T_i * D_i \qquad (2.12)$$

$\dot{B}_i$ and $\ddot{B}_i$ can be also computed by performing the differentiation of equation (2.12):

$$\dot{B}_i = \dot{T}_i * D_i + T_i * \dot{D}_i \qquad (2.13a)$$

and

$$\ddot{B}_i = \ddot{T}_i * D_i + 2\dot{T}_i * \dot{D}_i + T_i * \ddot{D}_i \qquad (2.13b)$$

From the expression in equation (2.5), it is easy to obtain the following results:

$$\dot{D}_i = \dot{P}_i \qquad (2.14a)$$

and

$$\ddot{D}_i = \ddot{P}_i \qquad (2.14b)$$

These equations will be applied when the transformation matrices have to be developed for use, especially in the simulation and discussion regarding further extension of this method.

CHAPTER 3

DYNAMIC ANALYSIS AND MODELLING FOR FLEXIBLE MANIPULATORS

3.1 INTRODUCTION

Based on a kinematic analysis, one can obtain the expression for the manipulator system's kinetic energy and potential energy which are required for use in developing the system Lagrange's equations. The system potential energy comprises two parts: the potential energy due to gravity and the elastic potential energy(or the strain energy). As explained in the previcus chapter, all the different element potential energies due to gravity have to be expressed with reference to the unified coordinate system, which is the base coordinate. A summation over all the manipulator elements will provide the total system potential energy due to gravity. The same strategy is used to sum up the elastic potential energy and kinetic energy over all the elements of the system.

After the above procedure, the equations of motion can be derived using Lagrange's principle. The degrees of freedom of the system can be separated into two groups, i.e., the rigid body degrees of freedom and the elastic degrees of freedom.

3.2 THE SYSTEM KINETIC ENERGY AND ITS DERIVATIVES

In order to develop the manipulator finite element formulation in a generalized manner, all the expressions are managed to result in the form of multiplications of [4x4] transformation matrices. The absolute velocity of one point on the element is given by expression (2.7) and it is now possible to develop an expression for the kinetic energy of a point on the *jth* element of *i*th link

$$dk_{ij} = \frac{1}{2} Tr[\dot{\boldsymbol{r}}_{ij}(x_{ij}) * \dot{\boldsymbol{r}}_{ij}^{T}(x_{ij})]\, dm$$
$$= \frac{1}{2} Tr[\dot{\boldsymbol{r}}_{ij}(x_{ij}) * \dot{\boldsymbol{r}}_{ij}^{T}(x_{ij})]\, \mu dx_{ij} \qquad (3.1)$$

where μdx_{ij} is the differential mass of the point under consideration.

Since $Tr(ab^{T}) = Tr(ba^{T})$ and substituting $\dot{\boldsymbol{r}}_{ij}(x_{ij}) = T_i * \dot{\boldsymbol{r}}_{ij}^{i}(x_{ij}) + \dot{T}_i * \boldsymbol{r}_{ij}^{i}(x_{ij})$ from equation (2.7), the expanded form of equation (3.1) can be written as [11].

$$dk_{ij} = \frac{1}{2}\mu dx_{ij} Tr[\dot{T}_i * \boldsymbol{r}_{ij}^{i}(x_{ij}) * \boldsymbol{r}_{ij}^{iT}(x_{ij}) * \dot{T}_i^{T} + 2\dot{T}_i$$
$$* \boldsymbol{r}_{ij}^{i}(x_{ij}) * \dot{\boldsymbol{r}}_{ij}^{iT}(x_{ij}) * T_i^{T} + T_i * \dot{\boldsymbol{r}}_{ij}^{i}(x_{ij}) * \dot{\boldsymbol{r}}_{ij}^{iT}(x_{ij}) * T_i^{T}] \qquad (3.2)$$

By integrating the above expression over the entire link, one can obtain the total link kinetic energy. Here, μ is the

mass density of the link and the integration limits over x_{ij} are from 0 to l_{ij}. Summing over all the m links, the system total kinetic energy is obtained from:

$$K = \sum_{i=1}^{m} \sum_{j=1}^{n} \int_{0}^{l_{ij}} dk_{ij} \tag{3.3}$$

where n is the number of elements in each link. This is under the assumption that each link can be separated into n equal elements.

Noting that the deflection of a component is a collection of multiplication of shape functions and nodal variables, the expression for it in terms of two matrices multiplication is now introduced:

$$\boldsymbol{\phi}_{ij}^{i} = \boldsymbol{U}_{ij}^{i} N_{ij}^{T} \tag{3.4}$$

Here $\boldsymbol{\phi}_{ij}^{i}$ is the deflection vector of the jth element of link i with respect to coordinate system $O_i X_i Y_i Z_i$. The superscript i denotes reference to the coordinate system employed and here it is $O_i X_i Y_i Z_i$. Further,

$$\boldsymbol{\phi}_{ij}^{i} = [\phi_{ijx}^{i}(x_{ij}, t), \ \phi_{ijy}^{i}(x_{ij}, t), \ \phi_{ijz}^{i}(x_{ij}, t), \ 0] \tag{3.5}$$

$\boldsymbol{U_{ij}^i}$ is the nodal variable vector of the jth element of link i with respect to coordinate $O_iX_iY_iZ_i$, the quantities with respect to which are referred by the superscript i. Then,

$$\boldsymbol{U_{ij}^i} = [\boldsymbol{u_{ijx}^i}, \quad \boldsymbol{u_{ijy}^i}, \quad \boldsymbol{u_{ijz}^i}]_{1x12} \tag{3.6}$$

where

$$\boldsymbol{u_{ijx}^i} = [u_{ijx1}^i, \quad u_{ijx2}^i, \quad u_{ijx3}^i, \quad u_{ijx4}^i] \tag{3.7a}$$

$$\boldsymbol{u_{ijy}^i} = [u_{ijy1}^i, \quad u_{ijy2}^i, \quad u_{ijy3}^i, \quad u_{ijy4}^i] \tag{3.7b}$$

and

$$\boldsymbol{u_{ijz}^i} = [u_{ijz1}^i, \quad u_{ijz2}^i, \quad u_{ijz3}^i, \quad u_{ijz4}^i] \tag{3.7c}$$

N_{ij}^T is the shape function matrix of the jth element of link i and is given by,

$$N_{ij} = [N_{ijx} \quad N_{ijy} \quad N_{ijz}]_{4x12} \tag{3.8}$$

where

$$N_{ijx}=\begin{bmatrix} N_{ijx1} & N_{ijx2} & N_{ijx3} & N_{ijx4} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (3.9a)$$

$$N_{ijy}=\begin{bmatrix} 0 & 0 & 0 & 0 \\ N_{ijy1} & N_{ijy2} & N_{ijy3} & N_{ijy4} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (3.9b)$$

and

$$N_{ijz}=\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ N_{ijz1} & N_{ijz2} & N_{ijz3} & N_{ijz4} \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (3.9c)$$

The symbols N_{ijx1}, N_{ijx2}, N_{ijx3}, N_{ijx4}, N_{ijy1}, N_{ijy2}, N_{ijy3}, N_{ijy4}, N_{ijz1}, N_{ijz2}, N_{ijz3}, and N_{ijz4}, are the corresponding coefficients in equation (2.4), the symbol u_{ijx1}, u_{ijx2}, u_{ijx3}, u_{ijx4}, u_{ijy1}, u_{ijy2}, u_{ijy3}, u_{ijy4}, u_{ijz1}, u_{ijz2}, u_{ijz3}, and u_{ijz4} are the corresponding variables in equation (2.4).

The differentiation of equation (3.4) results in:

$$\dot{\phi}_{ij}^{i} = \dot{U}_{ij}^{i} N_{ij}^{T} \tag{3.10}$$

Defining C_{ij} as:

$$C_{ij} = [x_{ij} + (j-1)\, l_{ij} \quad 0 \quad 0 \quad 1] \tag{3.11}$$

then equation (2.3) can be arranged in the form:

$$r_{ij}^{i}(x_{ij}) = C_{ij}^{T} + \phi_{ij}^{iT} = C_{ij}^{T} + N_{ij} U_{ij}^{iT} \tag{3.12}$$

Similarly, the differentiation of equation (3.12) results in

$$\dot{r}_{ij}^{i}(x_{ij}) = N_{ij} \dot{U}_{ij}^{iT} \tag{3.13}$$

By introducing matrices F_{ij1}, F_{ij2} and F_{ij3} in the following manner,

$$F_{ij1} = \int_{0}^{l_{ij}} r_{ij}^{i}(x_{ij})\, r_{ij}^{iT}(x_{ij})\, \mu\, dx_{ij} = \int_{0}^{l_{ij}} \mu\, (C_{ij}^{T} + N_{ij} U_{ij}^{iT})$$
$$(C_{ij} + U_{ij}^{i} N_{ij}^{T})\, dx_{ij} = \int_{0}^{l_{ij}} \mu\, [C_{ij}^{T} C_{ij} + C_{ij}^{T} U_{ij}^{i} N_{ij}^{T}$$
$$+ N_{ij} U_{ij}^{iT} C_{ij} + N_{ij} U_{ij}^{iT} U_{ij}^{i} N_{ij}^{T}]\, dx_{ij} \tag{3.14}$$

$$F_{ij2} = \int_0^{l_{ij}} \boldsymbol{r}_{ij}^{i}(x_{ij})\, \dot{\boldsymbol{r}}_{ij}^{i\,T}(x_{ij})\, dm = \int_0^{l_{ij}} \mu\, (\boldsymbol{C}_{ij}^{T} + N_{ij}\boldsymbol{U}_{ij}^{i\,T})\, \dot{\boldsymbol{U}}_{ij}^{i} N_{ij}^{T} dx_{ij}$$

$$= \int_0^{l_{ij}} \mu\, (\boldsymbol{C}_{ij}^{T}\dot{\boldsymbol{U}}_{ij}^{i} N_{ij}^{T} + N_{ij}\boldsymbol{U}_{ij}^{i\,T}\dot{\boldsymbol{U}}_{ij}^{i} N_{ij}^{T})\, dx_{ij} \qquad (3.15)$$

and

$$F_{ij3} = \int_0^{l_{ij}} \dot{\boldsymbol{r}}_{ij}^{i}(x_{ij})\, \dot{\boldsymbol{r}}_{ij}^{i\,T}(x_{ij})\, dm = \int_0^{l_{ij}} \mu N_{ij}\dot{\boldsymbol{U}}_{ij}^{i\,T}\dot{\boldsymbol{U}}_{ij}^{i} N_{ij}^{T} dx_{ij} \qquad (3.16)$$

then, equation (3.3) becomes:

$$K = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} Tr\,[\dot{T}_i F_{ij1}\dot{T}_i^{T} + 2\dot{T}_i F_{ij2} T_i^{T} + T_i F_{ij3} T_i^{T}] \qquad (3.17)$$

Before constructing the Lagrange's equations, it may be noted that there exists following relations [11]:

$$\frac{\partial \dot{T}_i}{\partial \dot{\theta}_j} = \frac{\partial T_i}{\partial \theta_j} \qquad (3.18)$$

$$\frac{d}{dt}\left(\frac{\partial \dot{T}_i}{\partial \dot{\theta}_j}\right) = \frac{\partial \dot{T}_i}{\partial \theta_j} \qquad (3.19)$$

$$\frac{\partial \dot{T}_i}{\partial \dot{u}_j} = \frac{\partial T_i}{\partial u_j} \qquad (3.20)$$

and

$$\frac{d}{dt}\left(\frac{\partial \dot{T}_i}{\partial \dot{u}_j}\right) = \frac{\partial \dot{T}_i}{\partial u_j} \qquad (3.21)$$

It is also helpful in simplifying the result by noticing $\mathrm{Tr}(A)=\mathrm{Tr}(A^T)$ for any square matrix A.

After some simplification and cancellations of some terms like:

$$\frac{\partial T_i}{\partial \dot{\theta}_j} = 0 \qquad (3.22)$$

$$\frac{\partial T_i}{\partial \dot{u}_j} = 0 \qquad (3.23)$$

$$\frac{\partial \boldsymbol{U}_{rel}^{t}}{\partial \dot{\theta}_i} = 0 \qquad (3.24)$$

$$\frac{\partial U_{rs}^{t}}{\partial \theta_j} = 0 \tag{3.25}$$

and

$$\frac{\partial U_{rs}^{t}}{\partial \dot{u}_j} = 0 \tag{3.26}$$

the part of Lagrange's equation relative to the system kinetic energy corresponding to the rigid body variables results in

$$\frac{d}{dt}\frac{\partial K}{\partial \dot{\theta}_k} - \frac{\partial K}{\partial \theta_k} = \sum_{i=1}^{m}\sum_{j=1}^{n} Tr\left[\frac{\partial T_i}{\partial \theta_k}\left(F_{ij4}\dot{T}_i^T + F_{ij5}\ddot{T}_i^T + F_{ij6}T_i^T\right)\right] \tag{3.27}$$

Here

$$F_{ij4} = \int_0^{l_{ij}} \mu\left(2C_{ij}^{r}\dot{U}_{ij}^{i}N_{ij}^T + 2N_{ij}U_{ij}^{ir}\dot{U}_{ij}^{i}N_{ij}^T\right) dx_{ij} \tag{3.28}$$

$$F_{ij5} = F_{ij1} \tag{3.29}$$

and

$$F_{ij6} = \int_0^{l_{ij}} \mu\left(C_{ij}^{r}\ddot{U}_{ij}^{i}N_{ij}^T + N_{ij}U_{ij}^{ir}\ddot{U}_{ij}^{i}N_{ij}^T\right) dx_{ij} \tag{3.30}$$

Similarly, the part of Lagrange's equation corresponding to the elastic body variables is:

$$\frac{d}{dt}\frac{\partial K}{\partial \dot{u}_k} - \frac{\partial K}{\partial u_k} = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} Tr[\, 2\ddot{T}_i F_{ijk1} T_i^T + \dot{T}_i F_{ijk2} T_i^T + \dot{T}_i F_{ijk3} \dot{T}_i^T + T_i F_{ijk4} T_i^T + 2\frac{\partial T_i}{\partial u_k}[(\dot{F}_{ij1} - F_{ij2}) \dot{T}_i^T + F_{ij1}\ddot{T}_i^T + (\dot{F}_{ij2} - F_{ij3}) T_i^T]\,] \qquad (3.31)$$

Here

$$F_{ijk1} = \frac{\partial F_{ij2}}{\partial \dot{u}_k} = \int_0^{l_{ij}} \mu \left(C_{ij}^T \frac{\partial U_{ij}^1}{\partial u_k} N_{ij}^T + N_{ij} U_{ij}^{1T} \frac{\partial U_{ij}^1}{\partial u_k} N_{ij}^T \right) dx_{ij} \qquad (3.32)$$

$$F_{ijk2} = \int_0^{l_{ij}} 2\mu \left(C_{ij}^T \frac{\partial \dot{U}_{ij}^1}{\partial u_k} N_{ij}^T + 2N_{ij} \dot{U}_{ij}^{1T} \frac{\partial U_{ij}^1}{\partial u_k} N_{ij}^T + N_{ij} U_{ij}^{1T} \frac{\partial \dot{U}_{ij}^1}{\partial u_k} N_{ij}^T \right) dx_{ij} \qquad (3.33)$$

$$F_{ijk3} = \int_0^{l_{ij}} \mu \left(C_{ij}^T \frac{\partial U_{ij}^1}{\partial u_k} N_{ij}^T + N_{ij} U_{ij}^{1T} \frac{\partial U_{ij}^1}{\partial u_k} N_{ij}^T - N_{ij} \frac{\partial U_{ij}^{1T}}{\partial u_k} C_{ij} - N_{ij} \frac{\partial U_{ij}^{1T}}{\partial u_k} U_{ij}^1 N_{ij}^T \right) dx_{ij} \qquad (3.34)$$

and

$$F_{ijk4} = \int_0^{l_{ij}} \mu N_{ij} \left(\frac{\partial \dot{U}_{ij}^{1T}}{\partial u_k} \dot{U}_{ij}^1 N_{ij}^T + \frac{\partial U_{ij}^{1T}}{\partial u_k} \ddot{U}_{ij}^1 N_{ij}^T + \dot{U}_{ij}^{1T} \frac{\partial \dot{U}_{ij}^1}{\partial u_k} N_{ij}^T + \ddot{U}_{ij}^{1T} \frac{\partial U_{ij}^1}{\partial u_k} N_{ij}^T \right) dx_{ij} \qquad (3.35)$$

As the part of Lagrange's equations with respect to the system kinetic energy has been presented, another part refers to system potential energy is about to be developed to complete the Lagrange's equation.

3.3 THE SYSTEM POTENTIAL ENERGY AND ITS DERIVATIVES

The system's potential energy consists of two parts: the elastic potential energy and the gravitational potential energy.

The strain energy of element j on link i can be written as [26]:

$$(V_{ij})_e = \frac{1}{2}\int_0^{l_{ij}} [EI_y(\frac{\partial^2 \phi_{ijy}^i}{\partial x_{ij}^2})^2 + EI_z(\frac{\partial^2 \phi_{ijz}^i}{\partial x_{ij}^2})^2]\, dx_{ij} + \frac{1}{2}\int_0^{l_{ij}} GI_x(\frac{\partial \phi_{ij\beta}^i}{\partial x_{ij}})^2 dx_{ij} + \frac{1}{2}\int_0^{l_{ij}} EA(\frac{\partial \phi_{ijx}^i}{\partial x_{ij}})^2 dx_{ij} \quad (3.36)$$

where EA and GI_x are the structural rigidities of the link due to extension and torsion. EI_y and EI_z are the bending stiffness of the link in y, z directions. l_{ij} is a length integration domain and dx_{ij} is a differential length of element. ϕ_{ijx}^i, ϕ_{ijy}^i and ϕ_{ijz}^i are the linear displacements in x, y, and z directions and $\phi_{ij\beta}^i$ is the twist deflection along x axis.

The gravitational potential energy for element j on link

i is given by

$$(V_{ij})_g = \int_0^{l_{ij}} \mu \boldsymbol{g}^* T_i^* \boldsymbol{r}_{ij}^i(x_{ij})\, dx_{ij} \qquad (3.37)$$

The gravitational acceleration $\boldsymbol{g}$ is a vector which can have components along x, y, z direction. If the base coordinate is set such that the z axis is pointing vertically up from the ground, then

$$\boldsymbol{g}=[0,\ 0,\ 9.8,\ 0]\ (m/s^2)$$

Thus, the system potential energy is obtained by summing the potential energy expressed in equation (3.36) and (3.37) over all the elements in the system. That is,

$$V=\sum_{i=1}^{m}\sum_{j=1}^{n}[(V_{ij})_e + (V_{ij})_g] \qquad (3.38)$$

From the expression for the system's potential energy, it is clear that no velocity components have been involved. Hence, the derivatives of the system potential energy with respect to joint velocity variables are zero. That is,

$$\frac{\partial V}{\partial \dot{\theta}_k} = 0 \qquad (3.39)$$

It may be noticed that in the expression for system potential energy, only the gravitational component of potential energy contains rigid body variables. Therefore, the derivatives of the elastic potential energy with respect to

rigid body variables vanished and the following equation is obtained:

$$\frac{\partial V}{\partial \theta_k} = \sum_{i=1}^{m} \sum_{j=1}^{n} \int_0^{l_{ij}} \mu g^* \frac{\partial T_i}{\partial \theta_k} r_{ij}^i(x_{ij})\, dx_{ij} \qquad (3.40)$$

where θ_k is the kth joint variable

Before evaluating the derivatives of the potential energy with respect to elastic variables, the system's potential energy is rearranged into a more compact form. In order to achieve this, the matrices K_x, K_y, K_z, K_β are introduced into the following derivations. Let

$$K_x = EA \int_0^{l_{ij}} \frac{\partial N_{ijx}^T}{\partial x_{ij}} \frac{\partial N_{ijx}}{\partial x_{ij}} dx_{ij} \qquad (3.41a)$$

$$K_y = EI_y \int_0^{l_{ij}} \frac{\partial^2 N_{ijy}^T}{\partial x_{ij}^2} \frac{\partial^2 N_{ijy}}{\partial x_{ij}^2} dx_{ij} \qquad (3.41b)$$

$$K_z = EI_z \int_0^{l_{ij}} \frac{\partial^2 N_{ijz}^T}{\partial x_{ij}^2} \frac{\partial^2 N_{ijz}}{\partial x_{ij}^2} dx_{ij} \qquad (3.41c)$$

and

$$K_\beta = GI_x \int_0^{l_{ij}} \frac{\partial N_{ij\beta}^T}{\partial x_{ij}} \frac{\partial N_{ij\beta}}{\partial x_{ij}} dx_{ij} \qquad (3.41d)$$

As a consequence, the following expressions can now be rewritten:

$$\frac{1}{2}\int_0^{l_{ij}} EA\left(\frac{\partial \phi_{ijx}^i}{\partial x_{ij}}\right)^2 dx_{ij} = \frac{1}{2} u_{ijx}^i K_x u_{ijx}^{i\tau} \qquad (3.42a)$$

$$\frac{1}{2}\int_0^{l_{ij}} EI_y\left(\frac{\partial^2 \phi_{ijy}^i}{\partial x_{ij}^2}\right)^2 dx_{ij} = \frac{1}{2} u_{ijy}^i K_y u_{ijy}^{i\tau} \qquad (3.42b)$$

$$\frac{1}{2}\int_0^{l_{ij}} EI_z\left(\frac{\partial^2 \phi_{ijz}^i}{\partial x_{ij}^2}\right)^2 dx_{ij} = \frac{1}{2} u_{ijz}^i K_z u_{ijz}^{i\tau} \qquad (3.42c)$$

and

$$\frac{1}{2}\int_0^{l_{ij}} GI_x\left(\frac{\partial \phi_{ij\beta}^i}{\partial x_{ij}}\right)^2 dx_{ij} = \frac{1}{2} u_{ij\beta}^i K_\beta u_{ij\beta}^{i\tau} \qquad (3.42d)$$

Let a stiffness matrix K' be constructed such that

$$K' = \begin{bmatrix} K_x & 0 & 0 & 0 \\ 0 & K_y & 0 & 0 \\ 0 & 0 & K_z & 0 \\ 0 & 0 & 0 & K_\beta \end{bmatrix} \qquad (3.43)$$

Further, let the system's potential energy be rewritten

in the form

$$V-\sum_{i=1}^{m}\sum_{j=1}^{n}\left[\left(\int_{0}^{l_{ij}}\mu\left[g_x,\ g_y,\ g_z,\ 0\right]*T_i\left(C_{ij}^{T}+N_{ij}U_{ij}^{iT}\right)dx_{ij}\right)+\frac{1}{2}\right.$$
$$\left.*\left(u_{ijx}^{i}K_x u_{ijx}^{iT}+u_{ijy}^{i}K_y u_{ijy}^{iT}+u_{ijz}^{i}K_z u_{ijz}^{iT}+u_{ij\beta}^{i}K_\beta u_{ij\beta}^{iT}\right)\right]=\sum_{i=1}^{m}\sum_{j=1}^{n}$$
$$\left[\left(\int_{0}^{l_{ij}}\mu\left[g_x,g_y,g_z,0\right]*T_i\left(C_{ij}^{T}+N_{ij}U_{ij}^{iT}\right)dx_{ij}\right)+\frac{1}{2}\overline{U}_{ij}^{i}K'\overline{U}_{ij}^{iT}\right] \quad (3.44)$$

Then, the differentiation of system's potential energy with respect to an arbitrary elastic variable is given by

$$\frac{\partial V}{\partial u_k}=\sum_{i=1}^{m}\sum_{j=1}^{n}\left[\int_{0}^{l_{ij}}\mu\left[g_x,\ g_y,\ g_z,\ 0\right]*\left(T_i N_{ij}\frac{\partial U_{ij}^{iT}}{\partial u_k}\right.\right.$$
$$\left.\left.+\frac{\partial T_i}{\partial u_k}\left(C_{ij}^{T}+N_{ij}U_{ij}^{iT}\right)\right)dx_{ij}+\overline{U}_{ij}^{i}K'\frac{\partial\overline{U}_{ij}^{iT}}{\partial u_k}\right] \quad (3.45)$$

where

$$\overline{U}_{ij}^{i}=\left[\ u_{ijx}^{i}\ u_{ijy}^{i}\ u_{ijz}^{i}\ u_{ij\beta}^{i}\right] \quad (3.46)$$

With the above, all the required parts for Lagrange's formulation have been prepared, and now the entire Lagrange's equation can be simply obtained by assembling the different expressions derived earlier.

3.4 LAGRANGE'S FORMULATION

As a result of equations (3.27) and (3.40), the Lagrangian equations for rigid body variables are expressed in the form

$$\frac{d}{dt}\frac{\partial K}{\partial \dot{\theta}_k} - \frac{\partial K}{\partial \theta_k} + \frac{\partial V}{\partial \theta_k} = \sum_{i=k}^{m}\sum_{j=1}^{n} [Tr[\frac{\partial T_i}{\partial \theta_k}(F_{ijk}\dot{T}_i^T + F_{ijs}\ddot{T}_i^T + F_{ijs}T_i^T)] + \int_0^{l_{ij}} \mu[g_x, g_y, g_z, 0] \frac{\partial T_i}{\partial \theta_k} \mathbf{r}_{ij}^i(x_{ij})\, dx_{ij}] = \tau_k, \quad k=1, \cdots m \qquad (3.47)$$

The generalized force for manipulator rigid body variables can be recognized according to the different type of joint the manipulator possesses. If the joint is revolute, the generalized force in Lagrange's equation is a torque exerted by the actuator. When the joint is prismatic, the generalized force in Lagrange's equation is the force exerted by the actuator. As mentioned in the earlier chapters, the revolute joint is adopted in this thesis in order to simplify the analysis for application to flexible robot manipulators. Therefore the generalized forces in equation (3.47) are the joint torques. The subscript k indicates the kth joint.

From equations (3.31) and (3.45), the dynamic equations corresponding to the elastic variables can be obtained. /

there are no external forces or moment exerted on the joints between the finite elements, the generalized force in this equation is zero.

$$\frac{d}{dt}\frac{\partial K}{\partial \dot{u}_i} - \frac{\partial K}{\partial u_k} + \frac{\partial V}{\partial u_k} = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} Tr[2\ddot{T}_i F_{ijk1} T_i^T + \dot{T}_i F_{ijk2} T_i^T + \dot{T}_i F_{ijk3} \dot{T}_i^T + T_i F_{ijk4} T_i^T + 2\frac{\partial T_i}{\partial u_k}[(\dot{F}_{ij1} - F_{ij2}^T + F_{ij2})\dot{T}_i^T + F_{ij1}\ddot{T}_i^T + (\dot{F}_{ij2} - F_{ij3})T_i^T]] + \sum_{i=1}^{m}\sum_{j=1}^{n}[\int_0^{l_{ij}} \mu g^* (T_i N_{ij} \frac{\partial U_{ij}^{i\,T}}{\partial u_k} + \frac{\partial T_i}{\partial u_k}(C_{ij}^T + N_{ij} U_{ij}^{i\,T}))\, dx_{ij} + \bar{U}_{ij}^{i} K' \frac{\partial \bar{U}_{ij}^{i\,T}}{\partial u_k}] = 0 \qquad (3.48)$$

From these Lagrange's equations, a flexible robot manipulator with revolute joints can be modelled for any dynamic analysis no matter how many links it possesses. One can also gain the required accuracy by choosing a proper number of finite elements to represent the link flexibility. An ideal procedure would be to solve the partial differential equations representing the distributed parameter all continuous dynamic system by certain numerical method. However, even if such methods are available, they may not be suitable for adoption in a PC based facility. Further, if one has to solve such a continuous system by an approximation to a discrete system with large number of elements, it will again lead to the same type of heavy computational requirements. These make a PC based software difficult to implement. That is why in most manipulator dynamic modelling an

optimal number of discrete elements are employed. This is done here also.

3.5 THE DEVELOPMENT OF FINITE ELEMENT ALGORITHM

Before a robot manipulator system design is carried out, the range of responses of the manipulator links due to certain design input have to be figured out to check if the designed system meets all the expected requirements. Then adjustments to the design can be carried out by varying the values of the system parameters until the system response satisfies the expected performance requirements.

The algorithm that is developed for use in the process of design may be outlined through a flow chart detailed in Fig 3.1. The step by step procedure can be restated as follows:

step 1: The system is initialized by transferring the initial conditions and system parameters to the dynamic modelling and analysis program.

step 2: For each finite element, the quantities F_{1j}, F_{1jk}, F_{1jkl}, F_{1jklm}, F_{1jklmn}, F_{1jklmn}, K' are obtained through integration and partial differentiation as set out earlier, while $\dot{T}_1$, $\dot{T}_1$ are gained through differentiation.

step 3: The required expressions for all the elements are assembled as given in equations (3.47) and (3.48).

step 4: The proposed numerical method is employed such that the equations arranged in step 3 are solved.

Step 5: If the loop is not ended, it will go to step 2 to repeat the procedure for the response corresponding to the next period.

Step 6: When the analysis time has reached the prescribed end time, the program is terminated.

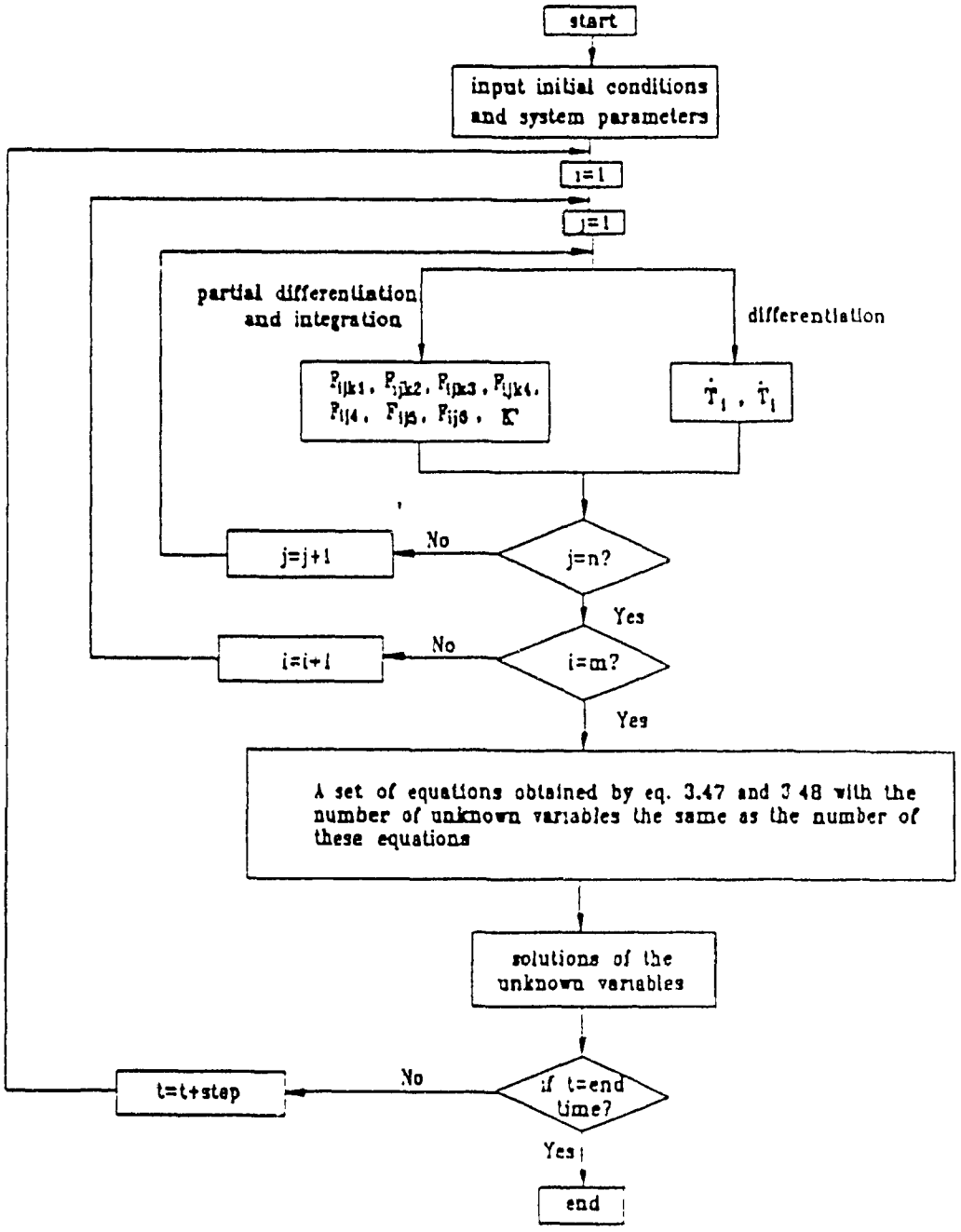


Fig. 3.1 The Flow Chart Based on the Finite Element Approach

CHAPTER 4

DYNAMIC SIMULATION FOR FLEXIBLE ROBOT MANIPULATOR VALIDATION

4.1 INTRODUCTION

Before a certain mathematical model can be relied upon in the design process, some procedures must be carried out to check its validity and reliability. A dynamic simulation, which imitates the behaviour of flexible robot manipulator by means of the given mathematical model, is one of these procedures that are often used before experimentation. Therefore, the following simulations will be performed to check the validity of the proposed model.

As the first step, a two-link system hanging freely under gravity is simulated. This is the same example that was used by Usoro [10]. A small initial deflection from its balance or equilibrium position, which corresponds to θ_1 =-90 degrees and θ_2 =5 degrees is given. The initial conditions and all the parameter values of the model used in this simulation are chosen to be the same as those in the simulation presented by Usoro [10]. The reason is that it will be easy to compare the results between two models with the same parameter values in order to establish the applicability of the procedure developed in this thesis.

As the first simulation described above is discussed

without the presence of any joint torque (as presented by Usoro [10]) in the links, a second simulation, which takes into account joint torque on a one-link system is also examined in order to establish the procedure developed in this thesis for other applications. By these two simulations that can be extended to represent general type of manipulator applications, one can conclude the proposed procedure has merit.

Generally speaking, the primary shape of the response fluctuations in the simulation should be in agreement with that of the real system that was designed. However, some factors from both simulation and design may affect the results. They are:

(i) The dynamic model is set up by finite element method. The different number of finite elements chosen will lead to different magnitudes of accuracy of the response.

(ii) The different numerical integration scheme adopted is another source that could affect the response. In this thesis, the Runge-Kutta method is chosen to solve the dynamic equations due to its self-starting property as pointed out in many applications [21][22]. It is believed that this numerical integration procedure will also give sufficient accuracy to the results.

(iii) In order to implement the control scheme for a real robot system, the control aspects may be designed in terms of engineering and economic viability. Hence, some key factors

influencing manipulator dynamic performance may get neglected.

(iv) The parameter values of the actual system may be slightly different from those used in the design process.

Only through simulations, the correctness and validity of the dynamic model presented in this thesis can be assessed. The real system can then be designed relying on the simulation results presented later in this chapter.

4.2 THE SIMULATION OF TWO-LINK MANIPULATOR

In order to be identical with Usoro's[10] example for comparison purposes, the same two-link flexible robot manipulator is investigated in this simulation. As shown in Fig. 4.1, the first link is initially freely hanging under gravity without any payload at the tip. The properties for the two-link manipulator are listed in Table 4.1 by Usoro[10].

Consider the two-link manipulator as depicted in Fig. 4.2. The manipulator consists of link 1 and link 2 which are divided into two finite elements of equal length *l* for sake of both computational efficiency and response accuracy needed in design applications. In this manner, a total number of 18 variables will be utilized (16 elastic variables and 2 rigid body variables) while planar or two dimensional motion is under discussion. However, by certain simplification procedures, the number of variables for such 2-link devices can be reduced to 10 for efficient computational algorithm

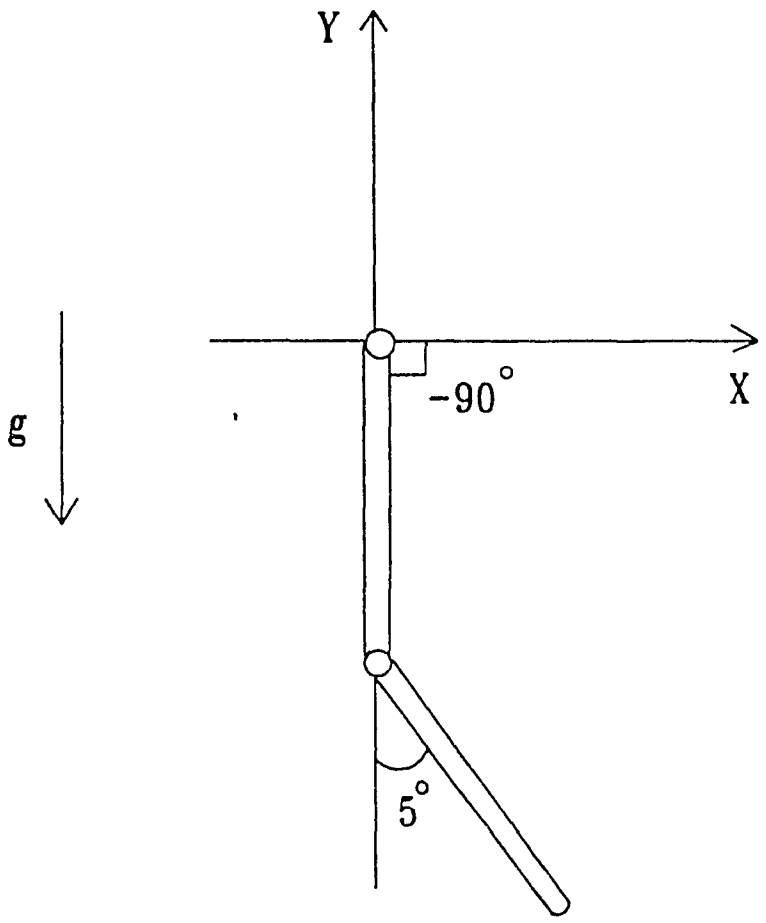


Fig. 4.1 Initial Conditions for a Two-link Manipulator Simulation

Table 4.1 Two-link manipulator properties

$l_1=l_2=l$	1m
$I_1=I_2$	5×10^{-9} m^4
$\mu_1=\mu_2$	5 kg/m
$E_1=E_2$	2×10^{11} N/m^2
$\tau_1=\tau_2$	0 N m
$\theta_1(0)$	-90 deg
$\theta_2(0)$	5 deg

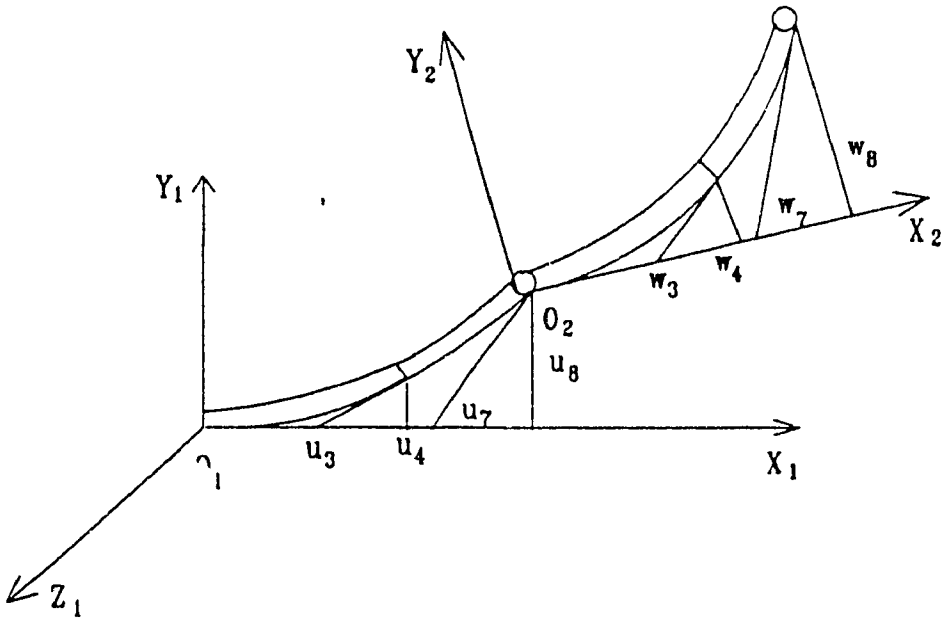


Fig. 4.2 Schematic of a Two-link Flexible Manipulator

development.

In the study of a linkage system as the one shown in Fig. 4.3, the elastic variables at the end of one element should describe the same elastic configuration as those corresponding to the starting point of the following element. Therefore, u_5, u_6 in the Fig. 4.3 could be replaced by u_3, u_4. In this simulation, a net number of 10 variables are employed after such simplification steps through elimination of 2 mid-link variables in each link.

Considering the fact that the joint 1 is fixed to the base with link 1 being fixed to joint 1, the joint 1 is constrained to have zero translational displacement and only an angular displacement of θ_1 with respect to the axis OX. This constrains elastic variables u_1, u_2 to be zero, i.e., $u_1(t)=0$, $u_2(t)=0$. The joint 2 has also the same features. It is constrained to have only the angular displacement of θ_2 relative to the first and constrains the elastic variables w_1, w_2 of the second link to be zero, i.e., $w_1(t)=0$, $w_2(t)=0$. According to these constraints, u_1, u_2, u_5, u_6 of link 1 and w_1, w_2, w_5, w_6 of link 2 could be eliminated in the simulation without affecting the results.

The transformation matrices T_1, D_1, A_1 given in chapter 2 for the first link in this system take the form:

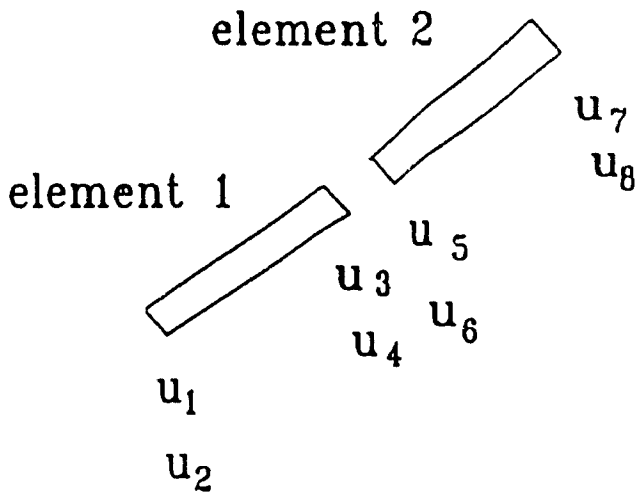


Fig. 4.3 Finite Element Representation

$$A_1 = T_1 = \begin{bmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

and

$$D_1 = \begin{bmatrix} 1 & 0 & 0 & 2l \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & -u_7 & 0 & 0 \\ u_7 & 0 & 0 & u_8 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -u_7 & 0 & 2l \\ u_7 & 1 & 0 & u_8 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.2}$$

The corresponding transformation matrices for link 2 are:

$$A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & 0 \\ S_2 & C_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

$$D_2 = \begin{bmatrix} 1 & -w_7 & 0 & 2l \\ w_7 & 1 & 0 & w_8 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.4}$$

and

$$T_2 = A_1 D_1 A_2 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -u_7 & 0 & 2l \\ u_7 & 1 & 0 & u_8 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} & 0 & t_{14} \\ t_{21} & t_{22} & 0 & t_{24} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.5}$$

where

$$t_{11} = c_{12} - s_{12} u_7 \tag{4.6a}$$

$$t_{12} = -s_{12} - c_{12} u_7 \tag{4.6b}$$

$$t_{14} = 2l c_1 - s_1 u_8 \tag{4.6c}$$

$$t_{21} = s_{12} + c_{12} u_7 \tag{4.6d}$$

$$t_{22} = c_{12} - s_{12} u_7 \tag{4.6e}$$

In the above equations, the following simplified forms of expressions have been employed:

$$c_1 = \cos\theta_1 \tag{4.7a}$$

$$c_2 = \cos\theta_2 \tag{4.7b}$$

$$s_1 = \sin\theta_1 \tag{4.7c}$$

$$s_2 = \sin\theta_2 \tag{4.7d}$$

$$s_{12} = \sin(\theta_1 + \theta_2) \tag{4.7e}$$

$$c_{12} = \cos(\theta_1 + \theta_2) \tag{4.7f}$$

The first order and second order differentiation of matrix T_1 are given by:

$$\dot{T}_1 = \begin{bmatrix} -s_1 & -c_1 & 0 & 0 \\ c_1 & -s_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \dot{\theta}_1 \tag{4.8}$$

and

$$\ddot{T}_1 = \begin{bmatrix} -c_1 & s_1 & 0 & 0 \\ -s_1 & -c_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \dot{\theta}_1^2 + \begin{bmatrix} -s_1 & -c_1 & 0 & 0 \\ c_1 & -s_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \ddot{\theta}_1 \tag{4.9}$$

The first order differentiation of matrix T_2 is:

$$\dot{T}_2 = \begin{bmatrix} t'_{11} & t'_{12} & 0 & t'_{14} \\ t'_{21} & t'_{22} & 0 & t'_{24} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.10}$$

where

$$t'_{11} = (-s_1\dot{\theta}_1 - c_1u_7\dot{\theta}_1 - s_1\dot{u}_7)c_2 - (c_1 - s_1u_7)s_2\dot{\theta}_2 - (c_1u_7 + s_1)c_2\dot{\theta}_2 - (c_1\dot{u}_7 - s_1u_7\dot{\theta}_1 + c_1\dot{\theta}_1)s_2$$
$$= -(\dot{\theta}_1 + \dot{\theta}_2)[s_{12} + u_7c_{12}] - \dot{u}_7s_{12} \qquad (4.11)$$

$$t'_{12} = -c_{12}(\dot{\theta}_1 + \dot{\theta}_2) + s_{12}(\dot{\theta}_1 + \dot{\theta}_2)u_7 - c_{12}\dot{u}_7$$
$$= \dot{\theta}_1(s_{12}u_7 - c_{12}) + \dot{\theta}_2(s_{12}u_7 - c_{12}) - c_{12}\dot{u}_7 \qquad (4.12)$$

$$t'_{14} = -2ls_1\dot{\theta}_1 - s_1\dot{u}_8 - c_1u_8\dot{\theta}_1$$
$$= -\dot{\theta}_1(2ls_1 + c_1u_8) - s_1\dot{u}_8 \qquad (4.13)$$

$$t'_{21} = c_{12}(\dot{\theta}_1 + \dot{\theta}_2) - s_{12}(\dot{\theta}_1 + \dot{\theta}_2)u_7 + c_{12}\dot{u}_7$$
$$= \dot{\theta}_1(c_{12} - s_{12}u_7) + \dot{\theta}_2(c_{12} - s_{12}u_7) + c_{12}\dot{u}_7 \qquad (4.14)$$

$$t'_{22} = -s_{12}(\dot{\theta}_1 + \dot{\theta}_2) - c_{12}(\dot{\theta}_1 + \dot{\theta}_2)u_7 - s_{12}\dot{u}_7$$
$$= -\dot{\theta}_1(s_{12} + c_{12}u_7) - \dot{\theta}_2(s_{12} + c_{12}u_7) - s_{12}\dot{u}_7 \qquad (4.15)$$

and

$$t'_{24} = 2lc_1\dot{\theta}_1 - s_1u_8\dot{\theta}_1 + c_1\dot{u}_8$$
$$= \dot{\theta}_1(2lc_1 - s_1u_8) + c_1\dot{u}_8 \qquad (4.16)$$

The second order differentiation of T is:

$$\ddot{T}_2 = \begin{bmatrix} t''_{11} & t''_{12} & 0 & t''_{14} \\ t''_{21} & t''_{22} & 0 & t''_{24} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (4.17)$$

where

$$t''_{11} = -(\ddot{\theta}_1+\ddot{\theta}_2)(s_{12}+u_7c_{12}) - (\dot{\theta}_1+\dot{\theta}_2)[c_{12}(\dot{\theta}_1+\dot{\theta}_2) + \dot{u}_7c_{12} - s_{12}u_7(\dot{\theta}_1+\dot{\theta}_2)] - \ddot{u}_7s_{12} - \dot{u}_7c_{12}(\dot{\theta}_1+\dot{\theta}_2)$$
$$= -\ddot{\theta}_1(s_{12}+u_7c_{12}) - \ddot{\theta}_2(s_{12}+u_7c_{12}) - \ddot{u}_7s_{12} - (\dot{\theta}_1+\dot{\theta}_2)^2(c_{12}-s_{12}u_7) - 2(\dot{\theta}_1+\dot{\theta}_2)\dot{u}_7c_{12} \qquad (4.18)$$

$$t''_{12} = \ddot{\theta}_1(s_{12}u_7 - c_{12}) + \ddot{\theta}_2(s_{12}u_7 - c_{12}) + (\dot{\theta}_1+\dot{\theta}_2)[c_{12}u_7(\dot{\theta}_1+\dot{\theta}_2) + s_{12}\dot{u}_7 + s_{12}(\dot{\theta}_1+\dot{\theta}_2)] + s_{12}\dot{u}_7(\dot{\theta}_1+\dot{\theta}_2) - c_{12}\ddot{u}_7$$
$$= \ddot{\theta}_1(s_{12}u_7 - c_{12}) + \ddot{\theta}_2(s_{12}u_7 - c_{12}) - \ddot{u}_7c_{12} + (\dot{\theta}_1+\dot{\theta}_2)^2(c_{12}u_7+s_{12}) + 2(\dot{\theta}_1+\dot{\theta}_2)s_{12}\dot{u}_7 \qquad (4.19)$$

$$t''_{14} = -\ddot{u}_8s_1 - \dot{\theta}_1^2(2lc_1 - s_1u_8) - 2\dot{\theta}_1c_1\dot{u}_8 \qquad (4.20)$$

$$t''_{21} = \ddot{\theta}_1(c_{12}-s_{12}u_7) + \ddot{\theta}_2(c_{12}-s_{12}u_7) + (\dot{\theta}_1+\dot{\theta}_2)[-s_{12}(\dot{\theta}_1+\dot{\theta}_2) - c_{12}u_7(\dot{\theta}_1+\dot{\theta}_2) - s_{12}\dot{u}_7] + c_{12}\ddot{u}_7 - s_{12}\dot{u}_7(\dot{\theta}_1+\dot{\theta}_2)$$
$$= \ddot{\theta}_1(c_{12}-s_{12}u_7) + \ddot{\theta}_2(c_{12}-s_{12}u_7) + \ddot{u}_7c_{12} + (\dot{\theta}_1+\dot{\theta}_2)^2(-s_{12}-c_{12}u_7) - 2s_{12}\dot{u}_7(\dot{\theta}_1+\dot{\theta}_2) \qquad (4.21)$$

$$t''_{22} = -\ddot{\theta}_1(s_{12}+c_{12}u_7) - \ddot{\theta}_2(s_{12}+c_{12}u_7) - (\dot{\theta}_1+\dot{\theta}_2)[(c_{12}(\dot{\theta}_1+\dot{\theta}_2) - s_{12}u_7(\dot{\theta}_1+\dot{\theta}_2) + c_{12}\dot{u}_7] - s_{12}\ddot{u}_7 - c_{12}\dot{u}_7(\dot{\theta}_1+\dot{\theta}_2)$$
$$= -\ddot{\theta}_1(s_{12}+c_{12}u_7) - \ddot{\theta}_2(s_{12}+c_{12}u_7) - \ddot{u}_7s_{12} - (\dot{\theta}_1+\dot{\theta}_2)^2(c_{12}-s_{12}u_7) - 2c_{12}\dot{u}_7(\dot{\theta}_1+\dot{\theta}_2) \qquad (4.22)$$

$$t''_{24} = \ddot{\theta}_1 (2lc_1 - s_1 u_8) + \dot{\theta}_1 (-2ls_1\dot{\theta}_1 - c_1 u_8 \dot{\theta}_1 - s_1 \dot{u}_8) + c_1 \ddot{u}_8 - s_1 \dot{u}_8 \dot{\theta}_1 = \ddot{\theta}_1 (2lc_1 - s_1 u_8) + \ddot{u}_8 c_1 - \dot{\theta}_1^2 (2ls_1 + c_1 u_8) - 2s_1 \dot{\theta}_1 \dot{u}_8 \qquad (4.23)$$

The Hermitian polynomials are employed for the shape function N_{ijy}. As the simulation restricts the motion in the plane OXY, the shape function describing the displacement along axis z need not be considered. Since there is no torque acting about the link, the effect of the torsion about x axis is so insignificant compared to that due to bending and therefore can be neglected. It is also suggested that the deflection in the axial direction can be considered to be of little significance in relation to the magnitudes of lateral deflections [15], stating that there is very little coupling effect. Hence, an assumption of axial rigidity is employed in the present simulation. As a result, the shape functions are now given by:

$$N_{y1} = N_{1jy1} = x - 2\frac{x^2}{l} + \frac{x^3}{l^2} \qquad (4.24a)$$

$$N_{y2} = N_{1jy2} = 1 - 3\frac{x^2}{l^2} + 2\frac{x^3}{l^3} \qquad (4.24b)$$

$$N_{y3} = N_{1jy3} = -\frac{x^2}{l} + \frac{x^3}{l^2} \qquad (4.24c)$$

and

$$N_{y4} = N_{1jy4} = 3\frac{x^2}{l^2} - 2\frac{x^3}{l^3} \qquad (4.24d)$$

According to the definition of equation (3.11), the vectors $\mathbf{C}_{ij}$ for element 1 and 2 of link 1 are given respectively by the following:

$$\boldsymbol{C}_{11} = [x \;\; 0 \;\; 0 \;\; 1] \tag{4.25}$$

and

$$\boldsymbol{C}_{12} = [x+l \;\; 0 \;\; 0 \;\; 1] \tag{4.26}$$

Similarly, the vectors for element 1 and 2 of link 2 are respectively given by following expressions:

$$\boldsymbol{C}_{21} = [x \;\; 0 \;\; 0 \;\; 1] \tag{4.27}$$

and

$$\boldsymbol{C}_{22} = [x+l \;\; 0 \;\; 0 \;\; 1] \tag{4.28}$$

As mentioned in section 4.2, for element 1 on the first link, u, w are zero. So $\boldsymbol{r}_{11}^{1}$ becomes:

$$\boldsymbol{r}_{11}^{1} = \boldsymbol{C}_{11}^{T} + N_{11}\boldsymbol{U}_{11}^{1T} = [x, \;\; N_{y3}u_3 + N_{y4}u_4, \;\; 0, \;\; 1]^{T} \tag{4.29}$$

Let u_3, u_4 replace u_5, u_6, then $\boldsymbol{r}_{12}^1$ for the element 2 on the first link is:

$$\boldsymbol{r}_{12}^1 = \boldsymbol{C}_{12}^T + N_{12}\boldsymbol{U}_{12}^{1T} = [x+l, \quad N_{y1}u_3+N_{y2}u_4+N_{y3}u_7+N_{y4}u_8, \quad 0, \quad 1]^T \qquad (1.30)$$

Similar to the link 1, element 1 and element 2 of link 2 have following equations respectively:

$$\boldsymbol{r}_{21}^1 = \boldsymbol{C}_{21}^T + N_{21}\boldsymbol{U}_{21}^{2T} = [x, \quad N_{y3}w_3+N_{y4}w_4, \quad 0, \quad 1]^T \qquad (1.31)$$

and

$$\boldsymbol{r}_{22}^1 = \boldsymbol{C}_{22}^T + N_{22}\boldsymbol{U}_{22}^{2T} = [x+l, \quad N_{y1}w_3+N_{y2}w_4+N_{y3}w_7+N_{y4}w_8, \quad 0, \quad 1] \qquad (1.32)$$

Because only simulation of a two-link manipulator is presented, the new symbols have been introduced instead of those in the derivations. The variables u are the variables of first link and the w indicates the variables of second link.

Following the strategy developed in equations (3.17) to (3.48), the system's kinetic energy and potential energy can now be obtained. After arranging the Lagrange's equation in the form of equation (1.1), appropriate numerical method is utilized to solve the dynamic equations.

The numerical method used in this simulation is the 4th-order Runge-Kutta method, and the strategy of the method is illustrated in Table 4.2. The Runge-Kutta method of computation is popular since it is self-starting and results in good accuracy [21]. In the method, the second-order differential equation is first reduced to two first-order differential equations. Here a differential equation of the form

$$\ddot{x} = f(x, \dot{x}, t) \tag{4.33}$$

is rewritten as first order sets:

$$\dot{x} = y \tag{4.34}$$

$$\dot{y} = f(x, y, t) \tag{4.35}$$

and then the computation is performed as described in Table 4.2.

The quantities in Table 4.2 are used in the following recurrence formula [22]:

$$x_{i+1} = x_i + \frac{h}{6}[Y_1 + 2Y_2 + 2Y_3 + Y_4] \tag{4.36}$$

$$y_{i+1} = y_i + \frac{h}{6}[F_1 + 2F_2 + 2F_3 + F_4] \tag{4.37}$$

Table 4.2 4th order Runge-kutta method

t	x	y=x	f =$\dot{y}$=$\ddot{x}$
$T_1=t_1$	$X_1=x_1$	$Y_1=y_1$	$F_1=f(T_1,X_1,Y_1)$
$T_2=t_1+\frac{h}{2}$	$X_2=x_1+Y_1\frac{h}{2}$	$Y_2=y_1+F_1\frac{h}{2}$	$F_2=f(T_2,X_2,Y_2)$
$T_3=t_1+\frac{h}{2}$	$X_3=x_1+Y_2\frac{h}{2}$	$Y_3=y_1+F_2\frac{h}{2}$	$F_3=f(T_3,X_3,Y_3)$
$T_4=t_1+h$	$X_4=x_1+Y_3 h$	$Y_4=y_1+F_3 h$	$F_4=f(T_4,X_4,Y_4)$

where h is step length.

The sampling frequency is chosen according to the system under discussion. When the sampling frequency is smaller than a critical frequency, singularity occurs. At the same time, if the sampling frequency is much larger than the critical frequency, the accumulating errors grow with increasing computational time. Therefore, a proper sampling frequency should be chosen to off-set the above disadvantages and to obtain the correct response of the system. By certain amount of trial and error calculations, starting from larger time steps, the step interval in the program loop of the simulation is chosen as 0.0005 seconds. Under such a sampling rate, a set of plots for positions θ_1, θ_2, u_3, u_4, u_7, u_8, w_3, w_4, w_7 and w_8 with their velocities $\dot{\theta}_1$, θ_2, $\dot{u}_3$, $\dot{u}_4$, $\dot{u}_7$, $\dot{u}_8$, $\dot{w}_3$, $\dot{w}_4$, $\dot{w}_7$ and $\dot{w}_8$ are obtained as illustrated in Fig. 4.4 to 4.23.

From plots of θ_1 and θ_2, it seems that both of them are undergoing a vibrating or oscillating motion. Both motions do not keep a constant amplitude simply because of the coupling effects between link 1 and link 2.

The flexural responses are featured in the two modes of oscillations are: one being a relatively fast vibration mode with high frequency(result of presence of large stiffness) and the other a relatively slow vibration mode. The slow vibration mode shows the similar oscillation shape as that of corresponding joint deflection. When compared with the results

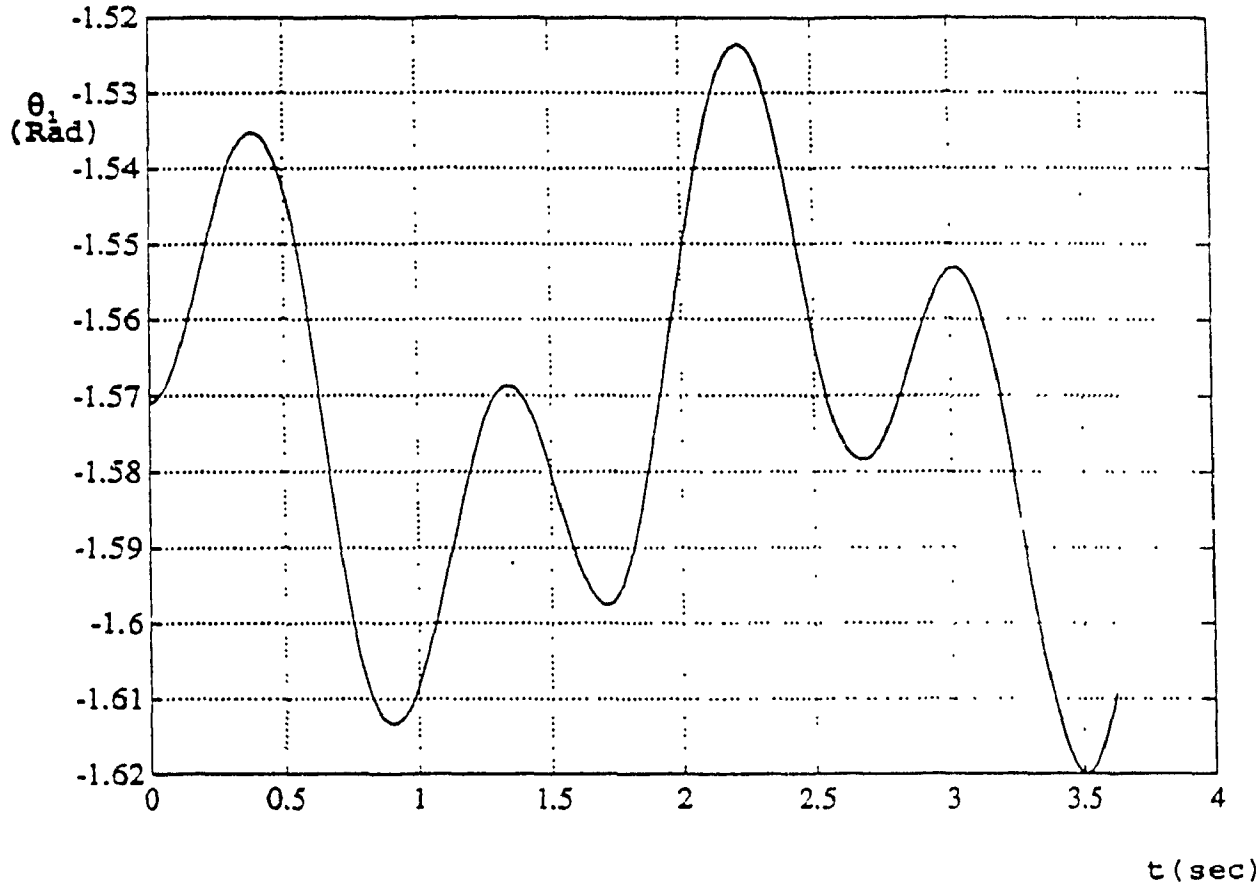


Fig. 4.4 The Angular Deflection of Joint 1 in the First Simulation

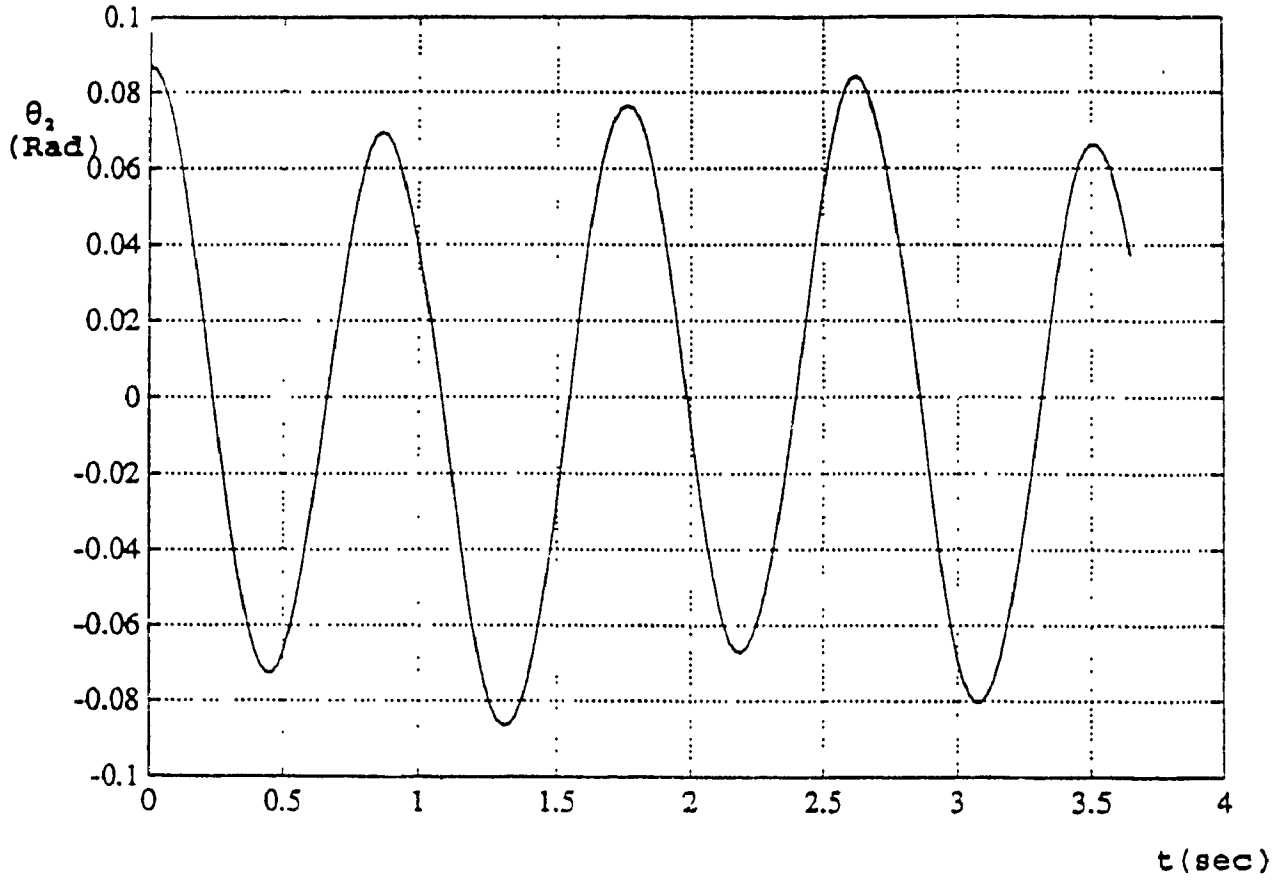


Fig. 4.5 The Angular Deflection of Joint 2 in the First Simulation

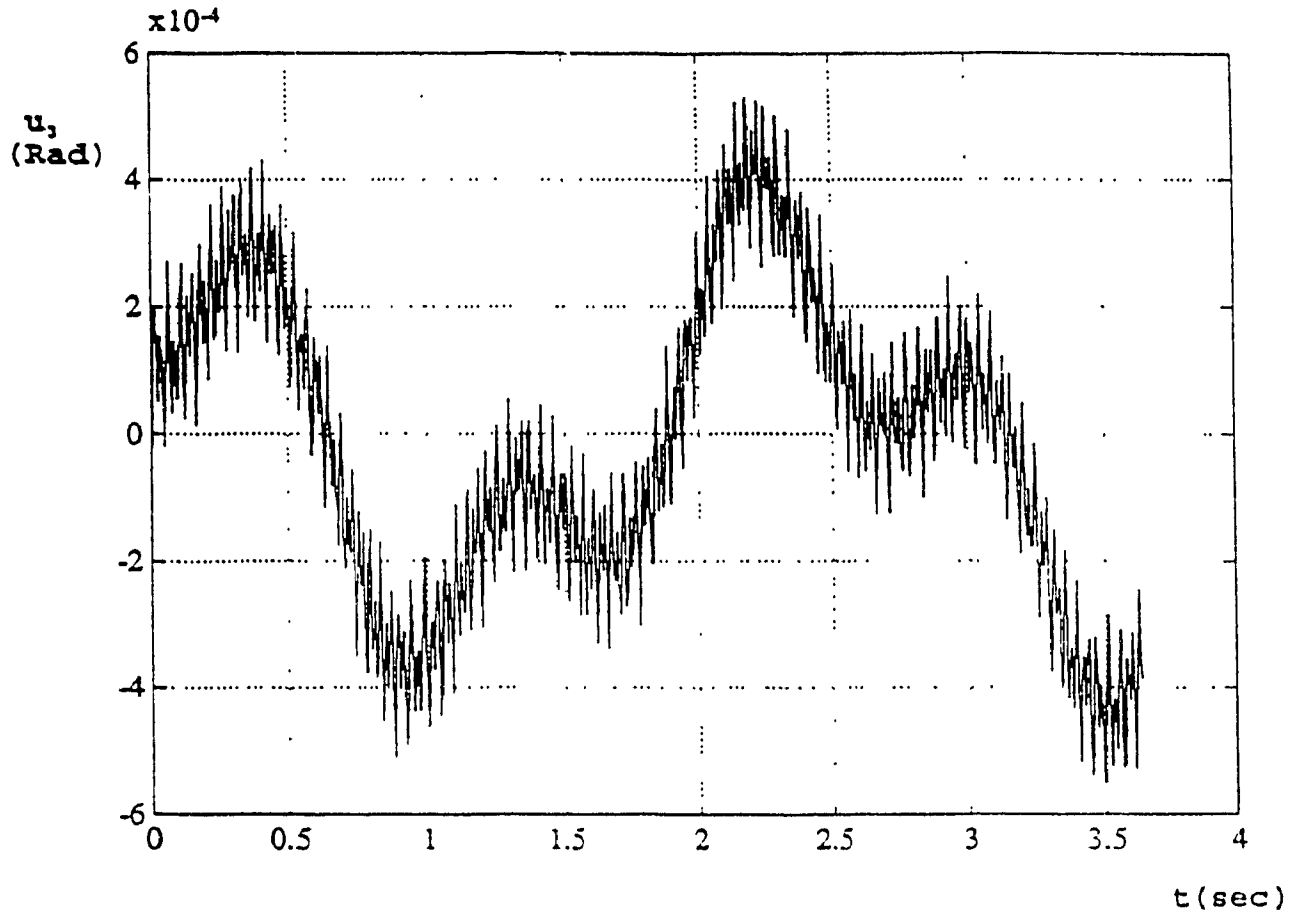


Fig. 4.6 Flexural Slope at Link 1 Midpoint in the First Simulation

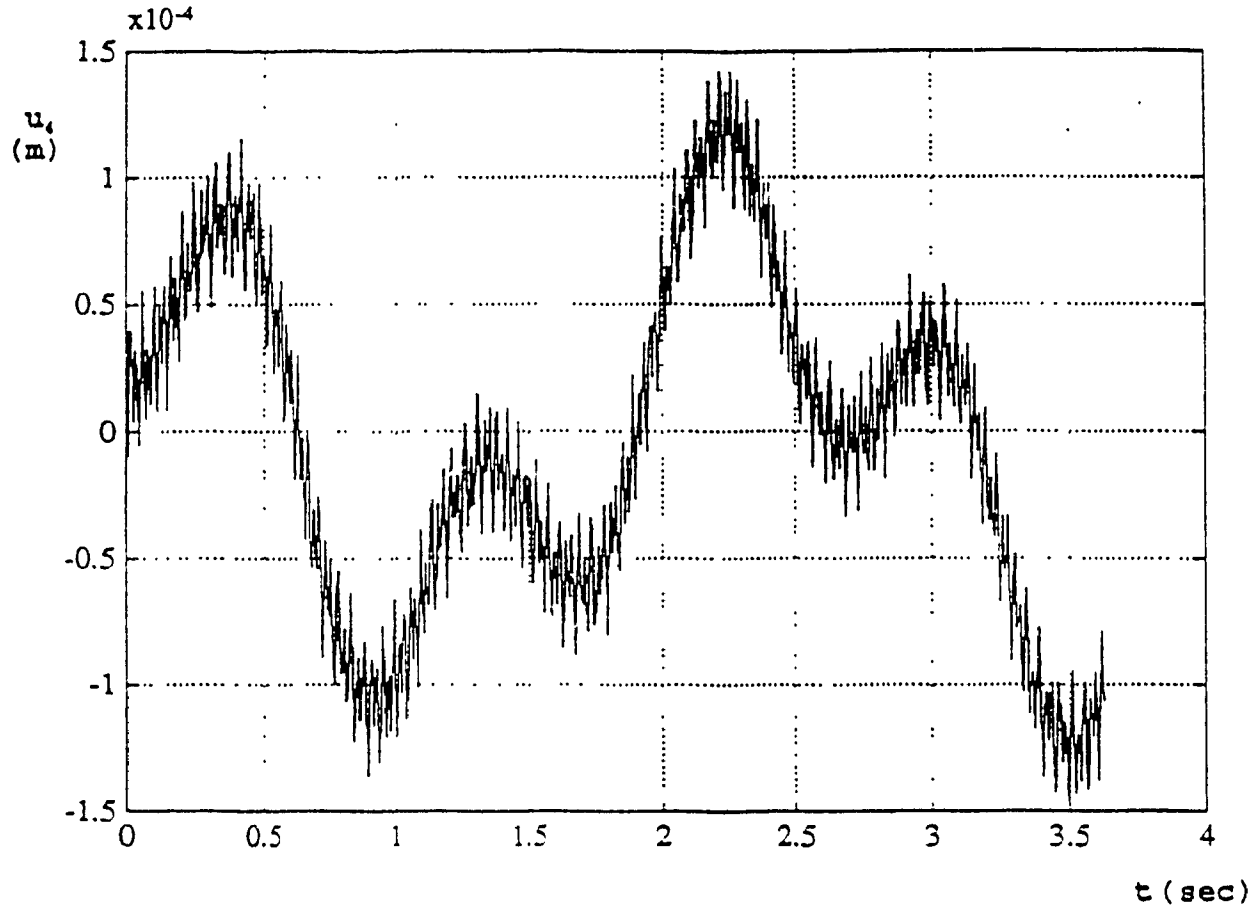


Fig. 4.7 Flexural Displacement at Link 1 Midpoint in the First Simulation

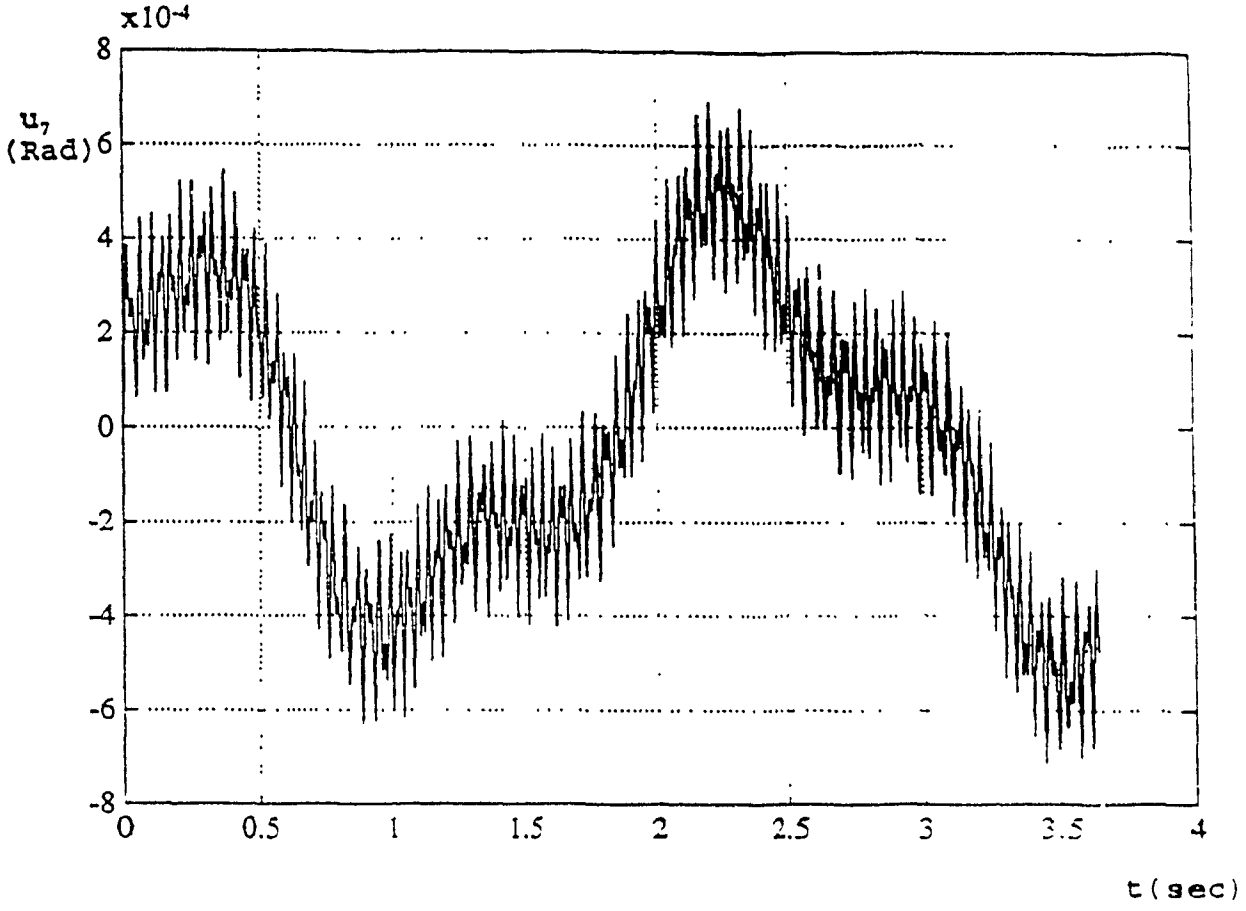


Fig. 4.8 Flexural Slope at Link 1 Tip in the First Simulation

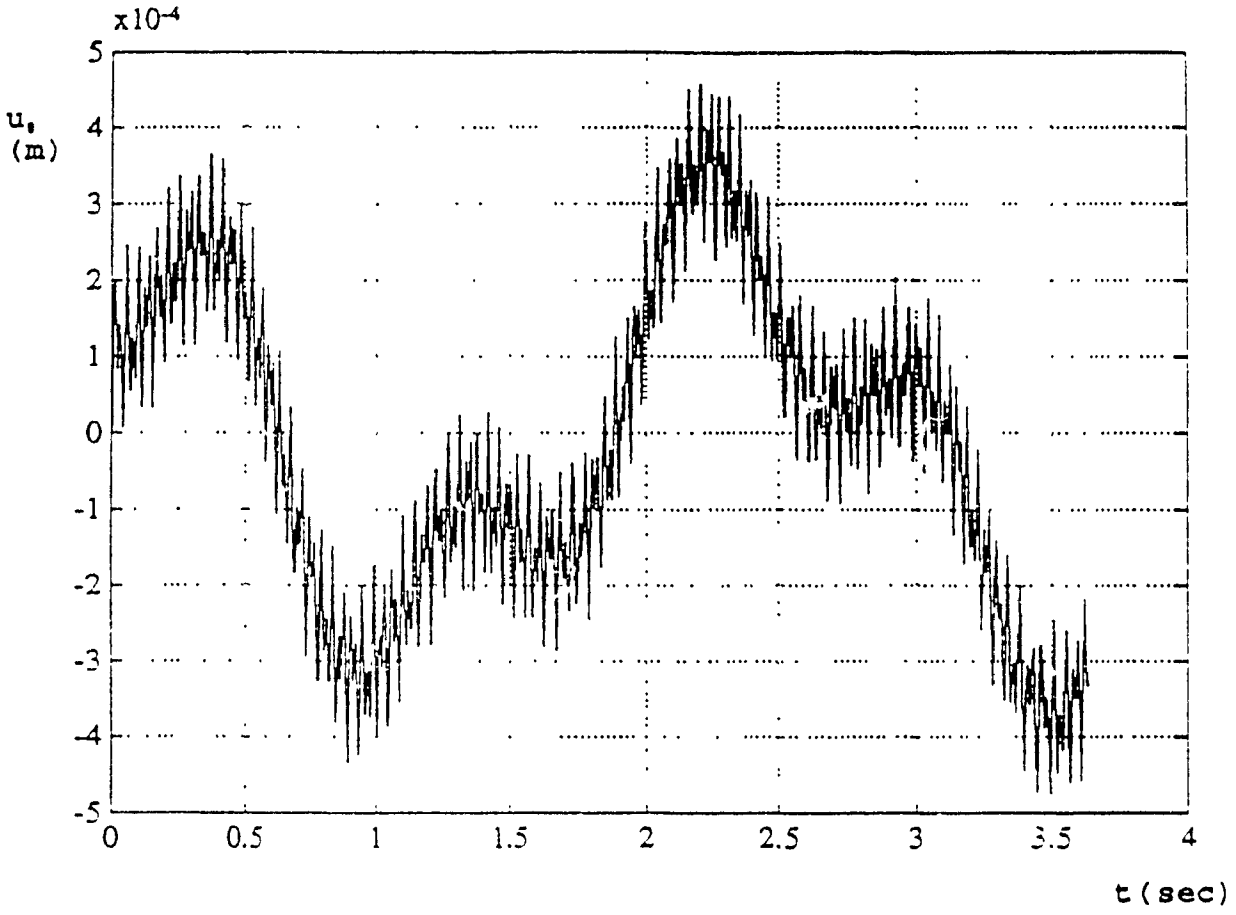


Fig. 4.9 Flexural Displacement of Link 1 Tip
in the First Simulation

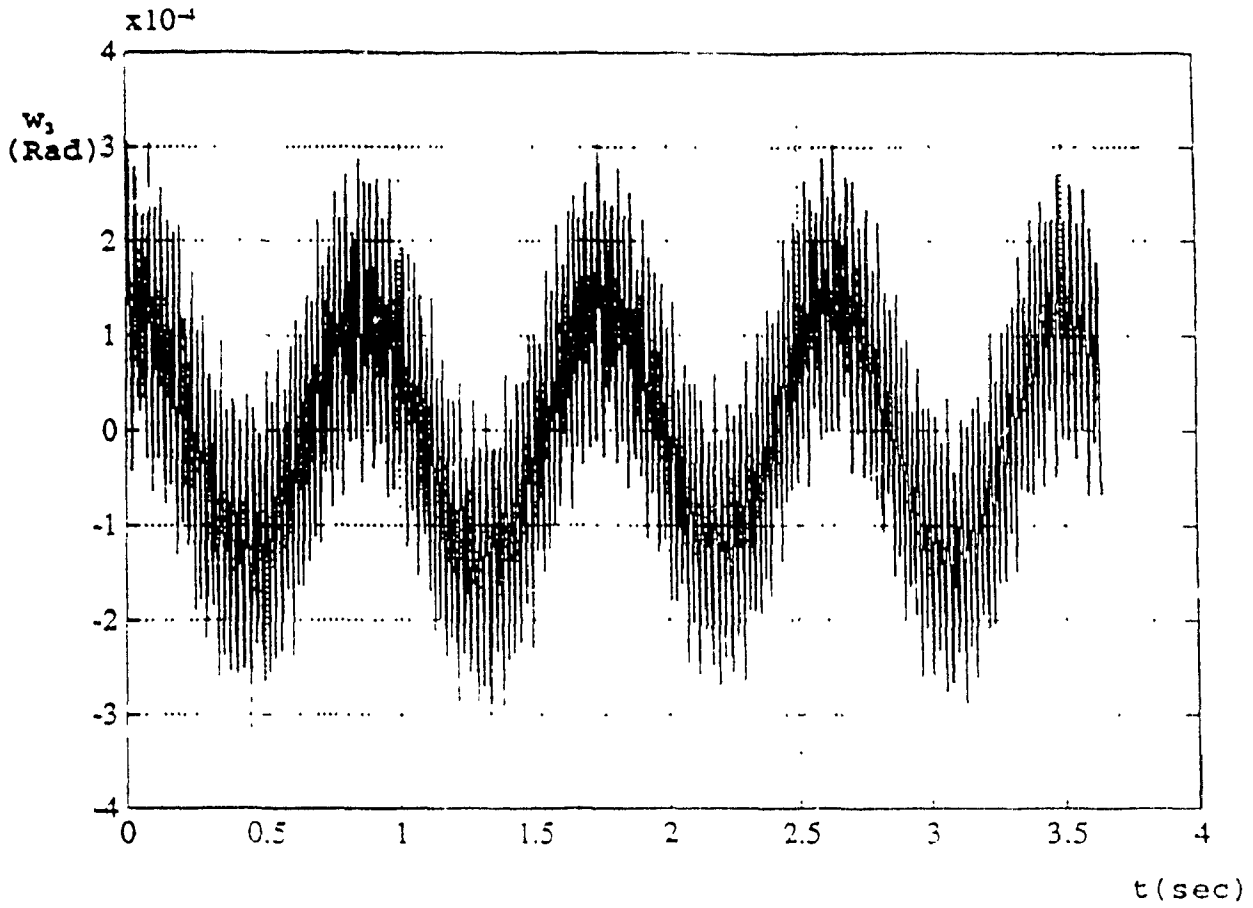


Fig. 4.10 Flexural Slope at Link 2 Midpoint in the First Simulation

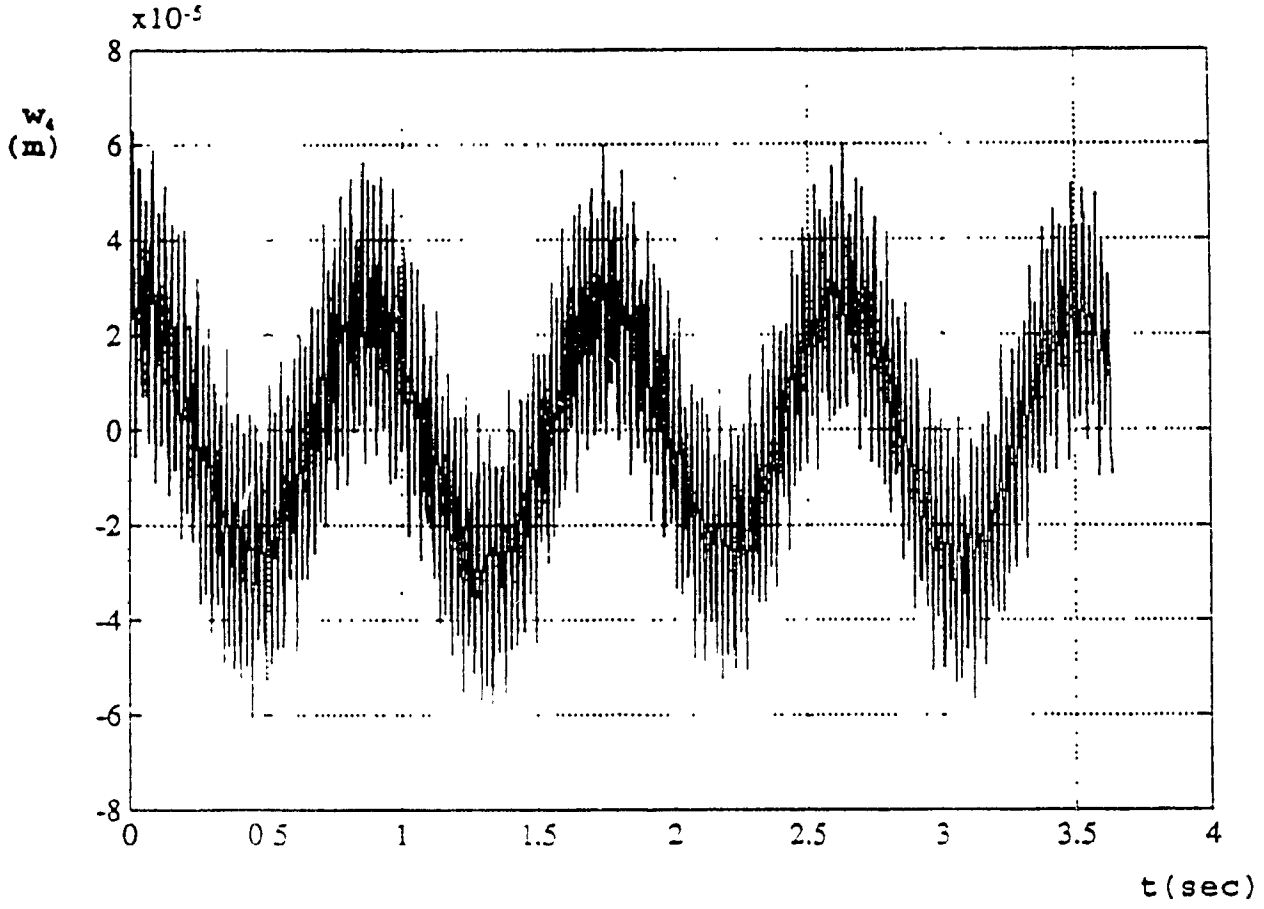


Fig. 4.11 Flexural Displacement at Link 2 Midpoint in the First Simulation

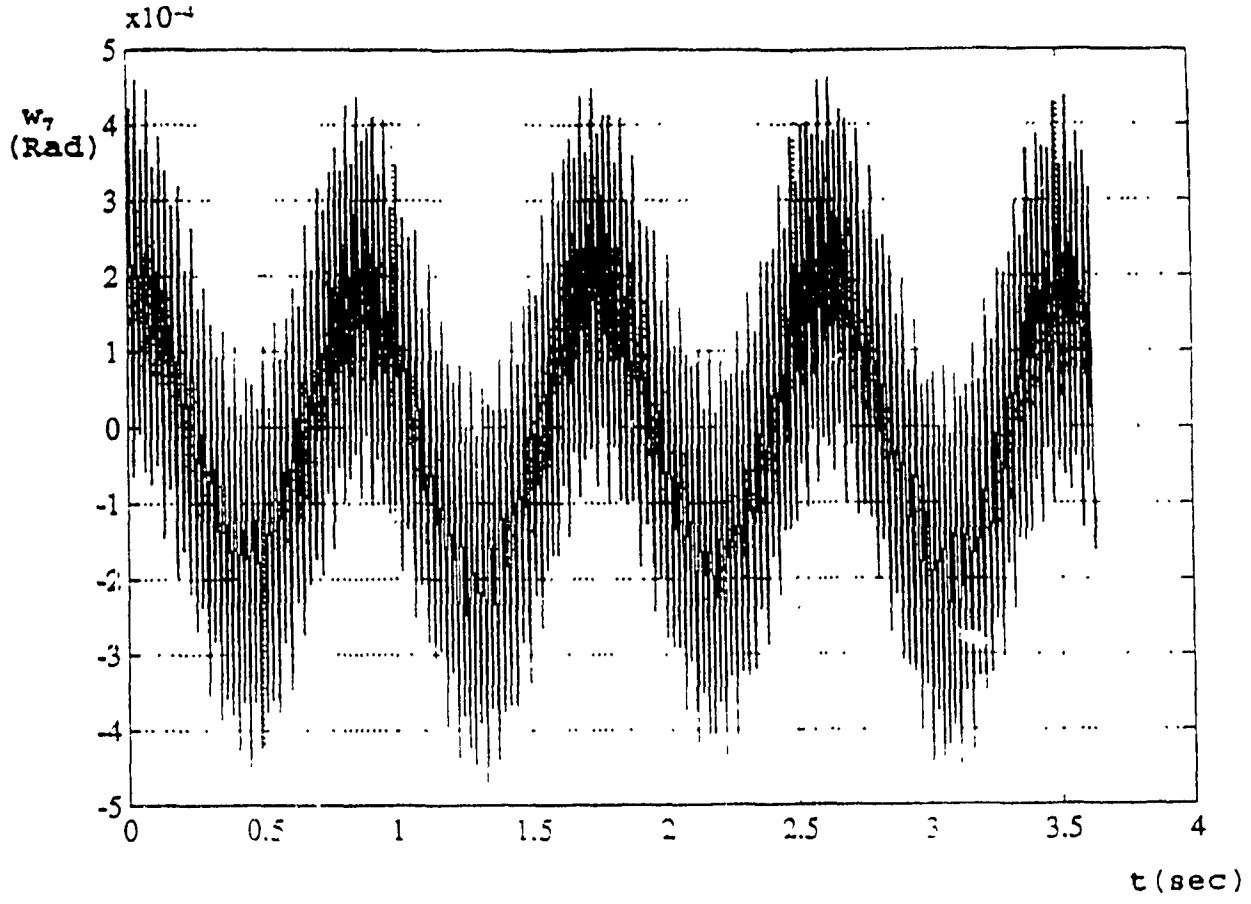


Fig. 4.12 Flexural Slope at Link 2 Tip in the First Simulation

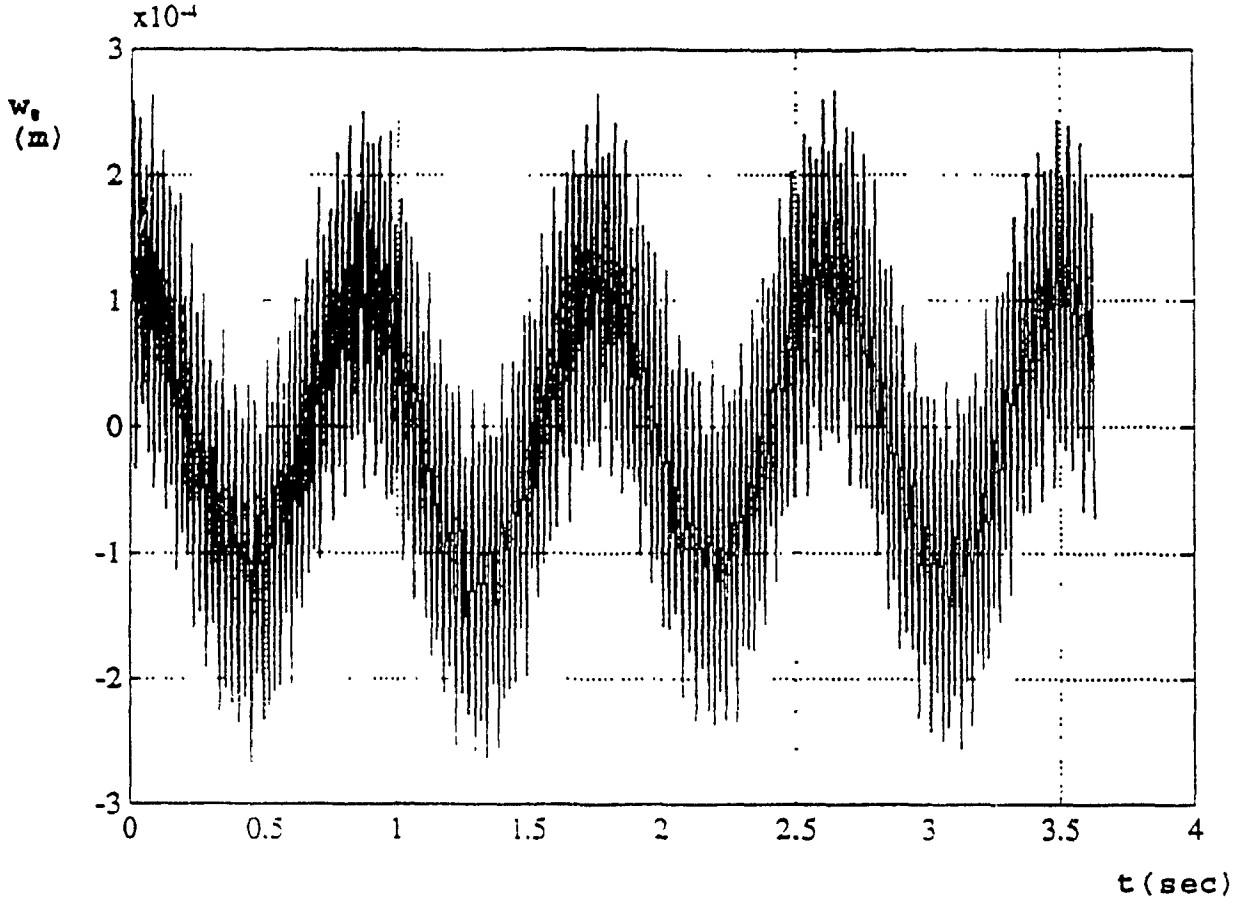


Fig. 4.13 Flexural Displacement of Link 2 Tip in the First Simulation

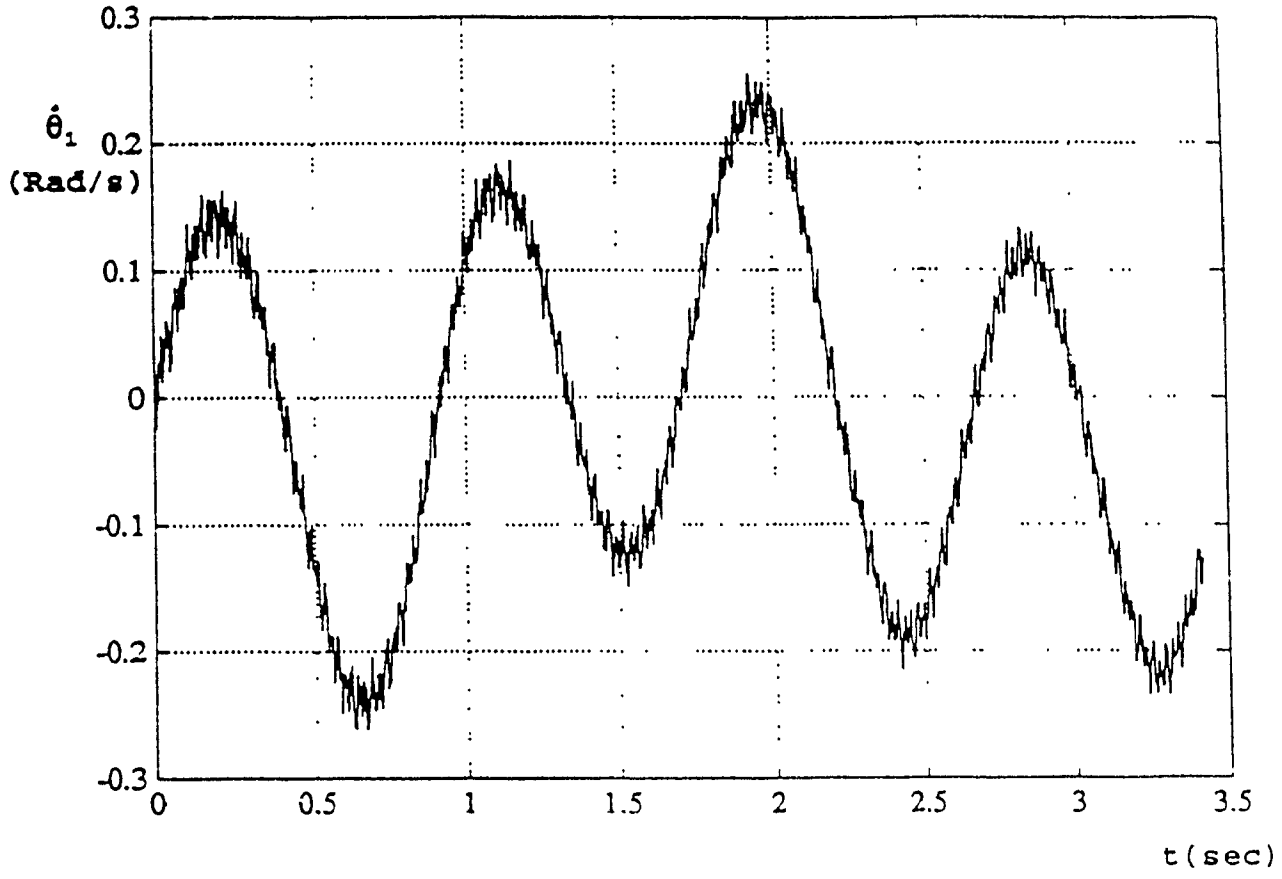


Fig. 4.14 The Angular Velocity of Joint 1
In The First Simulation

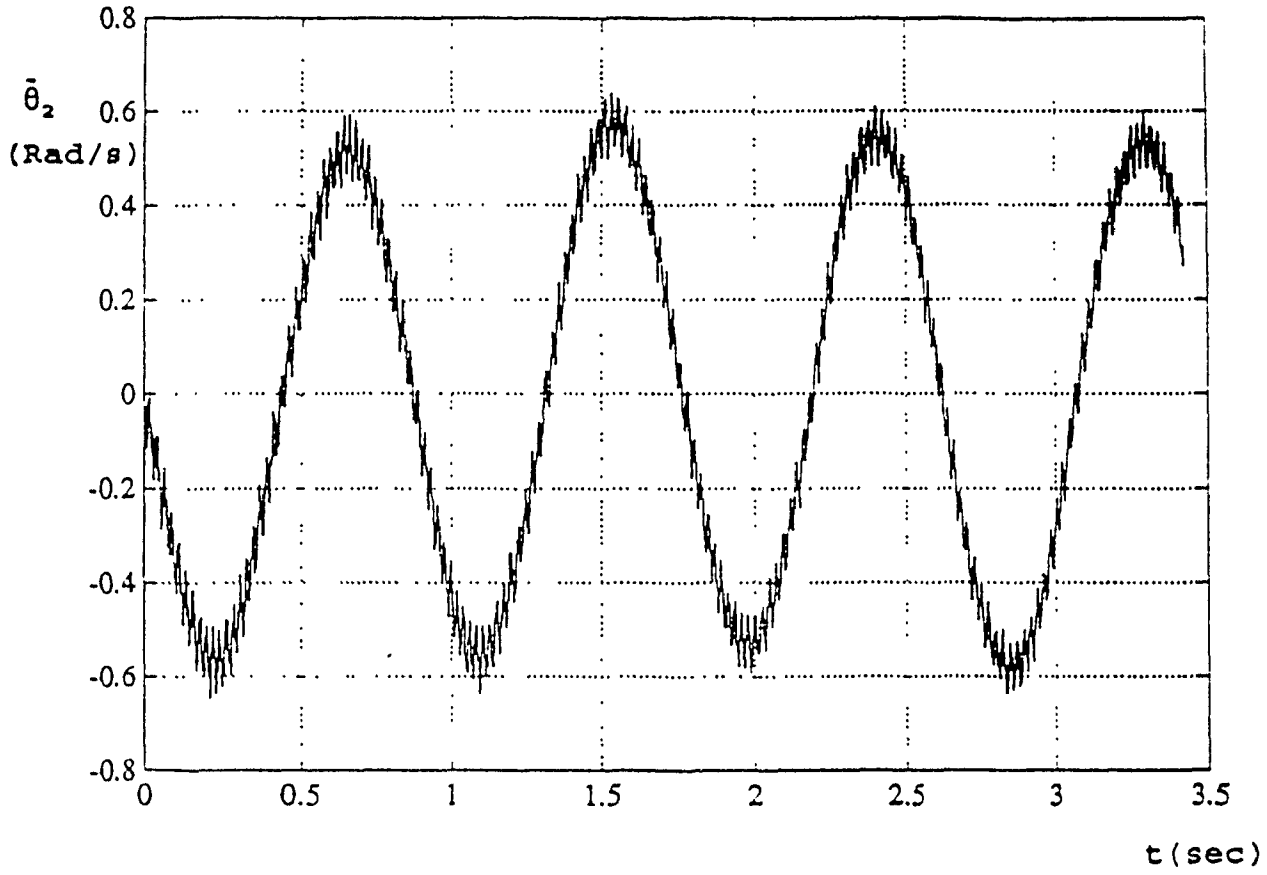


Fig. 4.15 The Angular Velocity of Joint 2 In The First Simulation

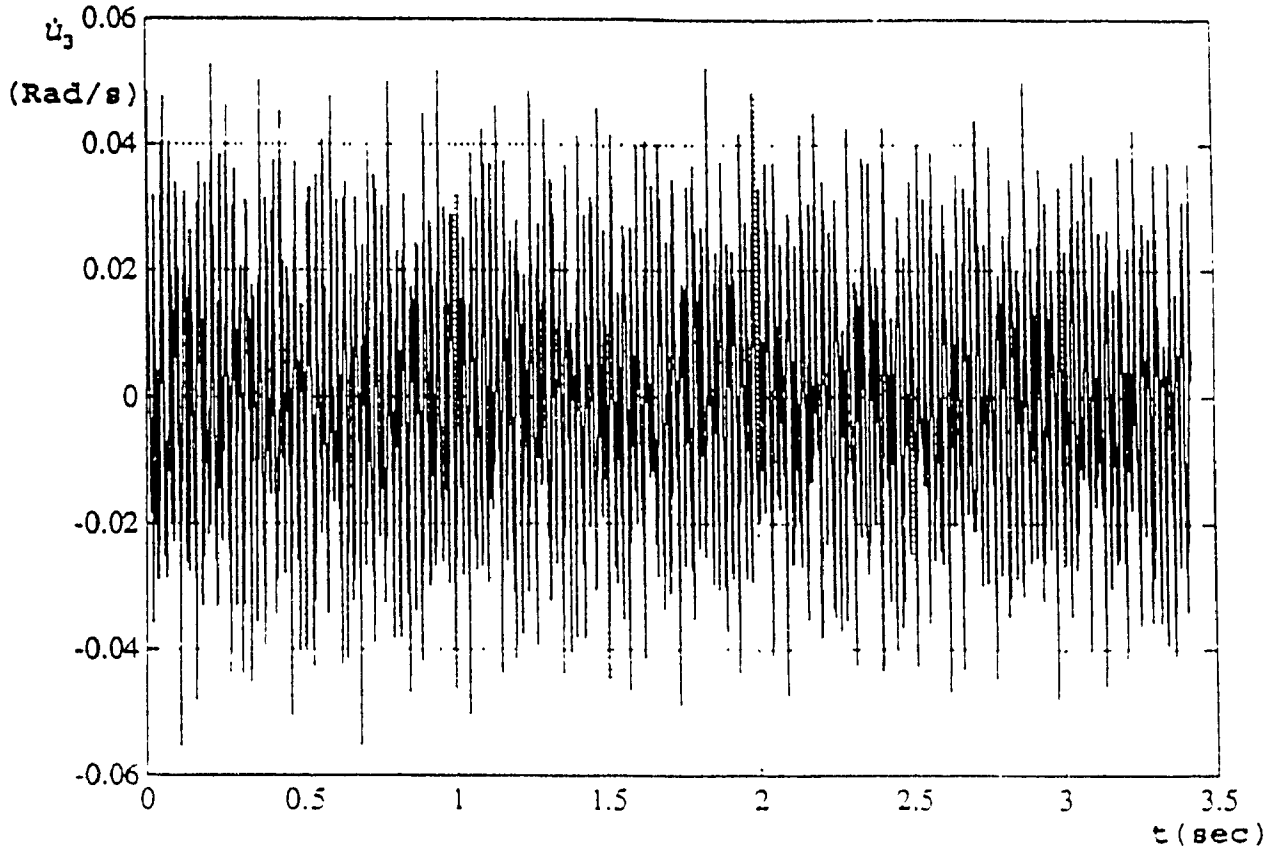


Fig. 4.16 Flexural Angular Velocity at Link 1 Midpoint in the First Simulation

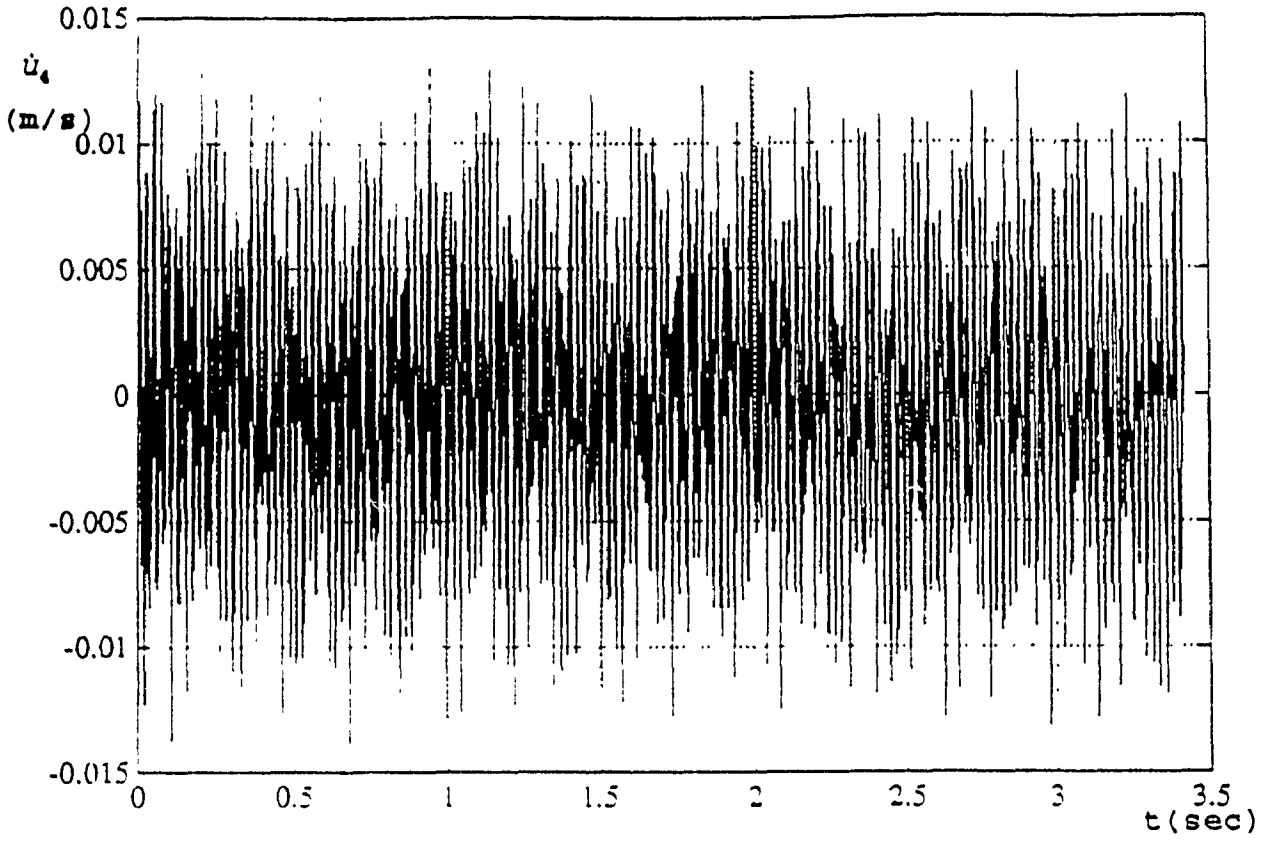


Fig. 4.17 Flexural Velocity at Link 1 Midpoint in the First Simulation

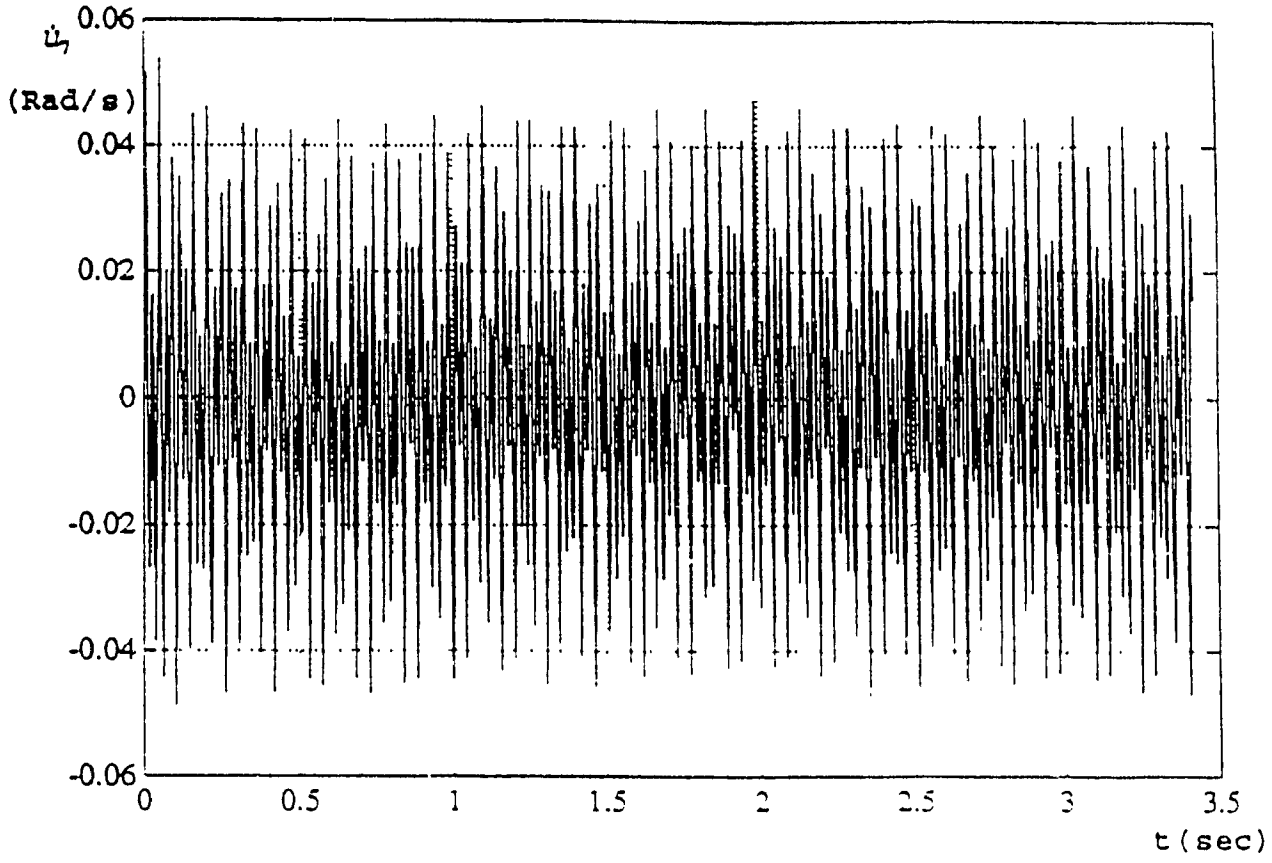


Fig. 4.18 Flexural Angular Velocity at Link 1 Tip in the First Simulation

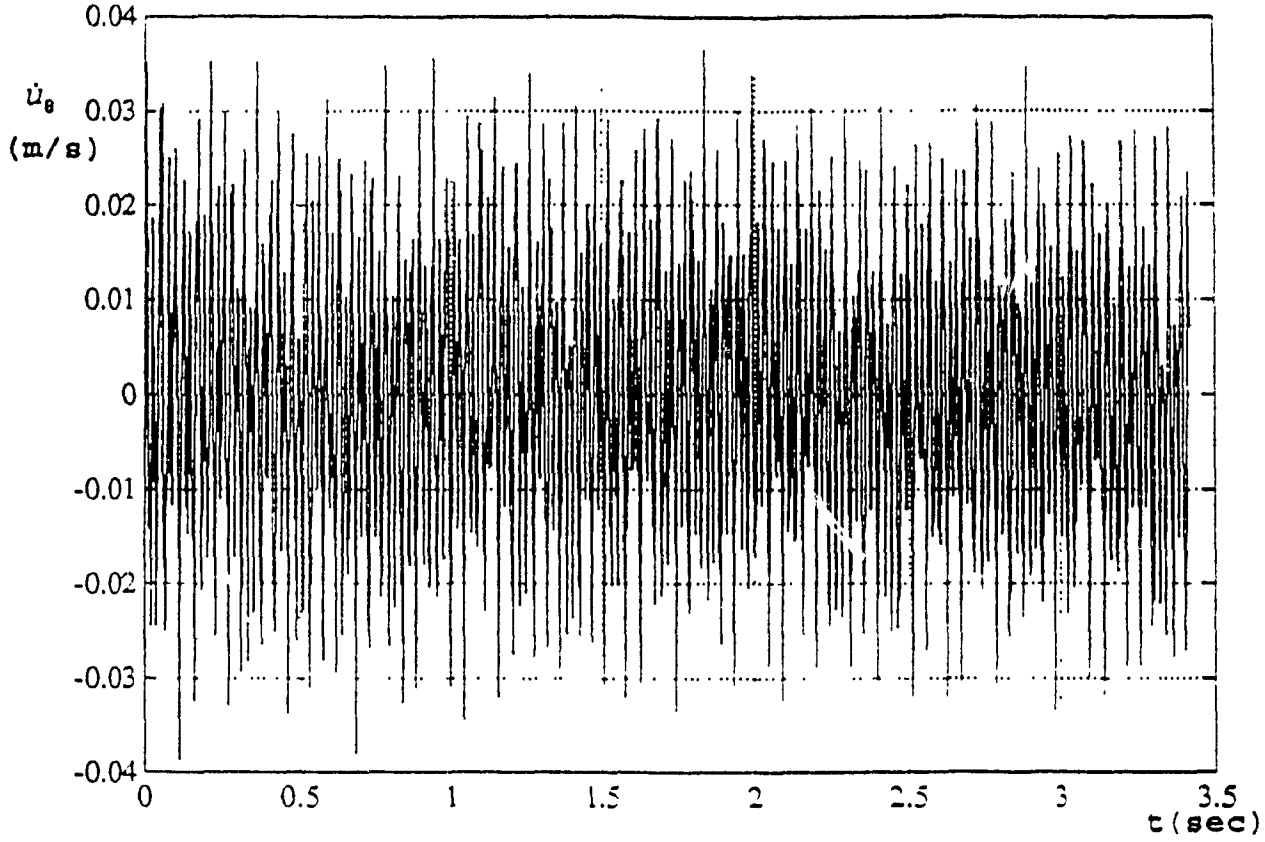


Fig. 4.19 Flexural Velocity of Link 1 Tip in the First Simulation

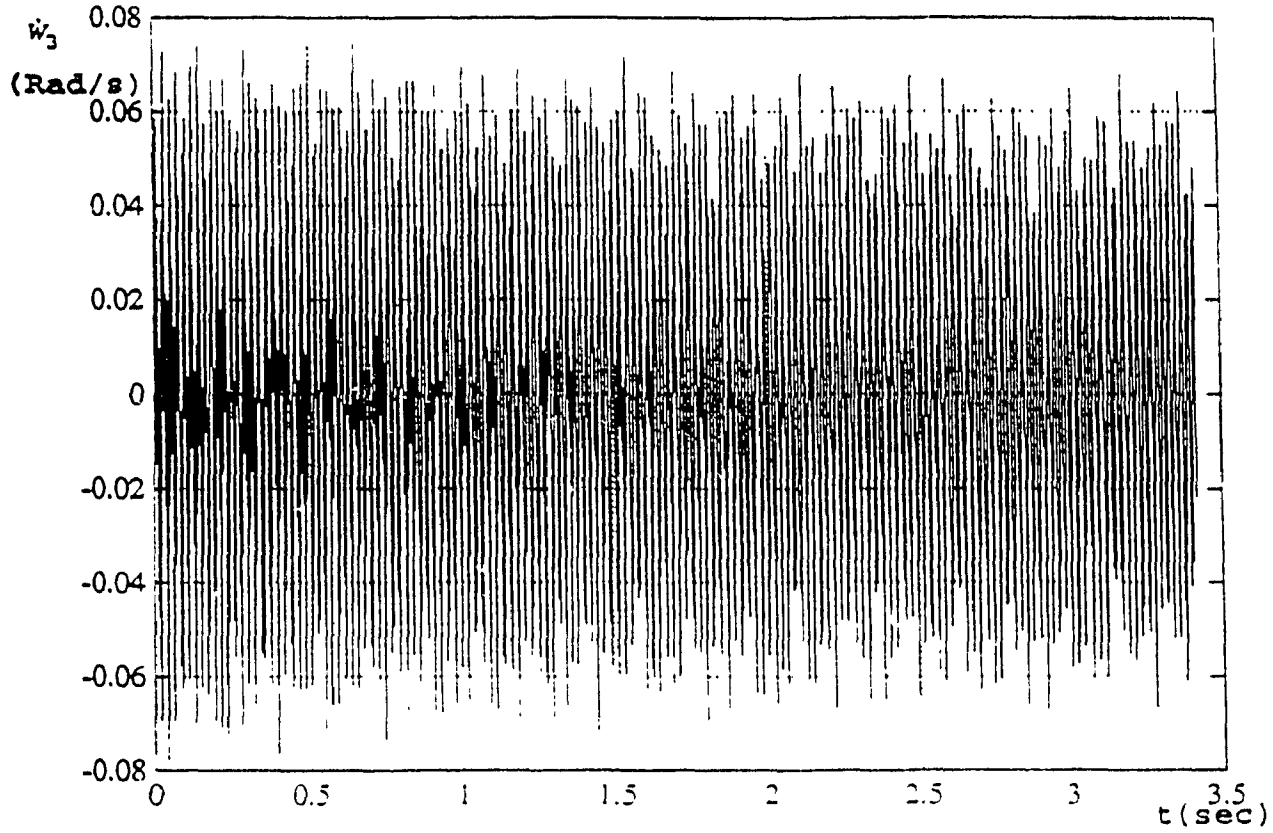


Fig. 4.20 Flexural Angular Velocity at Link 2 Midpoint in the First Simulation

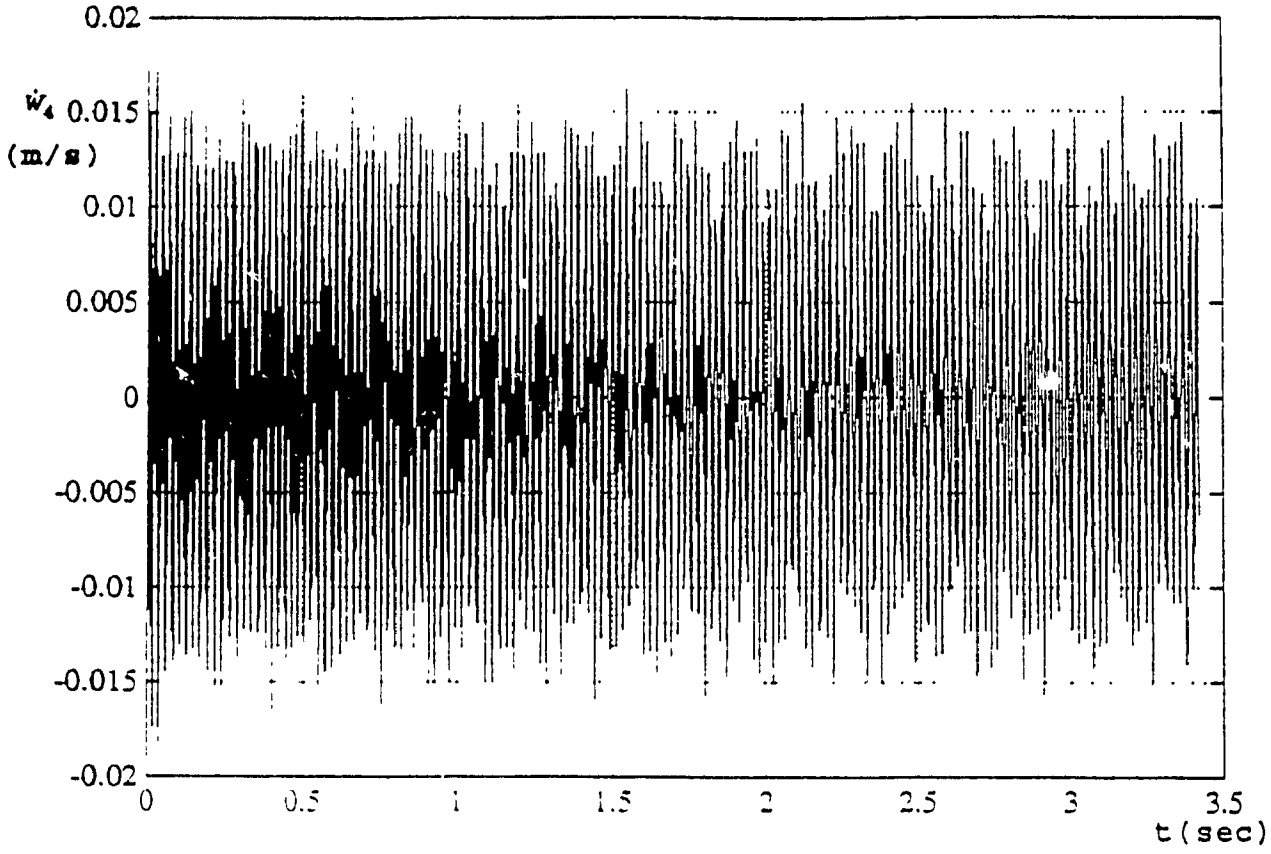


Fig. 4.21 Flexural Velocity at Link 2 Midpoint in the First Simulation

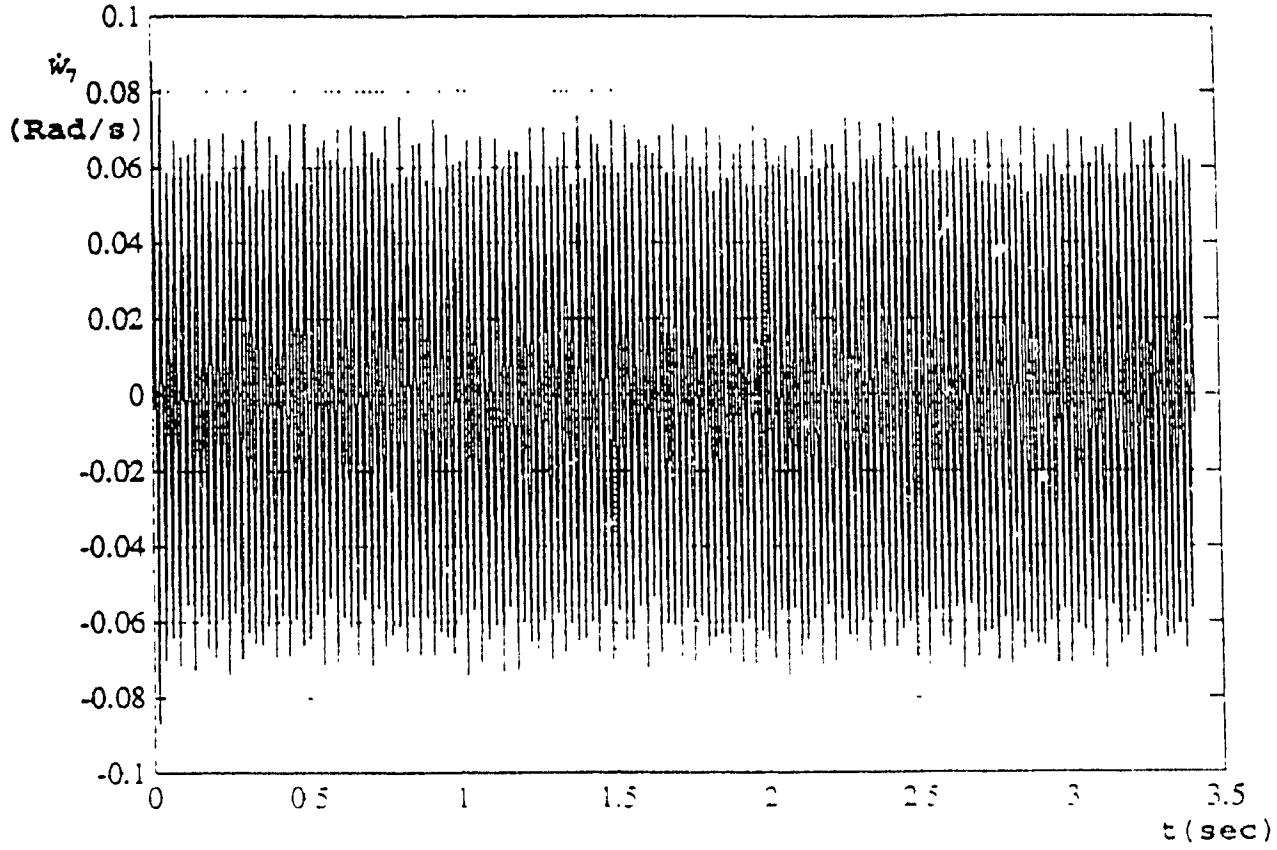


Fig. 4.22 Flexural Angular Velocity at Link 2 Tip in the First Simulation

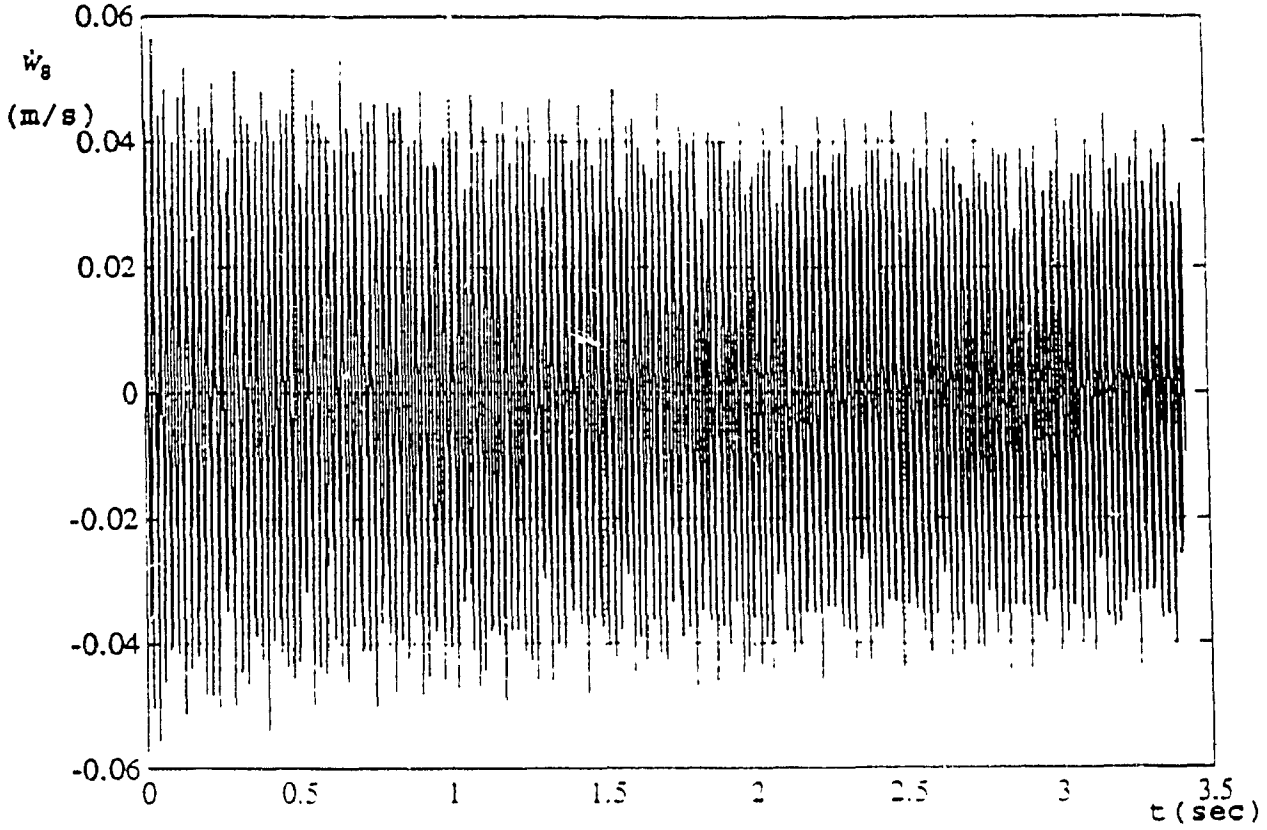


Fig. 4.23 Flexural Velocity of Link 2 Tip in the First Simulation

of Usoro's work [10], all the responses in this simulation coincide with those of Usoro's except for four of them, namely, u_1, u_4, u_7, u_8. The differences can be seen from Fig. 4.6 to Fig. 4.9 and in Fig. 4.24, Fig 4.25. Fig. 4.6 to Fig. 4.9 illustrate the responses obtained from the procedures developed in this thesis while Fig. 4.24 and 4.25 shows the corresponding responses given in Usoro's work. Although they are quite different, there is no adequate evidence to judge which is right. However, it may be observed that the shape of slow vibration mode of flexural deflection follows the shape of its corresponding joint deflection (Fig. 4.4, Fig. 4.6 to 4.9). These are already shown in the responses of the second link obtained from this investigation as well as from Usoro's work (Fig. 4.5, Fig. 4.10 to 4.13; Fig. 4.24, Fig. 4.25). Further, Fig. 4.6 to Fig. 4.9 have been obtained for a more generalized unrestricted dynamic model and hence the results presented in the thesis are believed to be valid.

4.3 THE SIMULATION OF ONE-LINK MANIPULATOR WITH TORQUE

In order to check validity of the model developed for the more common case when external torques have to be taken into account and also as well to reassure the applicability and validility of the model, a simple one-link system with revolute base joint is introduced. In this one-link manipulator simulation, the torque is exerted on the first

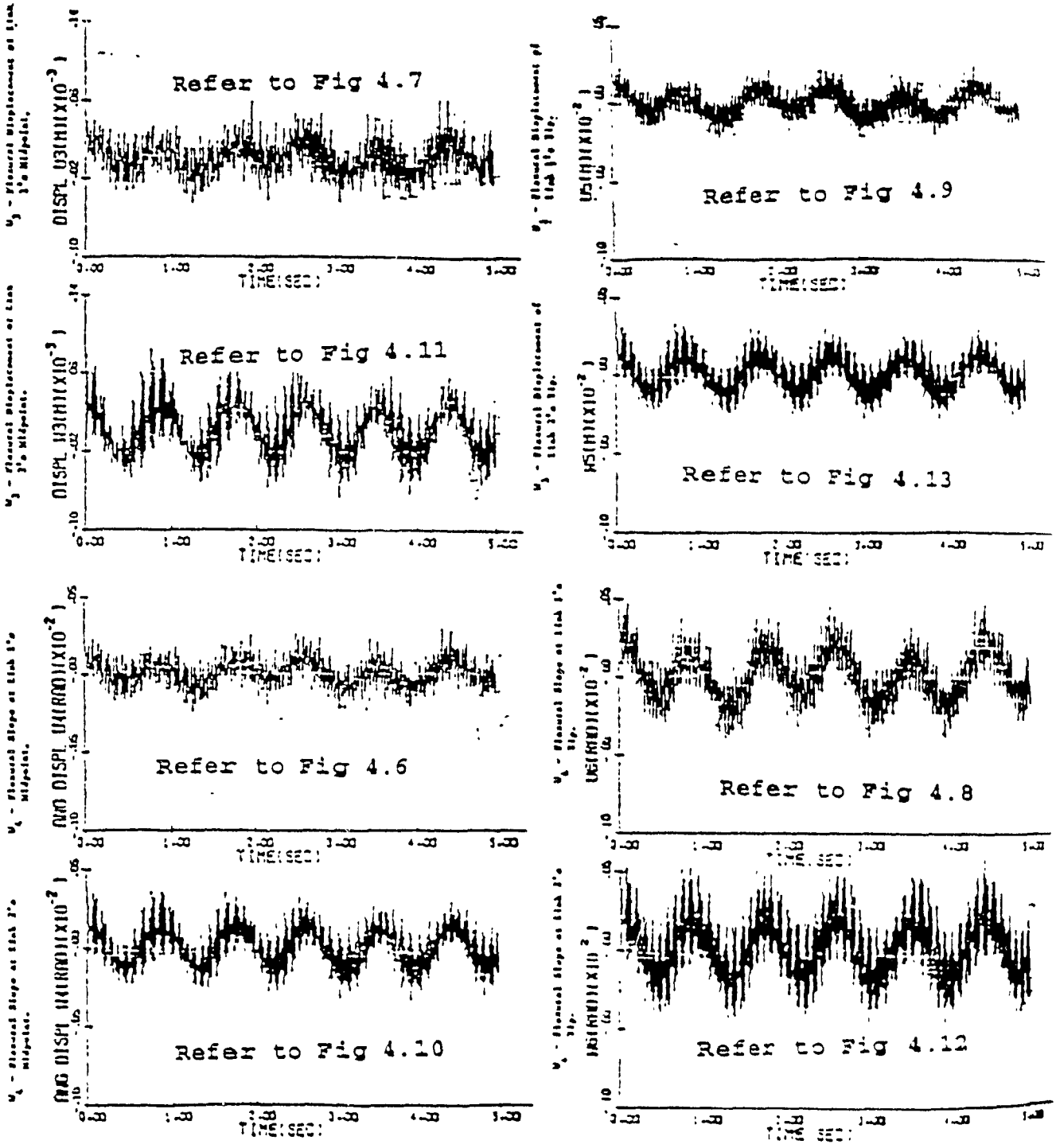
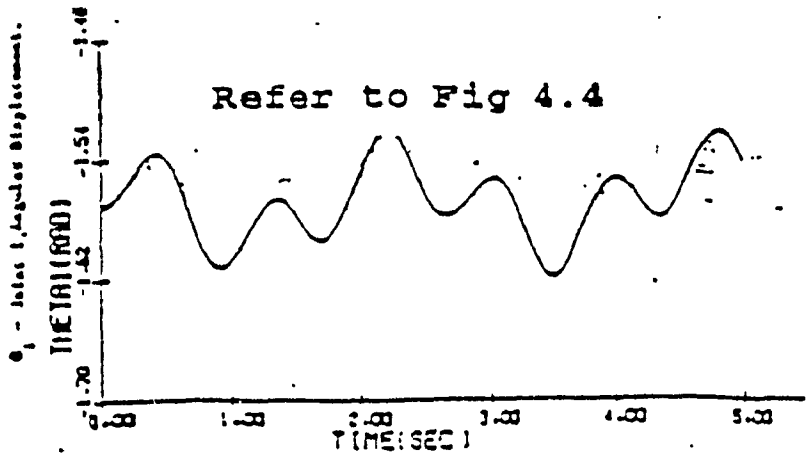


Fig. 4.24 The Elastic Deflection from Usoro's Work



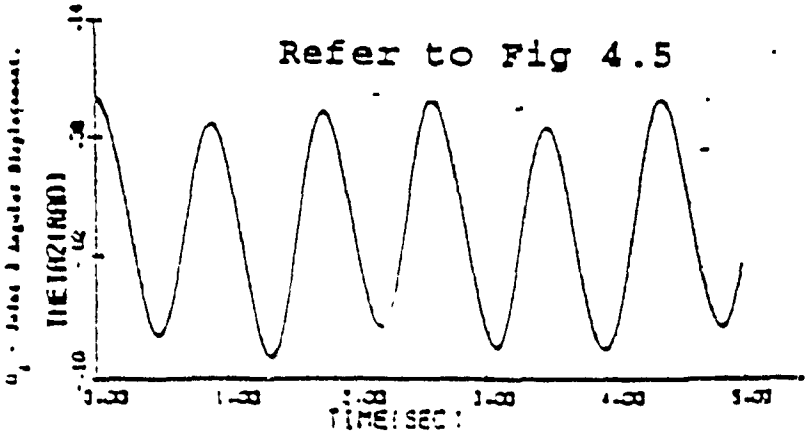


Fig. 4.25 The Joint Angular Deflection from Usoro's Work

joint. The joint input torques are show in Fig 4.26 and are specified as follows:

$$\tau = 35\ N{\cdot}M, \quad 0 \le t < 1s \tag{4.38a}$$

$$\tau = -35\ N{\cdot}M, \quad 1 \le t < 2s \tag{4.38b}$$

$$\tau = 0\ N{\cdot}M, \quad t \ge 2s \tag{4.38c}$$

Other properties for the one-link manipulator are listed in Table 4.3 and are taken from [10]. The initial configuration of the one-link manipulator is also shown in Fig. 4.27.

By using the same shape functions as employed in the two link manipulator simulation and following the same procedure as mentioned in the first simulation, the system's energy (potential energy and kinetic energy) as well as the dynamic equations for this system can be obtained.

By arranging the dynamic equations into the form of equation (1.1), the following matrices corresponding to those in equation (1.1) are obtained:

$$A = \begin{bmatrix} a_{11} & 0.0417 & -0.0833 & 1.25 & 1.0625 \\ 0.0417 & 0.0119 & -0.0045 & 0 & 0.0387 \\ -0.0833 & -0.0045 & 0.006 & -0.0387 & -0.0655 \\ 1.25 & 0 & -0.0387 & 1.8571 & 0.3214 \\ 1.0625 & 0.0387 & -0.0655 & 0.3214 & 0.9286 \end{bmatrix} \tag{4.39}$$

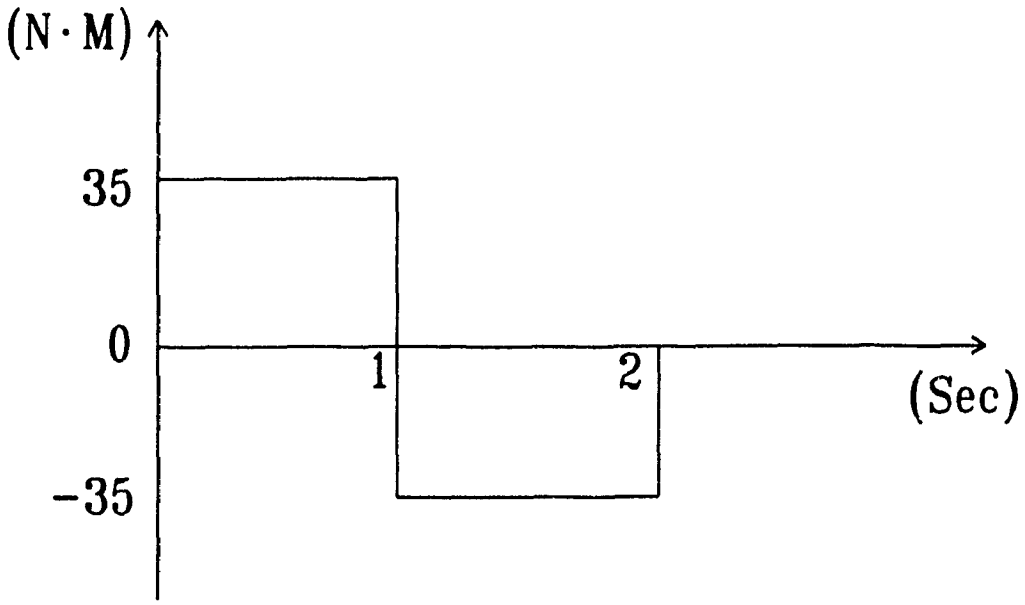


Fig. 4.26 Joint Actuator Torque for the Simulation

Table 4.3 One-link manipulator properties

l_1	1	m
I_1	5×10^{-9}	m^4
μ_1	5	kg/m
E_1	2×10^{11}	N/m^2
$\theta_1(0)$	0	deg

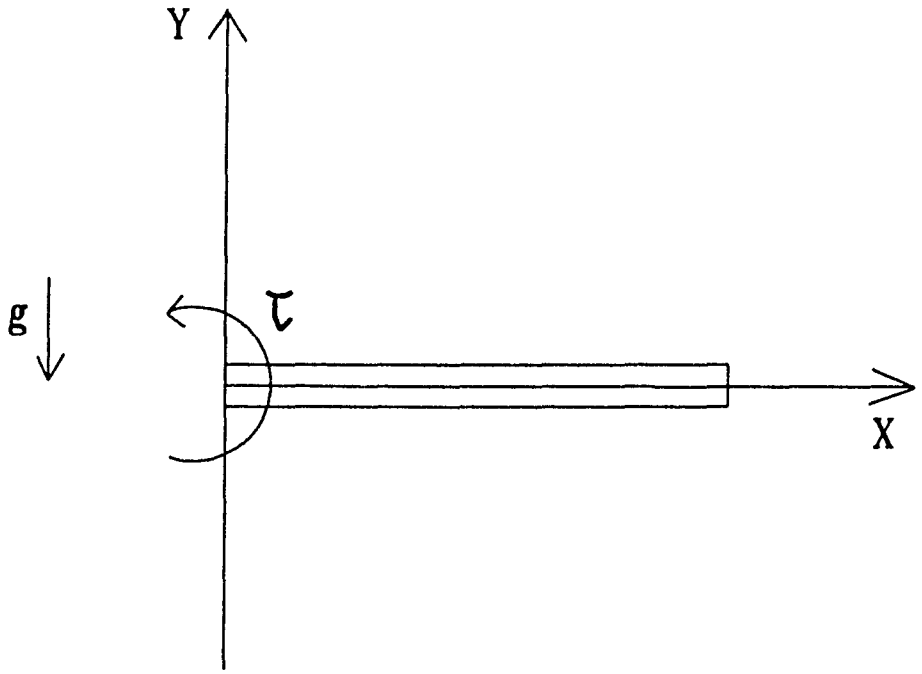


Fig. 4.27 Initial Condition for One-link Manipulator Simulation

$$H=\begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{bmatrix} \tag{4.40}$$

and

$$\tau=\begin{bmatrix} \tau_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.41}$$

where

$$\begin{aligned} a_{11}= 0.5\mu(&5.33l^3+0.038l^3u_3^2+0.019l^3u_7^2+1.48lu_4^2 \\ &+0.74lu_8^2-0.0286l^3u_3u_4+0.124l^2u_3u_8 \\ &-0.21l^2u_7u_8-0.124l^2u_4u_7+0.514u_4u_8 \end{aligned} \tag{4.42a}$$

$$\begin{aligned} h_1=&\mu g(2l^2\cos\theta_1+0.083l^2u_7\sin\theta_1-lu_4\sin\theta_1-\frac{l}{2}u_8\sin\theta_1) \\ &+\mu\dot{\theta}_1(0.038l^3\dot{u}_3u_3+0.019l^3\dot{u}_7u_7+1.486l\dot{u}_4u_4+0.74l\dot{u}_8u_8 \\ &-0.014l^3(\dot{u}_3u_7+\dot{u}_7u_3)+0.062l^2(\dot{u}_3u_8+u_3\dot{u}_8)-0.10476l^2 \\ &(\dot{u}_7u_8+u_7\dot{u}_8)-0.062l^2(\dot{u}_7u_4+u_7\dot{u}_4)+0.257l(\dot{u}_4u_8+u_4\dot{u}_8) \end{aligned} \tag{4.42b}$$

$$h_2=-\mu\dot{\theta}_1^2(0.019l^3u_3-0.00714l^3u_7 +0.031l^2u_8)+EI(\frac{8}{l}u_3+\frac{2}{l}u_7-\frac{6}{l^2}u_8) \quad (4.42c)$$

$$h_3=0.5\mu\dot{\theta}_1^2(0.014l^3u_3+0.105l^2u_8+0.06l^2u_4-0.019l^3u_7) -0.083\mu gl^2\cos\theta_1+EI(\frac{4}{l}u_7+\frac{2}{l}u_3+\frac{6}{l^2}u_4-\frac{6}{l^2}u_8) \quad (4.42d)$$

$$h_4=-\mu\dot{\theta}_1(0.74lu_4-0.031l^2u_7+0.1285lu_8) +\mu gl\cos\theta_1+EI(\frac{24}{l^3}u_4+\frac{6}{l^2}u_7-\frac{12}{l^3}u_8) \quad (4.42e)$$

$$h_5=-\mu\dot{\theta}_1^2(0.37lu_8+0.031l^2u_3-0.0525l^2u_7+0.1285lu_4) +0.5\mu gl\cos\theta_1+EI(\frac{12}{l^3}u_8-\frac{6}{l^2}u_3-\frac{6}{l^2}u_7-\frac{12}{l^3}u_4) \quad (4.42f)$$

τ_1 in equation (4.41) is specified in equation set (4.38).

The variables vector $\boldsymbol{q}$ is $[\theta_1, u_3, u_4, u_7, u_8]^T$.

With the step of 0.0005 seconds, the simulation is implemented by the Runge-Kutta method computation procedure. The angular displacement of joint 1 is illustrated in Fig. 4.28. The joint angle increases till the time reaches certain point between 1 and 2 seconds. After this setting time, the

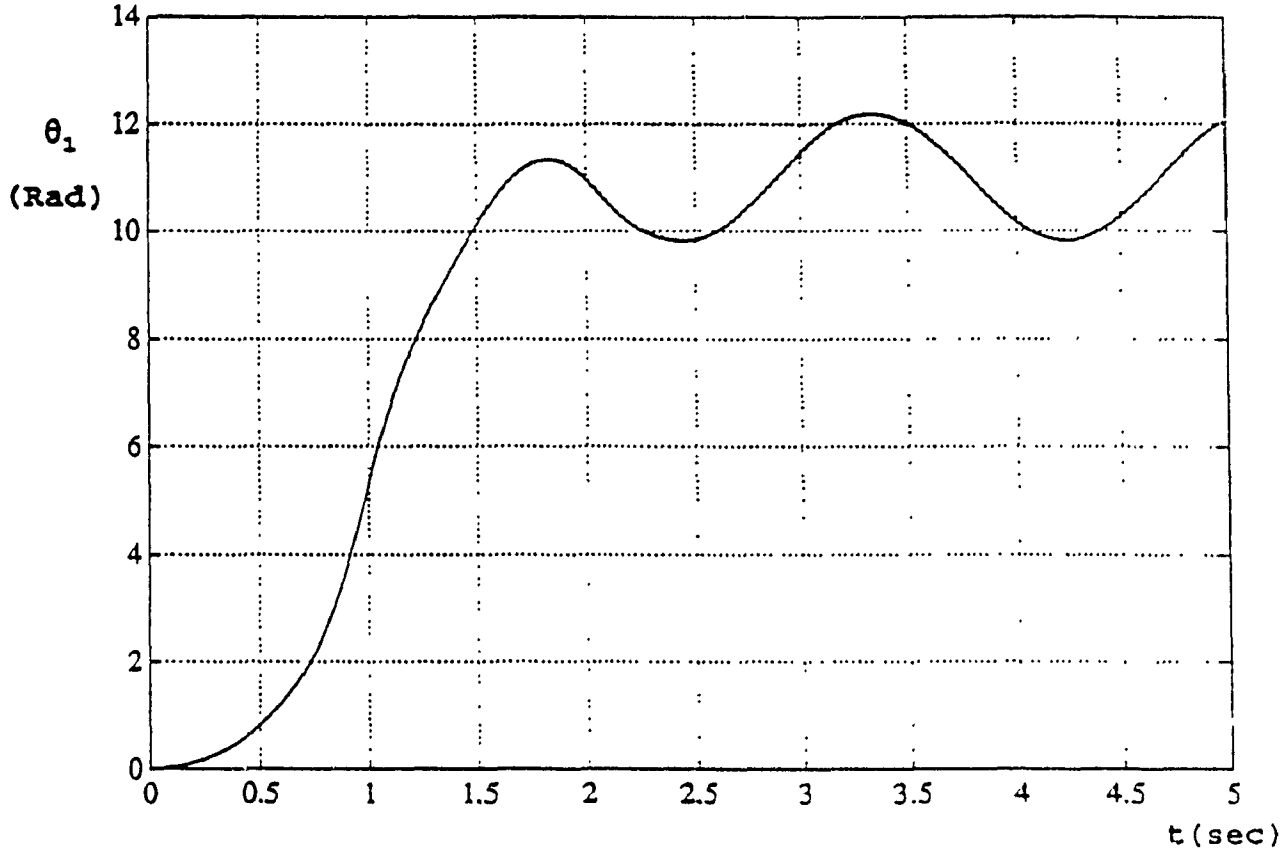


Fig. 4.28 The Angular deflection of Joint 1 in the Second Simulation

link performs with a free vibration response under the gravitational force.

Fig. 4.29 to Fig. 4.32 are responses for flexural deflection of the system. Each of the links undergoes 3 distinct periods of vibration response: from 0 to 1 second, the deflections are negative; from 1 to 2 seconds, they rise and most part of the deflection reaches positive values; after 2 seconds, they undergo free vibration. Fig. 4.33 is the angular velocity response of joint 1. This also can be identified in terms of 3 distinct periods: from 0 to 1 second, the velocity increases with the response oscillations; between 1 to 2 seconds, the velocity decreases with oscillations; after 2 seconds, they undergo free vibration with high frequency of oscillations. Fig. 4.34 to Fig. 4.37 are the flexural deflection velocities of link 1. The above mentioned 3 periods corresponding to each plot show a similar shape but with the different amplitude levels. These results are in conformance with the expected behaviour of the revolute jointed manipulator links with torque input , and hence the dynamic model developed can be said to have general application for design of flexible manipulators.

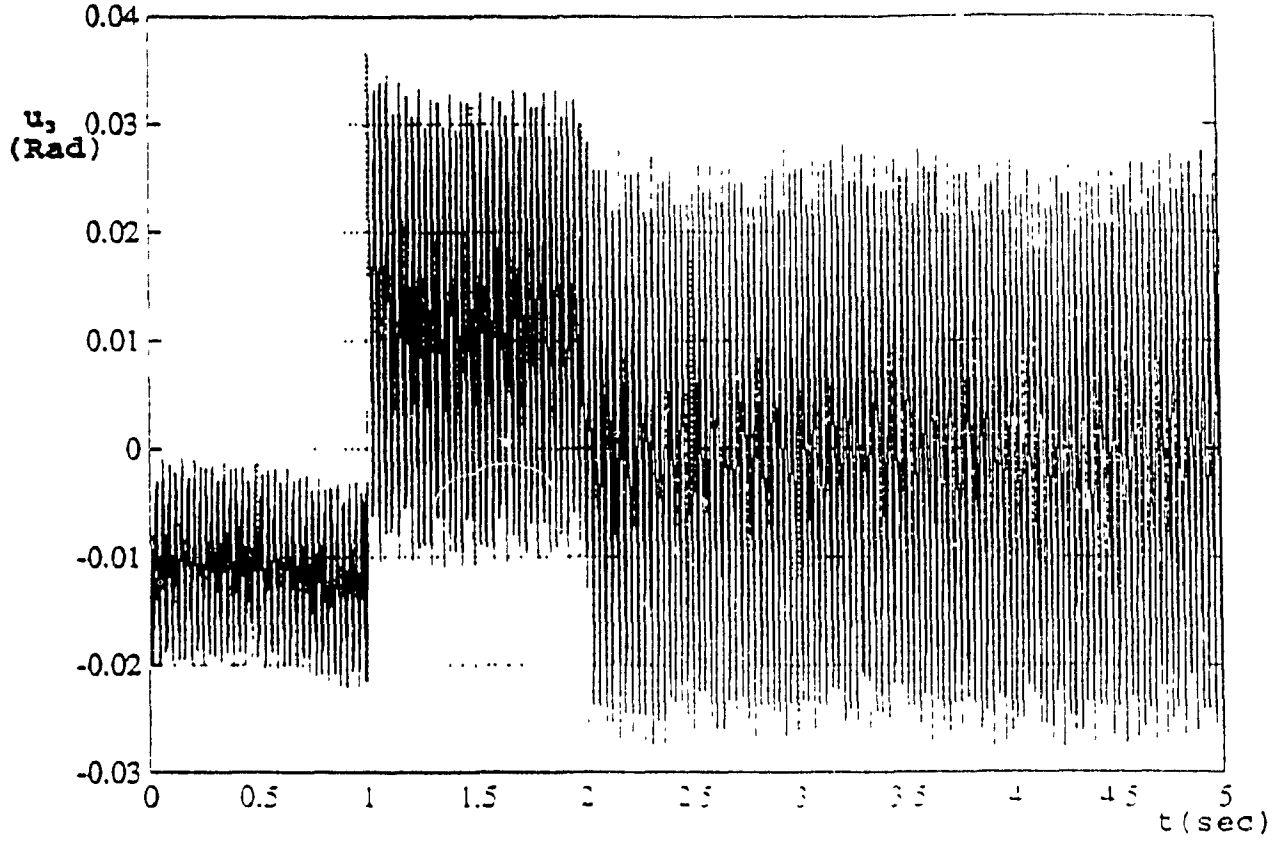


Fig. 4.29 Flexural Slope at Link 1 Midpoint in the Second Simulation

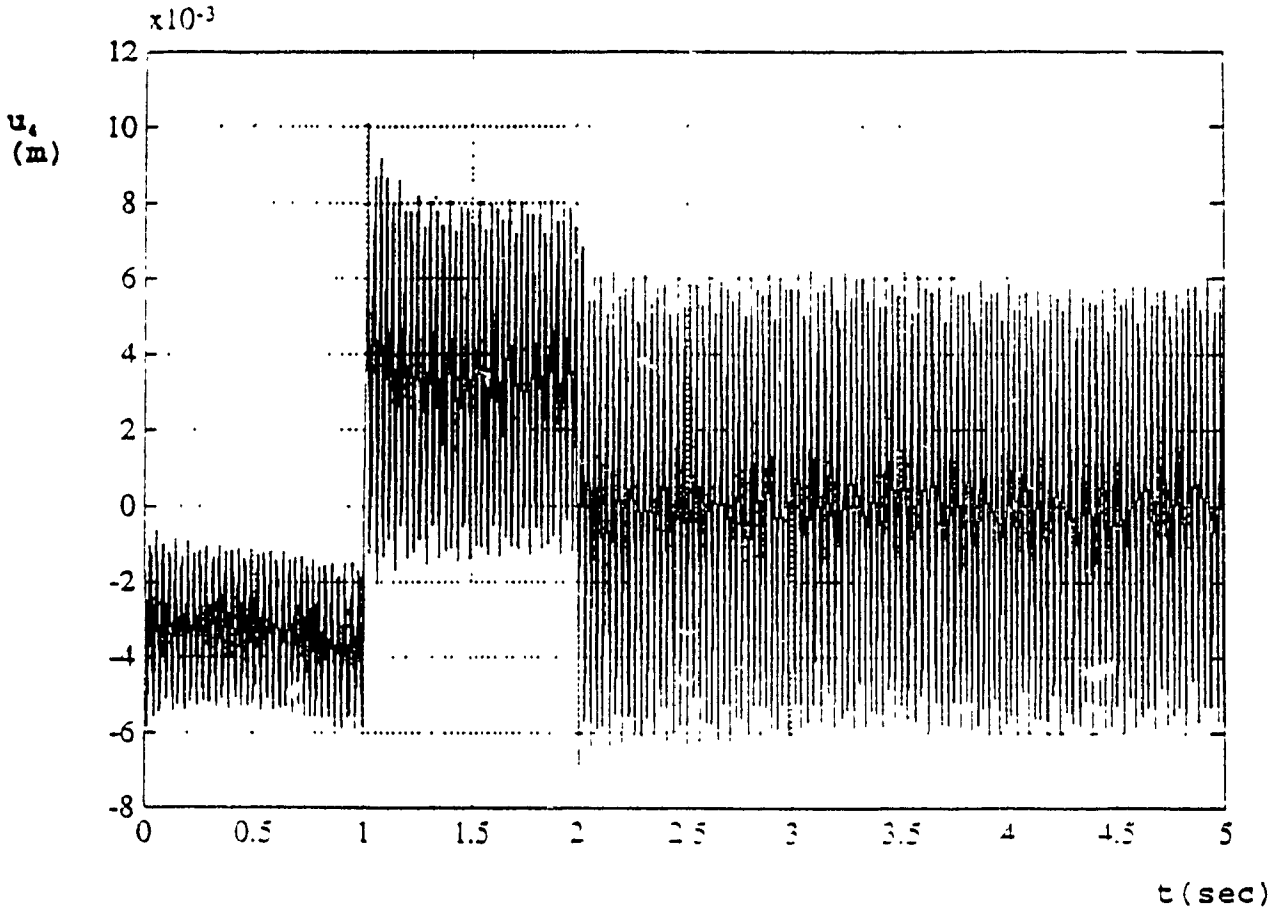


Fig. 4.30 Flexural Displacement at Link 1 Midpoint in the Second Simulation

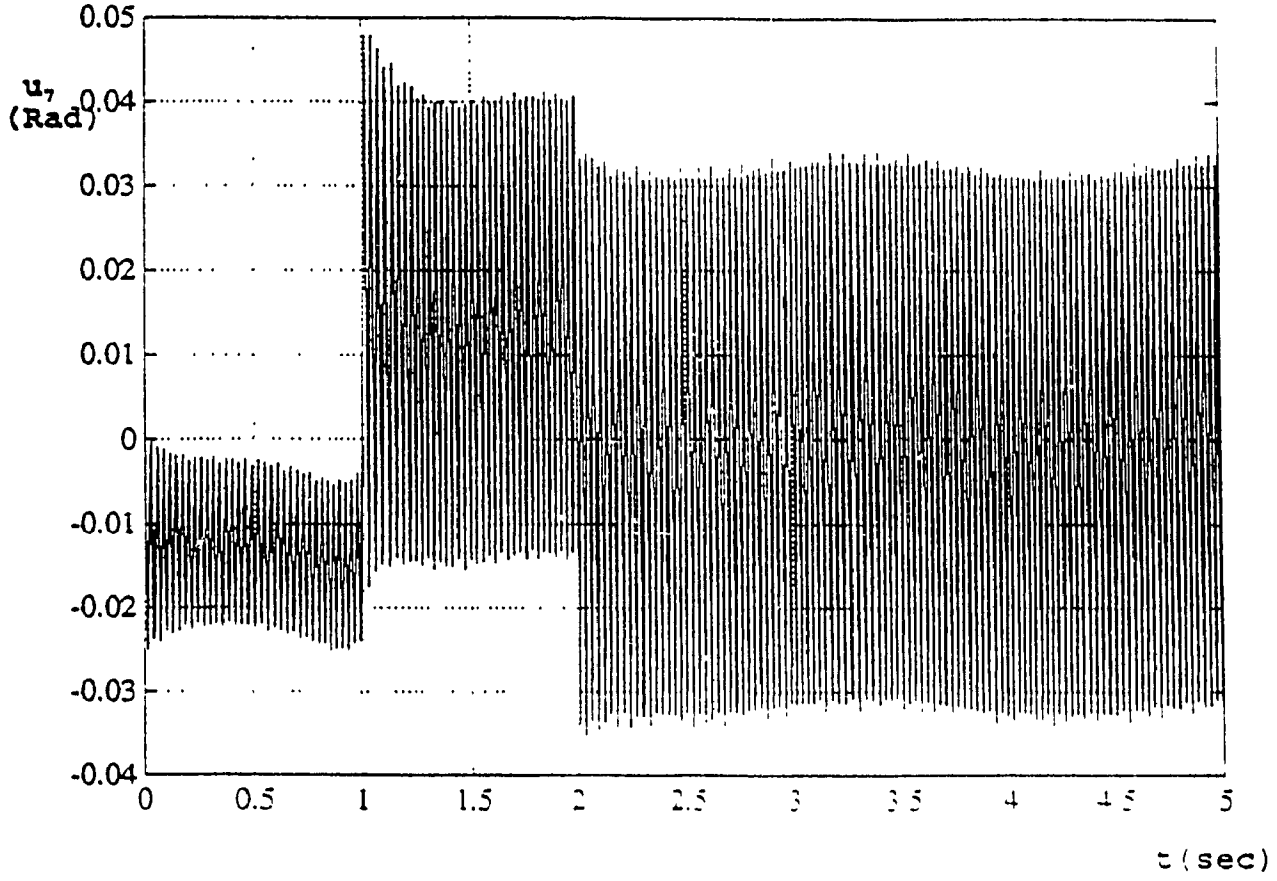


Fig. 4.31 Flexural Slope at Link 1 Tip in the Second Simulation

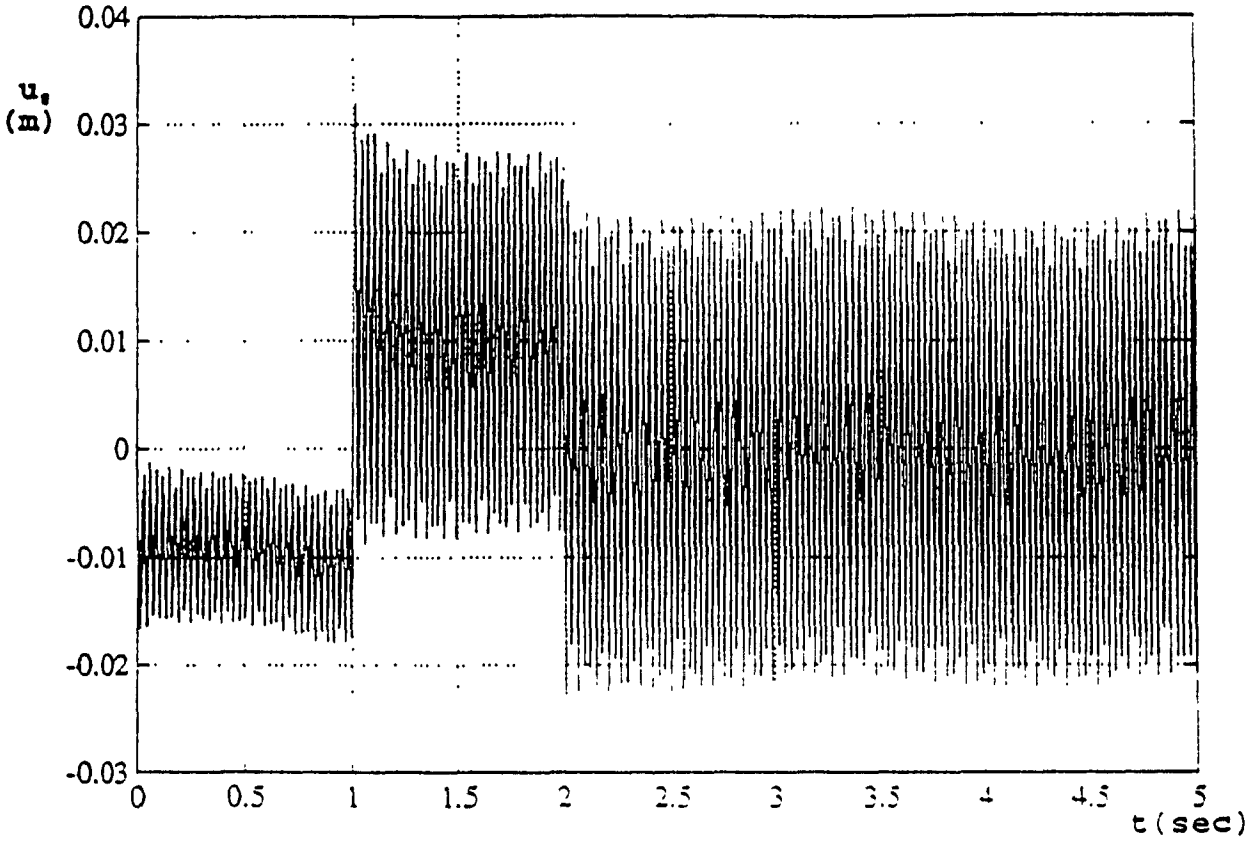


Fig. 4.32 Flexural Displacement of Link 1 Tip in the Second Simulation

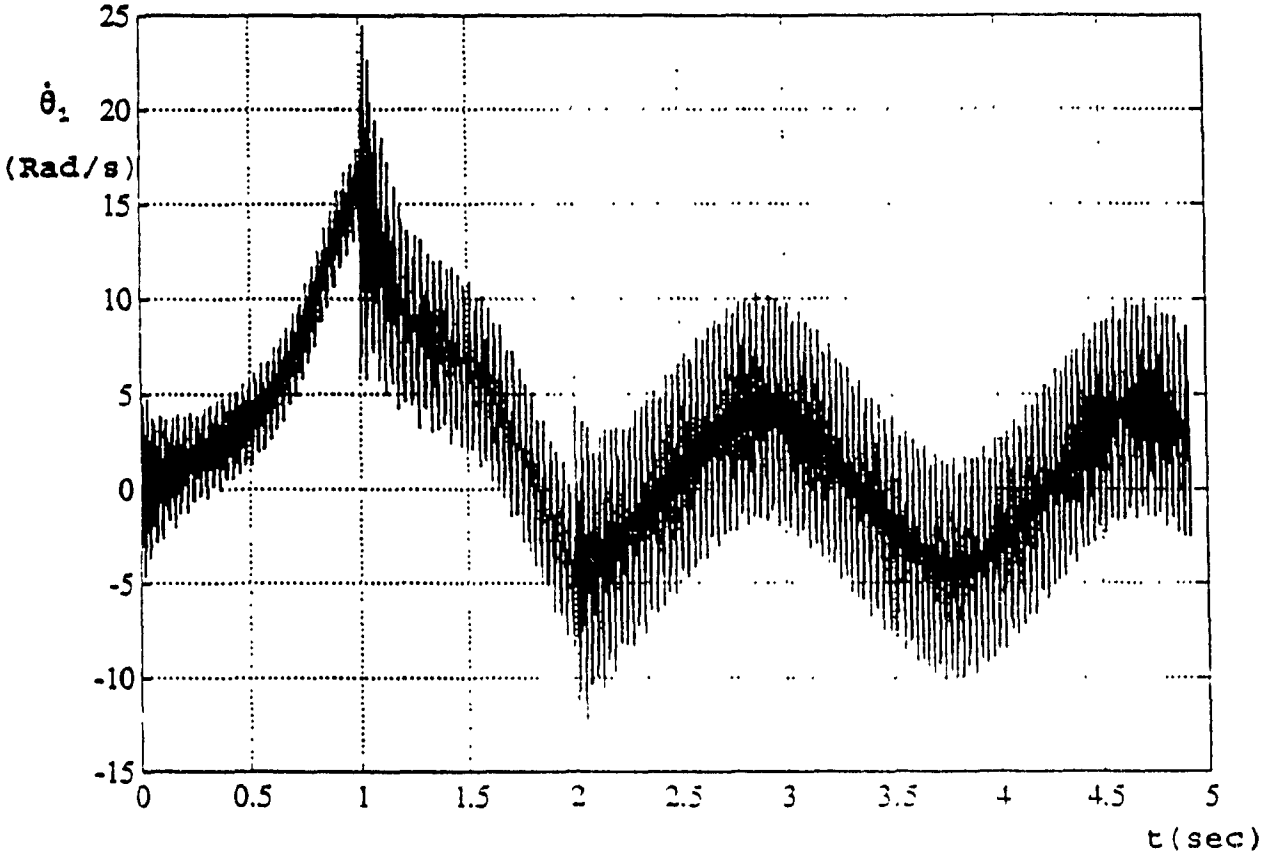


Fig. 4.33 The Angular Velocity of Joint 1 in the Second Simulation

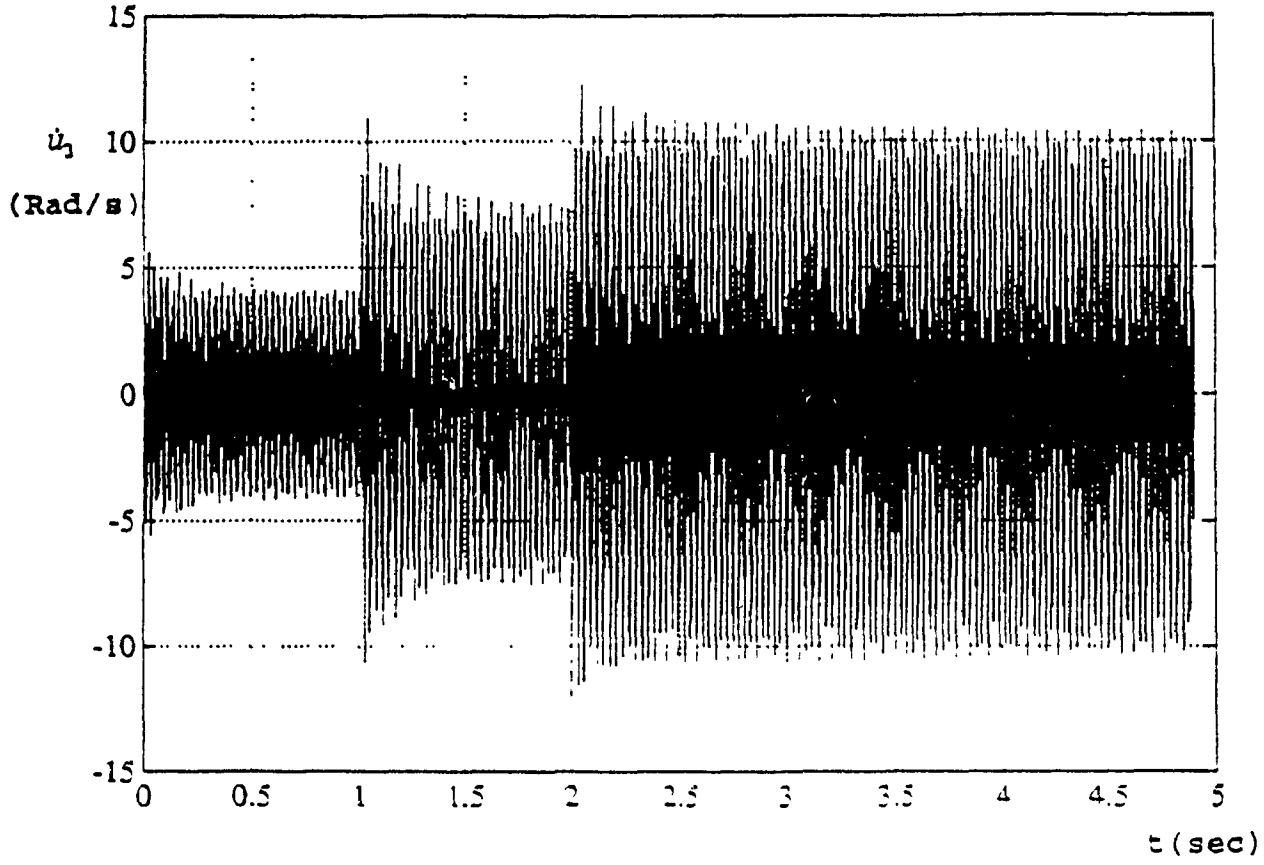


Fig. 4.34 Flexural Angular Velocity at Link 1 Midpoint in the Second Simulation

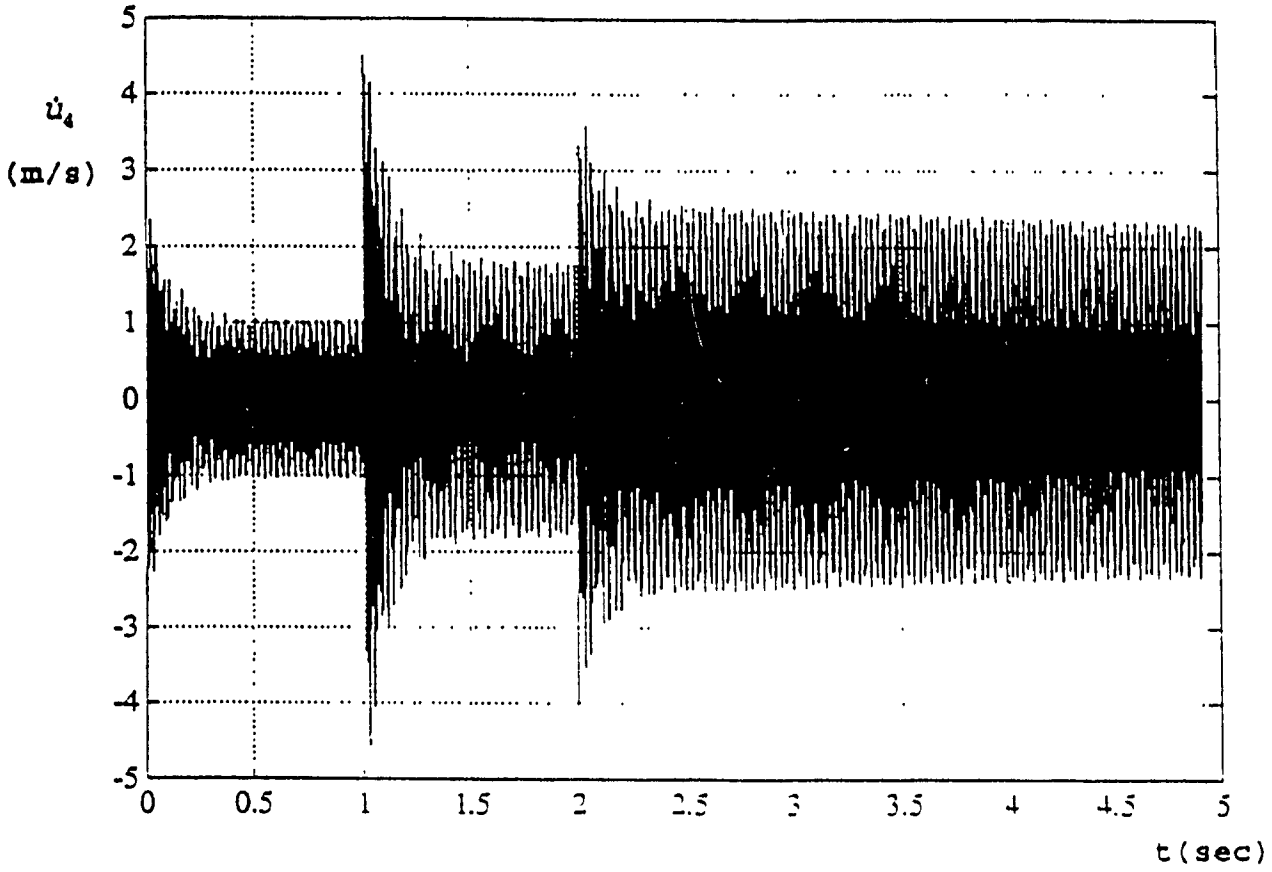


Fig. 4.35 Flexural Velocity at Link 1 Midpoint in the Second simulation

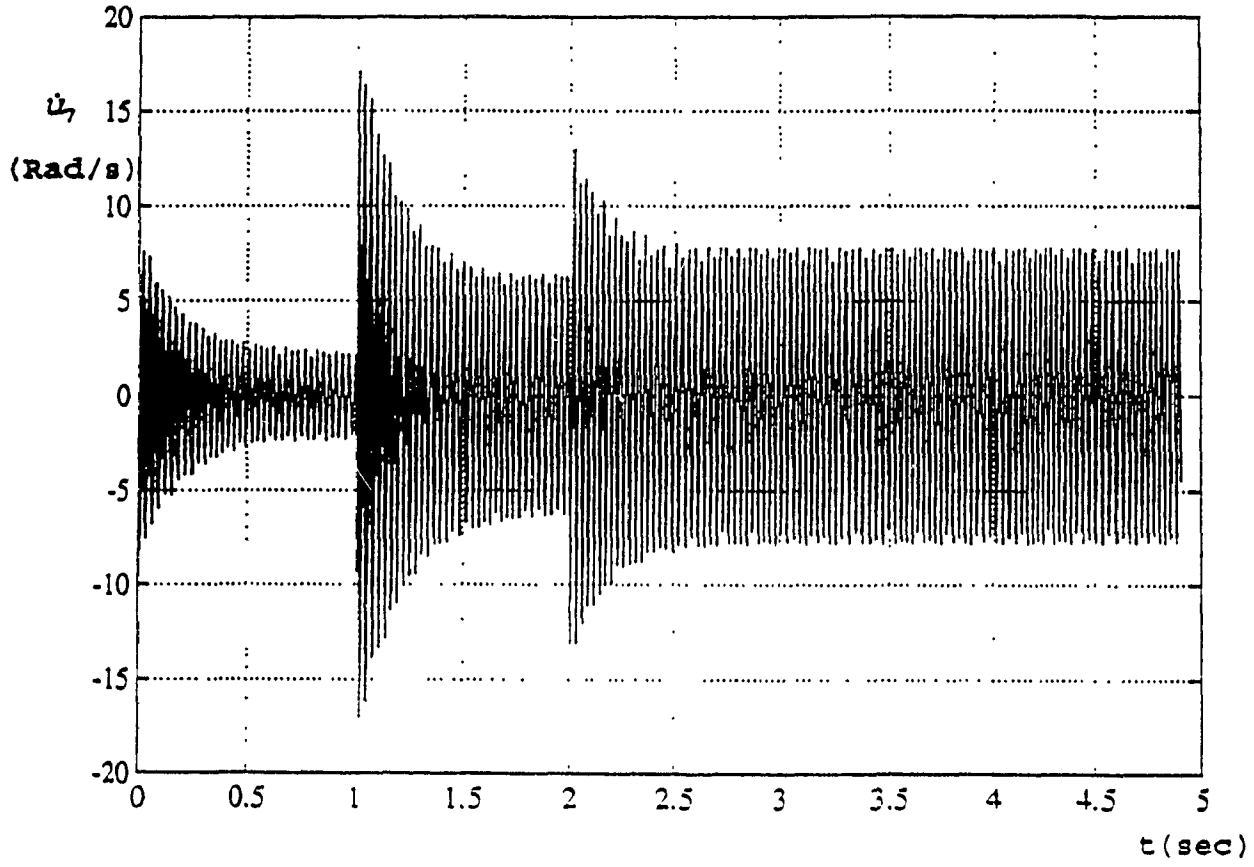


Fig. 4.36 Flexural Angular Velocity at Link 1 Tip in the Second Simulation

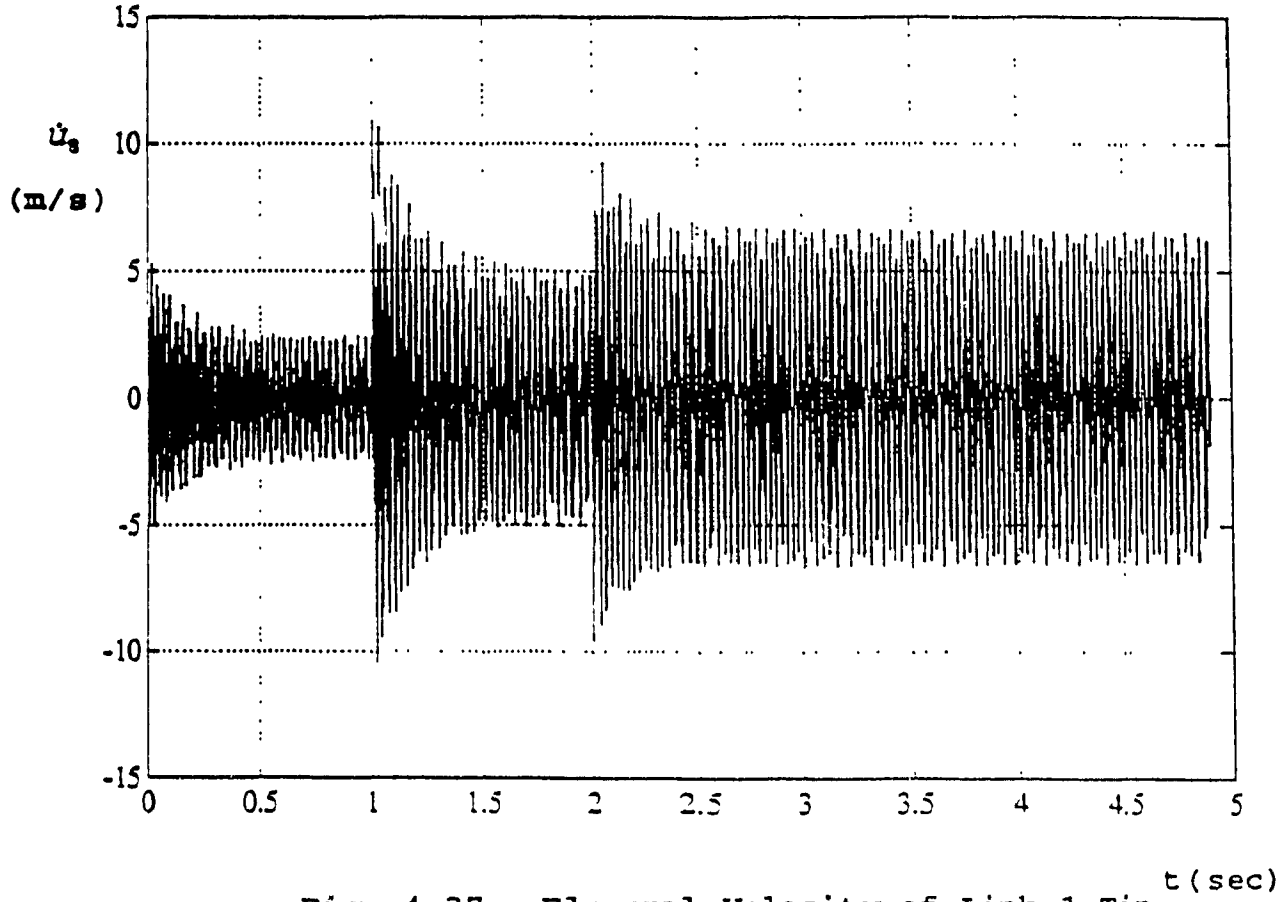


Fig. 4.37 Flexural Velocity of Link 1 Tip in the Second Simulation

CHAPTER 5

CONCLUSIONS AND FUTURE EXTENSION

5.1 INTRODUCTION

In this chapter, a summary of the work achieved and the limitations in its applications are presented. Further, ideas are given for extension of the present method to obtain more explicit expressions for the system variables. The expected challenges for any future extension of this analysis technique are also presented in this chapter.

5.2 SUMMARY OF THE INVESTIGATION

In this thesis, a generalized dynamic model for flexible robot manipulator is set up by finite element approach. As mentioned earlier, some previous works on dynamic modelling for flexible robot manipulator have been carried out by using finite element modelling. However, all these discussions are restricted to certain number of links or finite elements in each link. In order to develop the application of finite element approach to dynamic modelling of flexible robot manipulators, a more generalized formulation representing the dynamic characteristics of flexible robot manipulator with revolute joints and straight links is presented in this thesis. Unlike those works mentioned earlier, the method

presented in this thesis provides a solution for the dynamic model for a flexible robot manipulator with any number of revolute links and finite elements in each link, although in the simulation examples described here included only manipulators with fewer number of links and a small number of finite elements to simplify computation and for comparison. In order to simplify the derivations without loss of accuracy, small deflection is assumed when the manipulator contains more than one link so that the elements of matrix P_i can be linearized.

Through the present study of dynamic modelling of flexible robot manipulator, wherein the model properly describes more closely the real manipulator system, it may be possible to implement a control system as per the design requirement. By considering the torque on each joint as stated in this dynamic model, the position, velocity and acceleration at any location can be obtained. Two examples of simulations are given in this thesis. The first simulation investigates a two-link manipulator system without any torque present at the joints. The responses shown in Fig. 4.4 and 4.5 represent the coupling between two links. Both responses do not maintain a constant amplitude due to the coupling effects between the links.

In the second simulation, one-link revolute joint manipulator is discussed. Here, unlike the first simulation, a torque is exerted at the joint. Because one-link manipulator

is a simple system, it is chosen for the simulation so that the response can be easily understood. The link starts to oscillate under the gravitational force when the torque is relieved.

The work presented in this thesis provides useful information on computer based solution of manipulator dynamic model using Runge-Kutta procedure. The flow chart shown in Fig. 3.1 describes how to simulate the manipulator system. As the transformation matrix are expressed in the recursive form, any transformation matrix can be easily obtained by using equations (2.9) to (2.14). A set of second-order differential equations obtained from dynamic equations (3.47) and (3.48) can be solved by applying appropriate numerical integration. The Runge-Kutta procedure is preferred method in many earlier investigations [7] due to its self-starting property as well as for its reasonable level of accuracy.

5.3 LIMITATION OF PRESENT WORK

Although equations (3.47) and (3.48) represent the general dynamic equation for flexible robot manipulator, they are not arranged into the form as equation (1.1). In equations (3.47) and (3.48), the variables do not explicitly appear. This will bring difficulties to either simulation or control design since it may not be possible to compare the orders of magnitudes of different terms.

The challenge in any future extension will be to deal with the problem how to separate the variables from the dynamic equations.

5.4 BRIEF DESCRIPTION OF FUTURE EXTENSION

The work can comprise of two parts. One is the task of separating the joint variables, and another is question of separating the elastic variables.

First of all, the differential translation and rotation should be expressed in terms of the given coordinate frame. That is, given a coordinate frame A_k, there will exist the following expressions:

$$A_k + dA_k = A_k \cdot Rot(d\theta_k) \tag{5.1}$$

where $Rot(d\theta_k)$ is a transformation relation representing a differential rotation $d\theta_k$ about axis z in the given coordinate frame A_k.

Therefore, dA_k can be given as:

$$dA_k = A_k[Rot(d\theta_k) - 1] = A_k \cdot \Delta_k \tag{5.2}$$

where

$$\Delta_k = Rot(d\theta_k) - 1 \tag{5.3}$$

Let

$$\overline{\Delta}_k = \frac{\Delta_k}{d\theta_k} \tag{5.4}$$

Then the equation (5.2) could be rearranged in the following form:

$$dA_k = A_k \overline{\Delta}_k d\theta_k \tag{5.5}$$

So the derivative of A_k with respect to time may be easily obtained:

$$\frac{dA_k}{dt} = A_k \overline{\Delta}_k \dot{\theta}_k \tag{5.6}$$

The following equation is also obtained based on the previous results:

$$\frac{d^2 A_k}{dt^2} = A_k \overline{\Delta}_k \ddot{\theta}_k + A_k \overline{\Delta}_k^2 \dot{\theta}_k^2 \tag{5.7}$$

From equation (5.1) to (5.7), one may find that the joint variables have been separated from the joint transformation matrix A_k.

As the link transformation matrix has more than one

variable, the differentiation of it is different from what is for joint matrix A_k. By separating the elastic variables from the expression of link transformation matrix D_k, one can have,

$$D_k = D' + Q_x u_{kz3} + Q_y u_{kx3} + Q_z u_{ky3} + Q_1 u_{kx4} + Q_2 u_{ky4} + Q_3 u_{kz4} \qquad (5.8)$$

Here D' is the first part in equation (2.5), u_{kx3}, u_{ky3}, u_{kz3}, u_{kx4}, u_{ky4} and u_{kz4} are the elastic variables, and Q_x, Q_y, Q_z, Q_1, Q_2 and Q_3 are matrices which are defined below.

$$Q_x = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (5.9a)$$

$$Q_y = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (5.9b)$$

$$Q_z = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (5.9c)$$

$$Q_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.9d}$$

$$Q_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.9e}$$

$$Q_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.9f}$$

Therefore, the differentiation of equation (5.8) becomes:

$$\dot{D}_k = \frac{dD_k}{dt} = \frac{dp_k}{dt} = Q_x \dot{u}_{kz3} + Q_y \dot{u}_{kx3} + Q_z \dot{u}_{ky3} + Q_1 \dot{u}_{kx4} + Q_2 \dot{u}_{ky4} + Q_3 \dot{u}_{kz4} \tag{5.10}$$

and

$$\ddot{D}_k = Q_z \ddot{u}_{ky3} + Q_y \ddot{u}_{kx3} + Q_x \ddot{u}_{kz3} + Q_1 \ddot{u}_{kx4} + Q_2 \ddot{u}_{ky4} + Q_3 \ddot{u}_{kz4} \tag{5.11}$$

As the variables have been separated from the joint transformation matrix A_i and link transformation matrix D_k, the transformation matrix T_i which is made up of A_i and D_i could

also be expressed by the separated variables. After substituting T_k and its differentiations into dynamic equations (3.47) and (3.48), the dynamic equations can be arranged in the form of equation (1.1).

As a result of the above arrangement, the whole dynamic equations will become too massive to be handled. Therefore, further simplification have to be introduced. However, this procedure needs far more work and is beyond the scope of this thesis. Also it can be mentioned that a careful experimental verification of the responses may be required to establish a complete validation of the generalized dynamic modelling attempted in this investigation.

REFERENCES

1. Castelazo, I.A. and Lee, H., "Nonlinear Compensation for Flexible Manipulators", Journal of Dynamic Systems, Measurement, and Control, Vol.112, March 1990

2. Huang, Y. and Lee, C.S.G., "Generalization of Newton-Euler Formulation of Dynamic Equations to Nonrigid Manipulators", Journal of Dynamic Systems, Measurement, and Control,Vol.110,September 1988

3. Singh, S.N. and Schy, A.A., "Control of Elastic Robotic Systems by Nonlinear Inversion and Modal Damping", Transactions of the ASME, Vol.108, Sep.1986

4. Desai, C.S., "Elementary Finite Element Method", Prentice-Hall,Inc.,1979

5. Kalra, P. and Sharan, A.M., "Accurate Modelling of Flexible Manipulators Using Finite Element Analysis", Mech. Mach. Theory, Vol.26,No.3,1991

6. King, J.O., Gourishankar, V.G. and Rink, R.E., "Lagrangian Dynamics of Flexible Manipulators Using Angular Velocities Instead of Transformation Matrices", IEEE Transactions on Systems, Man, and Cybernetics,Vol.SMC-17,No.11,November/December 1987

7. Li, C.J. and Sankar, T.S., "A Systematic Method of Dynamics for Flexible Robot Manipulators", Journal of Robotic Systems, Vol.9, No.7, 1992

8. Geradin, M., Robert, G., and Bernardin, C., "Dynamic

Modelling of Manipulators with Flexible Members", Advanced Software In Robotics,Elsevier Science Publishers B.V.(North-Holland),1984

9. Bricout, J.N., Debus, J.C. and Micheau, P., "A Finite Element Model for the Dynamics of Flexible Manipulators", Mech.Mach.Theory, Vol.25, No.1,1990

10. Usoro, P.B., Nadira, R. and Mahil, S.S., "A Finite Element/Lagrange Approach to Modeling lightweight Flexible Manipulators", ASME. J. of Dynamic Systems, Measurement, and Control ,May 19,1986

11. Book, W.J., "Recursive Lagrangian Dynamics of Flexible Manipulator Arms", The Int. J. of Rob. Re. ,Vol.3,No.3,1984

12. Lee, J.L. and Wang, B.L., "Optimal Control of a Flexible Robot Arm", Computers & Structures Vol.29,No.3,1988

13. Lee, J.L. and Wang, B.L., "Dynamic Equations for a Two-link Flexible Robot Arm", Computers & Structures Vol.29,No.3,1988

14. Chang, L.W. and Hamilton, J.F., "Dynamics of Robotic Manipulators with Flexible Links", Journal of Dynamic Systems, Measurement, and Control, Vol.113,March 1991

15. Gaultier, P.E. and Cleghorn, W.L., "Modelling of Flexible Manipulators Employing a Spatially Translating and Rotating Beam Finite Element", Structural Vibration and Acoustics,Vol.34,ASME 1991

16. Naganathan, G. and Soni, A.H., "Coupling Effects of

Kinematics and Flexibility in Manipulators", The Int. Journal of Robotics Research, Vol.6,No.1,Spring 1987

17. Gordaninejad, F., Azhdari, A. and Chalhoub, N.G., "Nonlinear Dynamic Modelling of a Revolute-prismatic Flexible Composite-material Robot Arm", Journal of Vibration and Acoustics,Vol.113,October 1991

18. Jonker, B., "A Finite Element Dynamic Analysis of Flexible Manipulators", The Int. Journal of Robotics Research, Vol.9, No.4, August 1990

19. Chang, L.W. and Hamilton, J.F., "The Kinematics of Robotic Manipulators with Flexible Links Using an Equivalent Rigid Link System (ERLS) Model", Journal of Dynamic Systems, Measurement, and Control,Vol.113, March 1991

20. Zhu, J., "Robotics and Control Technique", Zhejiang University, 1991

21. Riley, K.F., "Mathematical Methods for the Physical Sciences", Cambridge University Press

22. Thomson, W.T., "Theory of Vibration with Applications", Prentice Hall, 1988

APPENDIX

The programs for the two simulations are listed in the following pages. They follow the 4th order Runge-kutta procedure which already was shown in Table 4.2. The coefficients for variables are obtained from equation (3.47) and (3.48) and arranged in the corresponding matrices as given in equation (1.1). Since the simulations are just for cases of robot manipulators with few members only, all the elements of matrices in equation (1.1) are simply carried out by hand instead of constructing another program. In other words, the listed programs execute the task of solving the differential equations which are in the form of equation (1.1). The initial conditions are also specified in the programs.

PROGRAM FOR SIMULATION 1 IN MATLAB

```
g=9.8;
l=0.5;
I=5/10^9;
mu=5;
E=2*10^11;
x(1,1)=-1.570796;
y(1,1)=0;
x(2,1)=0;
y(2,1)=0;
x(3,1)=0;
y(3,1)=0;
x(4,1)=0;
y(4,1)=0;
x(5,1)=0;
y(5,1)=0;
x(6,1)=0.0872664;
y(6,1)=0;
x(7,1)=0;
y(7,1)=0;
x(8,1)=0;
y(8,1)=0;
x(9,1)=0;
y(9,1)=0;
x(10,1)=0;
y(10,1)=0;
t=0;
h=0.0005;

      for j=1:8000,
```

```
X1=x;
Y1=y;
z1;
function;
F1=s;
B1=B;
A1=A;
        X2=x+Y1*h/2;
        Y2=y+F1*h/2;
        z2;
        function;
        F2=s;
        A2=A;
        B2=B;
          X3=x+Y2*h/2;
          Y3=y+F2*h/2;
          z3;
          function;
          F3=s;
          A3=A;
          B3=B;
            X4=x+Y3*h;
            Y4=y+F3*h;
            z4;
            function;
            F4=s;
            A4=A;
            B4=B;

            t=t+h;
            x=x+h/6*(Y1+2*Y2+2*Y3+Y4);
```

```
    y=y+h/6*(F1+2*F2+2*F3+F4);
    fprintf(' dat1.dat','%g       %g',t,x(1));
    fprintf(' dat2.dat','%g       %g',t,x(2));
    fprintf('dat3.dat','%g        %g',t,x(3));
    fprintf('dat4.dat','%g        %g',t,x(4));
    fprintf('dat5.dat','%g        %g',t,x(5));
    fprintf('dat6.dat','%g        %g',t,x(6));
    fprintf('dat7.dat','%g        %g',t,x(7));
    fprintf('dat8.dat','%g        %g',t,x(8));
    fprintf('dat9.dat','%g        %g',t,x(9));
    fprintf('dat10.dat','%g       %g',t,x(10));
end
```

FUNCTION

```
a1=15.2/3*l^3+2*(-2*l*sin(z(16))+cos(z(16))*z(15))*(-0.25/3*l^2*z(18)+..
l*z(19)+0.5*l*z(20))+4*l^2*(2*l*cos(z(16))+sin(z(16))*z(15));

a2=16/420*l^3*z(17)^2+8/420*l^3*z(18)^2+624/420*l*z(19)^2+312/420*l*..
z(20)^2-12/420*l^3*z(17)*z(18)+52/420*l^2*z(17)*z(20)-88/420*l^2*..
z(18)*z(20)-52/420*l^2*z(18)*z(19)+216/420*l*z(19)*z(20);

a3=(z(1)*(-2*l*sin(z(16))+cos(z(16))*z(15))-sin(z(16))*z(5))*2*(-0.25/3*
..
l^2*z(8)+l*z(9)+0.5*l*z(10))+(z(1)*z(6)*(-2*l*cos(z(16))-sin(z(16))*..
z(15))+z(1)*z(5)*cos(z(16))-cos(z(16))*z(5)*z(6))*2*(-0.25/3*l^2*z(18)+.
.
l*z(19)+0.5*l*z(20));
```

```
a4=4*l^2*(z(1)*(-2*l*sin(z(16))*z(6)+sin(z(16))*z(5)+cos(z(16))*..z(15)*
z(6))-sin(z(16))*z(5)*z(6));

a5=(16/420*l^3*z(7)*z(17)+8/420*l^3*z(8)*z(18)+624/420*l*z(9)*z(19)+..
312/420*l*z(10)*z(20)-6/420*l^3*(z(7)*z(18)+z(17)*z(8))+26/420*l^2*..
(z(7)*z(20)+z(17)*z(10))-44/420*l^2*(z(8)*z(20)+z(18)*z(10))-..
26/420*l^2*(z(18)*z(9)+z(8)*z(19))+108/420*l*(z(9)*z(20)+z(19)*z(10)))*.
.
2*(z(3)+z(1)+z(6));

a6=8/420*l^3*z(7)*z(17)+4/420*l^3*z(18)*z(8)+312/420*l*z(19)*z(9)+..
156/420*l*z(10)*z(20)-3/420*l^3*(z(17)*z(8)+z(7)*z(18))+13/420*l^2*..
(z(7)*z(20)+z(17)*z(10))-22/420*l^2*(z(8)*z(20)+z(18)*z(10))-13/420*..
l^2*(z(18)*z(9)+z(8)*z(19))+54/420*l*(z(9)*z(20)+z(19)*z(10));

a7=z(13)*(8/420*l^3*z(7)^2+4/420*l^3*z(8)^2+312/420*l*z(9)^2+..
156/420*l*z(10)^2-..
6/420*l^3*z(7)*z(8)+26/420*l^2*z(7)*z(10)-44/420*l^2*..
z(8)*z(10)-26/420*l^2*z(8)*z(9)+108/420*l*z(9)*z(10));

a8=z(1)^2*(8/420*l^3*z(13)-6/420*l^3*z(12)-44/420*l^2*z(15)-..
26/420*l^2*z(14))+8/3*l^3*(2*z(13)*(z(1)+z(6))^2);

a9=(-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*(-(z(1)^2+z(1)*z(6))*(2*l*..
cos(z(16))..
+sin(z(16))*z(15))-(z(1)+z(6))*cos(z(16))*z(5))+4*l^2*((z(1)^2+..
z(6)*z(1))*(-2*l*sin(z(16))+cos(z(16))*z(15))-(z(1)+z(6))*sin..
(z(16))*z(5));

a10=(8/420*l^3*z(17)^2+4/420*l^3*z(18)^2+312/420*l*z(19)^2+..
156/420*l*z(20)^2-6/420*l^3*z(17)*..
```

```
z(18)+26/420*l^2*z(17)*z(20)-44/420*l^2*z(18)*z(20)-26/420*l^2*..
z(18)*z(19)+108/420*l*z(19)*z(20))*2*..
(z(1)+z(6))^2*z(13);

a11=(0.2/3*l^3*z(7)-0.4/3*l^3*z(8)+l^2*z(9)+0.85*l^2*z(10))*2*..
(z(1)+z(6))*z(13)+(-0.25/3*l^2*z(8)+l*z(9)+0.5*l*z(10))*(z(1)*(-2*l*sin(
..
z(16))+cos(z(16))*z(15))-sin(z(16))*z(5));

a12=-mu*z(13)*(8/420*l^3*z(7)^2+4/420*l^3*z(8)^2+312/420*l*..
z(9)^2+156/420*l*z(10)^2-..
6/420*l^3*z(7)*z(8)+26/420*l^2*z(7)*z(10)-44/420*l^2*z(8)*z(10)-..
26/420*l^2*z(8)*z(9)+108/420*l*z(9)*z(10));

a13=mu*g*(-0.25/3*l^2*cos(z(11))+2*l^2*cos(z(11)+z(16))-sin(z(11)+z(16))
..
*(-0.25/3*l^2*z(18)+l*z(19)+0.5*l*z(20)))+E*I*(4/l*z(13)+2/l*z(12)+6/l^2
*..
z(14)-6/l^2*z(15));

a14=(-0.25/3*l^2*z(8)+l*z(9)+0.5*l*z(10))*(-(z(1)+z(6))*(sin(z(16))+..
cos(z(16))*z(13))-sin(z(16))*z(3));

a15=(-0.25/3*l^2*z(18)+l*z(19)+0.5*l*z(20))*(-(z(1)+z(6))*(cos(z(16))*..
z(6)-sin(z(16))*z(13)*z(6)+cos(z(16))*z(3))-cos(z(16))*z(3)*z(6))..
+2*l^2*((z(1)+z(6))*(-sin(z(16))*z(6)-cos(z(16))*z(13)*z(6)-..
sin(z(16))*z(3))-sin(z(16))*z(3)*z(6));

a16=(-0.25/3*l^2*z(8)+l*z(9)+0.5*l*z(10))*(-cos(z(16))*z(13)*z(6)-sin..
(z(16))*z(3)-sin(z(16))*z(6));
```

```
a17=z(1)^2*(312/420*l*z(15)+26/420*l^2*z(12)-44/420*l^2*z(13)+..
108/420*l*z(14));

a18=(-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*(-(z(1)^2+z(1)*z(6))*(-cos..
(z(16))+sin(z(16))*z(13))+z(1)*cos(z(16))*z(3));

a19=4*l^2*((z(1)^2+z(1)*z(6))*(sin(z(16))+cos(z(16))*z(13))+z(1)..
*sin(z(16))*z(3));

a20=-(1+mu*l)*z(1)^2*z(15)-mu*(-0.25/3*l^2*z(8)+l*z(9)+0.5*l*z(10))*..
(z(1)*(sin(z(16))+cos(z(16))*z(13)));

a21=1.5*mu*g*l*cos(z(11))+E*I*(12/l^3*z(15)-6/l^2*z(12)-6/l^2*z(13)-..
12/l^3*z(14));

a22=0.2/3*l^3*((z(1)+z(6))*2*z(13)*z(3))+(8/420*l^3*z(7)-..
3/420*l^3*z(8)+13/420*l^2*z(10))*z(3)*z(13);

a23=(8/420*l^3*z(17)-3/420*l^3*z(18)+13/420*l^2*z(20))*z(3)^2;

a24=mu*z(13)*z(3)*(16/420*l^3*z(7)-6/420*l^3*z(8)+26/420*l^2*z(10)),

a25=-mu/2*((16/420*l^3*z(17)-6/420*l^3*z(18)+26/420*l^2*z(20))*..
((z(1)+z(6))^2*..
(1+z(13)^2)+z(3)^2+2*(z(1)+z(6))*z(3)));

a26=-mu*((8/420*l^3*z(7)-3/420*l^3*z(8)+13/420*l^2*z(10))*..
z(3)*z(13));

a27=E*I*(8/l*z(17)+2/l*z(18)-6/l^2*z(20));
```

```
a28a=-0.8/3*l^3*(z(1)+z(6))*z(13)*z(3);
a28b=-0.25/3*l^2;
a28c=z(1)*(-2*l*sin(z(16))*z(6)..
+cos(z(16))*z(15)*z(6)+sin(z(16))*z(5)-2*l*cos(z(16))*z(13)*z(6)-2*l*..
sin(z(16))*z(3)-sin(z(16))*z(13)*z(15)*z(6)+cos(z(16))*(z(3)*z(15)+z(13)
*..
z(5)));
a28d=-cos(z(16))*z(5)*z(13)*z(6)-sin(z(16))*z(5)*z(3)-sin(z(16))*z(5)*z(
6);
a28=a28a+a28b*(a28c+a28d);

a29=(4/420*l^3*z(8)-3/420*l^3*z(7)-22/420*l^2*z(10)-13/420*..
l^2*z(9))*z(3)*z(13);

a30=(4/420*l^3*z(18)-3/420*l^3*z(17)-22/420*l^2*z(20)-..
13/420*l^2*z(19))*z(3)^2;

a31=mu*z(13)*z(3)*(8/420*l^3*z(8)-6/420*l^3*z(7)-44/420*l^2*z(10)-..
26/420*l^2*z(9));

a32a=-(z(1)^2+z(6)*z(1))*(2*l*cos(z(16))*z(13)+2*l*sin(z(16))-cos(z(16))
..
*z(15)+sin(z(16))*z(13)*z(15));
a32b=z(1)*(-2*l*sin(z(16))*z(3)+cos(z(16))*z(15)*z(3))..
-(z(1)+z(6))*(z(5)*sin(z(16))+cos(z(16))*z(13)*z(5))-sin(z(16))*z(3)*z(5
);
a32=a32a+a32b;

a33=(8/420*l^3*z(18)-6/420*l^3*z(17)-44/420*l^2*z(20)-26/420*..
l^2*z(19))*((z(1)+z(6)..
)^2*(1+z(13)^2)+z(3)^2+2*(z(1)+z(6))*z(3));
```

```
a34=-mu*(4/420*l^3*z(8)-3/420*l^3*z(7)-22/420*l^2*z(10)-
13/420*l^2*z(9))*z(3)*z(13);

a35=-0.25/3*l^2*(cos(z(16)+z(11))-sin(z(11)+z(16))*z(13))*mu*g+E*l*..
(4/l*z(18)+2/l*z(17)+6/l^2*z(19)-6/l^2*z(20));

a36=2*z(3)*z(13)*l^2*(z(1)+z(6))+l*z(1)*(-2*l*sin(z(16))*z(6)+cos(.
z(16))*z(15)*z(6)+sin(z(16))*z(5)-2*l*cos(z(16))*z(13)*z(6) 2*l*sin(z(1
))..
*z(3)-sin(z(16))*z(13)*z(15)*z(6)+cos(z(16))*z(3)*z(15)+cos(z(16))*z(13)
*z(5));

a37=(-cos(z(16))*z(5)*z(13)*z(6)-sin(z(16))*z(5)*z(3)-sin(z(16))*z(5)*
z(6))*l;

a38=(312/420*l*z(9)-13/420*l^2*z(8)+54/420*l*z(10))*z(3)*z(13);

a39=(312/420*l*z(19)-13/420*l^2*z(18)+54/420*l*z(20))*z(3) 2,

a40=mu*z(3)*z(13)*(624/420*l*z(9)-26/420*l^2*z(8)+108/420*l*z(10)),

a41=-(z(1)^2+z(1)*z(6))*(2*l*cos(z(16))*z(13)+2*l*sin(z(16)) cos(z(16))
.
*z(15)+sin(z(16))*z(13)*z(15));

a42=z(1)*(-2*l*sin(z(16))*z(3)+cos(z(16))*z(15)*z(3))-(z(1)+z(6))*(.
z(5)*sin(z(16))+cos(z(16))*z(13)*z(5))-sin(z(16))*z(3)*z(5),

a43=(1.485714*l*z(19)-0.0619047*l^2*z(18)+108/420*l*z(20))*((z(1)+z(6))^
2*..
(1+z(13)^2)+z(3)..
```

```
^2+2*(z(1)+z(6))*z(3));

a44=-mu*((312/420*l*z(9)-13/420*l^2*z(8)+54/420*l*z(10))*z(3)*z(13));

a45=mu*g*l*(cos(z(11)+z(16))-sin(z(11)+z(16))*z(13))+E*I*(24/l^3*z(19)+..
.
6/l^2*z(18)..
-12/l^3*z(20));

a46=1.7*l^2*z(3)*z(13)*(z(1)+z(6));

a47=z(1)*(-2*l*sin(z(16))*z(6)+cos(z(16))*z(15)*z(6)+sin(z(16))..
*z(5)-2*l*cos(z(16))*z(13)*z(6)-2*l*sin(z(16))*z(3)-sin(z(16))*z(13)*..
z(15)*z(6)+cos(z(16))*z(3)*z(15)+cos(z(16))*z(13)*z(5));

a48=-cos(z(16))*z(5)*z(13)*z(6)-sin(z(16))*z(5)*z(3)-sin(z(16))*z(5)*z(6
);

a49=(156/420*l*z(10)+13/420*l^2*z(7)-22/420*l^2*z(8)+54/420*..
l*z(9))..
*z(3)*z(13);

a50=(156/420*l*z(20)+13/420*l^2*z(17)-22/420*l^2*z(18)+..
54/420*l*z(19))*z(3)^2;

a51=mu*z(3)*z(13)*(312/420*l*z(10)+26/420*l^2*z(7)-44/420*l^2*z(8)+..
108/420*l*z(9));

a52=-(z(1)^2+z(1)*z(6))*(2*l*cos(z(16))*z(13)+2*l*sin(z(16))-cos(z(16)).
.
*z(15)+sin(z(16))*z(15)*..
```

```
z(13))+z(1)*(-2*l*sin(z(16))*z(3)+cos(z(16))*z(15)*z(3));

a53=-(z(1)+z(6))*(z(5)*sin(z(16))+cos(z(16))*z(13)*z(5))-sin(z(16))*z(3)
..
*z(5);

a54=(312/420*l*z(20)+26/420*l^2*z(17)-44/420*l^2*z(18)+108/420*..
l*z(19))*((z(1)+z(6))^2*..
(1+z(13)^2)+z(3)^2+2*(z(1)+z(6))*z(3));

a55=-mu*(156/420*l*z(10)+13/420*l^2*z(7)-22/420*l^2*z(8)+..
54/420*l*z(9))*z(3)*z(13);

a56=0.5*mu*g*l*(cos(z(11)+z(16))-sin(z(11)+z(16))*z(13))+..
E*I*(12/l^3*z(20)-6/l^2*..
z(17)-6/l^2*z(18)-12/l^3*z(19));

a57=16/3*l^3+16/420*l^3*z(12)^2+8/420*l^3*z(13)^2+624/420*..
l*z(14)^2+312/420*l*z(15)^2-..
12/420*l^3*z(12)*z(13);

a58=52/420*l^2*z(12)*z(15)-88/420*l^2*z(13)*z(15)-52/420*l^2*z(13)*..
z(14)+216/420*l*z(14)*z(15);

a59=16/3*l^3*(1+z(13)^2)-4*(-0.25/3*l^2*z(18)+l*z(19)+0.5*l*.
z(20))*(2*l*cos(z(16))*z(13)+..
2*l*sin(z(16))-cos(z(16))*z(15)+sin(z(16))*z(13)*z(15));

a60=8/420*l^3*z(17)^2+4/420*l^3*z(18)^2+312/420*l*z(19)^2+156/420*..
l*z(20)^2-6/420*l^3*z(17)*z(18)+26/420*l^2*z(17)*z(20)-..
44/420*l^2*z(18)*z(20)-26/420*l^2*z(18)*z(19)+108/420*l*z(19)*z(20),
```

```
a61=8*l^2*(2*l*cos(z(16))-2*l*sin(z(16))*z(13)+sin(z(16))*z(15)+cos(z(16
))*..
z(13)*z(15));

a62=16/3*l^3*(1+z(13)^2)-2*(-0.25/3*l^2*z(18)+l*z(19)+0.5*l*..
z(20))*(2*l*cos(z(16))*z(13)+2*l*..
sin(z(16))-cos(z(16))*z(15)+sin(z(16))*z(13)*z(15));

a63=4*l^2*(2*l*cos(z(16))-2*l*sin(z(16))*z(13)+sin(z(16))*z(15)+..
cos(z(16))*z(13)*z(15));

a64=(-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*(-2*l*sin(z(16))+cos(z(16))*..
z(15))+4*l^2*(2*..
l*cos(z(16))+sin(z(16))*z(15));

a65=16/420*l^3*z(2)*z(12)+8/420*l^3*z(3)*z(13)+624/420*l*z(4)*z(14)+..
312/420*l*z(5)*z(15)-..
6/420*l^3*(z(2)*z(13)+z(12)*z(3))+26/420*l^2*(z(2)*z(15)+z(12)*z(5));

a66=-44/420*l^2*(z(3)*z(15)+z(13)*z(5))-26/420*l^2*(z(3)*z(14)+..
z(13)*z(4))+108/420*l*(z(4)*..
z(15)+z(14)*z(5));

a67=-0.5/3*l^2*z(8)+2*l*z(9)+l*z(10);

a68=-(2*z(1)+z(6))*(2*l*cos(z(16))*z(13)+2*l*sin(z(16))-cos(z(16))*z(15)
+..
sin(z(16))*z(13)*z(15))-2*l*sin(z(16))*z(3)+cos(z(16))*..
z(15)*z(3)-z(5)*sin(z(16))-..
cos(z(16))*z(13)*z(5);
```

```
a69=-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20);

a70=-(2*z(1)+z(6))*(2*l*cos(z(16))*z(3)-2*l*sin(z(16))*z(13)*z(6)+..
2*l*cos(z(16))*z(6)-cos(z(16))*z(5)+sin(z(16))*z(15)*z(6)+sin(z(16))*..
z(13)*z(5)+sin(z(16))*z(3)*z(15)+cos(z(16))*z(13)*z(15)*z(6));

a71=-2*l*cos(z(16))*z(3)*z(6)+cos(z(16))*z(5)*z(3)-sin(z(16))*z(15)*z(3)
*..
z(6)-z(5)*cos(z(16))*z(6)-cos(z(16))*z(3)*z(5)+sin(z(16))*z(13)*z(5)*..
z(6);

a72=-2*l*sin(z(16))*z(6)-2*l*sin(z(16))*z(3)-2*l*cos(z(16))*z(13)*z(6)+
.
sin(z(16))*z(5)+cos(z(16))*z(15)*z(6)+cos(z(16))*z(13)*..
z(5)+cos(z(16))*z(3)*..
z(15)-sin(z(16))*z(13)*z(15)*z(6);

a73a=-sin(z(16))*z(5)*z(6)-sin(z(16))*z(3)*z(5)-cos(z(16))*z(13)*z(5)*.
z(6)-2*l*sin(z(16))*z(3)*z(6);
a73b=sin(z(16))*z(3)*z(5)+cos(z(16))*z(3)*z(15)*z(6);
a73=a73a+a73b;

a74=16/420*l^3*z(17)*z(7)+8/420*l^3*z(8)*z(18)+624/420*l*z(9)*z(19)+..
312/420*l*z(10)*z(20)-..
6/420*l^3*(z(7)*z(18)+z(17)*z(8));

a75=26/420*l^2*(z(7)*z(20)+z(17)*z(10))-44/420*l^2*(z(8)*z(20)+.
z(18)*z(10))-26/420*l^2*..
(z(8)*z(19)+z(18)*z(9))+108/420*l*(z(9)*z(20)+z(19)*z(10));

a76=2*(z(1)+z(6))*(1+z(13)^2)+2*z(3);
```

```
a77=4*(z(1)+z(6))*z(13)*z(3);

a78=4*mu*l*z(1)*z(15)*z(5);

a79=(0.2/3*l^3*z(7)-0.4/3*l^3*z(8)+..
l^2*z(9)+0.85*l^2*z(10))*2*z(3)*z(13);

a80=-0.25/3*l^2*z(8)+l*z(9)+0.5*l*z(10);

a81=-2*l*sin(z(16))*z(6)+cos(z(16))*z(15)*z(6)+sin(z(16))*z(5)-2*l*(..
sin(z(16))*z(3)+cos(z(16))*z(13)*z(6))+cos(z(16))*z(3)*z(15)+..
cos(z(16))*z(13)*z(5)..
-sin(z(16))*z(13)*z(15)*z(6);

a82=0.25/3*l^2*sin(z(11))*z(13)-l*sin(z(11))*z(14)-1.5*l*..
sin(z(11))*z(15)+6*l^2*..
cos(z(11));

a83=2*l^2*(cos(z(11)+z(16))-sin(z(11)+z(16))*z(13))-(sin(z(11)+z(16))+..
cos(z(11)+z(16))*z(13))*(-0.25/3*l^2*z(18)+l*z(19)+0.5*l*z(20));

a84=16/3*l^3*(1+z(13)^2)-(-0.5/3*l^2*z(18)+2*l*z(19)+..
l*z(20))*(2*l*cos(z(16))*z(13)+2*l*..
sin(z(16))-cos(z(16))*z(15)+sin(z(16))*z(13)*z(15));

a85=4*l^2*(2*l*cos(z(16))-2*l*sin(z(16))*z(13)+sin(z(16))*z(15)+..
cos(z(16))*z(13)*z(15));

a86=(-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*(-sin(z(16))-cos(z(16))*z(13))+
..
4*l^2*(cos(z(16)..
```

```
)-sin(z(16))*z(13));

a87=32/3*l^3*(z(1)+z(6))*z(13)*z(3);

a88=-0.5/3*l^2*z(8)+2*l*z(9)+l*z(10);

a89=-z(1)*(2*l*cos(z(16))*z(13)+2*l*sin(z(16))-cos(z(16))*z(15)+sin(z(16
))*..
z(13)*z(15))-z(5)*sin(z(16))-cos(z(16))*z(13)*z(5);

a90=-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20);

a91=-z(1)*(2*l*cos(z(16))*z(3)-2*l*sin(z(16))*z(13)*z(6)+2*l*cos(z(16))*
..
z(6)+sin(z(16))*z(15)*z(6)-cos(z(16))*z(5)+cos(z(16))*z(13)*z(15)*z(6)..
+sin(z(16))*z(3)*z(15)+sin(z(16))*z(13)*z(5));

a92=-cos(z(16))*z(5)*z(6)+sin(z(16))*z(13)*z(5)*z(6)-cos(z(16))*z(3)*..
z(5);

a93=z(1)*(-2*l*sin(z(16))*z(6)-2*l*cos(z(16))*z(13)*z(6)-2*l*sin..
(z(16))*z(3)+cos(z(16))*z(15)*z(6)+sin(z(16))*z(5)-sin(z(16))*z(13)*z(15
)*..
z(6)+cos(z(16))*z(3)*z(15)+cos(z(16))*z(13)*z(5));

a94=-sin(z(16))*z(5)*z(6)-cos(z(16))*z(13)*z(5)*z(6)-sin(z(16))*z(3)*..
z(5);

a95=16/420*l^3*z(7)*z(17)+8/420*l^3*z(8)*z(18)+624/420*l*z(9)*z(19)+..
312/420*l*z(10)*z(20);
```

```
a96=-6/420*l^3*(z(7)*z(18)+z(17)*z(8))+26/420*l^2*(z(7)*z(20)+..
z(17)*z(10),-44/420*l^2*..
(z(8)*z(20)+z(18)*z(10))-26/420*l^2*(z(8)*z(19)+z(18)*z(9))+108/420*..
l*(z(9)*z(20)+z(19)*z(10));

a97=2*(z(1)+z(6))*(1+z(13)^2)+2*z(3);

a98=4*(z(1)+z(6))*z(13)*z(3);

a99=0.2/3*l^3*z(7)-0.4/3*l^3*z(8)+l^2*z(9)+0.85*l^2*z(10);

a100=-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20);

a101=-(z(1)^2+z(1)*z(6))*(-2*l*sin(z(16))*z(13)+2*l*cos(z(16))+..
sin(z(16))*z(15)+cos(z(16))*z(13)*z(15))+z(1)*(-2*l*cos(z(16))*..
z(3)-sin(z(16))*..
z(15)*z(3))-(z(1)+z(6))*(z(5)*cos(z(16))-sin(z(16))*z(13)*z(5))-cos(z(16
)..
)*z(3)*z(5);

a102=(z(1)^2+z(1)*z(6))*(-2*l*sin(z(16))-2*l*cos(z(16))*z(13)+cos(..
z(16))*z(15)-sin(z(16))*z(13)*z(15));

a103=(z(1)+z(6))*(-sin(z(16))*z(5)-cos(z(16)..
)*z(13)*z(5))+z(1)*(-2*l*sin(z(16))*z(3)+cos(z(16))*z(3)*z(15))-sin(z(16
))*..
z(3)*z(5);

a104=-0.25/3*l^2*z(8)+l*z(9)+0.5*l*z(10);

a105=z(1)*(-2*l*sin(z(16))+cos(z(16))*z(15)-2*l*cos(z(16))*z(15)-sin(z(1
```

```
6))*..
z(13)*z(15))-cos(z(16))*z(5)*z(13)-sin(z(16))*z(5);

a106=mu*g*(2*l^2*(cos(z(11)+z(16))-sin(z(11)+z(16))*z(13))-(sin(z(11)+..
z(16))+cos(z(11)+z(16))*z(13))*(-0.25/3*l^2*z(18)+l*z(19)+0.5*l*z(20))),

A(1,1)=0.5*mu*(a57+a58+a59+2*(1+z(13)^2)*a60+a61)+2*mu*l*(4*l^2+z(15)^2)
;
A(1,2)=0.2/3*mu*l^3;
A(1,3)=0.5*mu*(15.2/3*l^3+a64+a60*2);
A(1,4)=mu*l^2;
A(1,5)=mu/2*(1.7*l^2+(-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*..
(-sin(z(16))-cos(z(16))..
*z(13))+4*l^2*(cos(z(16))-sin(z(16))*z(13)))+2*l*2*mu*l;

A(1,6)=mu/2*(a62+a63+2*(1+z(13)^2)*a60);
A(1,7)=0.2/3*mu*l^3*(1+z(13)^2);
A(1,8)=mu*(-0.4/3*l^3*(1+z(13)^2)-0.25/3*l^2*(2*l*..
cos(z(16))+sin(z(16))*z(15)-..
2*l*sin(z(16))*z(13)+cos(z(16))*z(13)*z(15)));

A(1,9)=mu*(l^2*(1+z(13)^2)+l*(2*l*cos(z(16))+sin(z(16))*z(15)-2*l*sin(z(
16))*..z(13)+cos(z(16))*z(13)*z(15)));

A(1,10)=mu*(0.85*l^2*(1+z(13)^2)+0.5*l*(2*l*cos(z(16))+sin(z(16))*z(15)-
2*l..
*sin(z(16))*z(13)+cos(z(16))*z(13)*z(15)));

A(2,1)=0.2/3*mu*l^3;
A(2,2)=8/420*mu*l^3;
A(2,3)=-3/420*mu*l^3;
```

```
A(2,5)=13/420*mu*l^2;
A(2,4)=0;
A(2,6)=0;
A(2,7)=0;
A(3,1)=mu/2*(a1+a2);
A(3,2)=-3/420*mu*l^3;
A(3,3)=mu/2*((16/3+8/420)*l^3+a2);
A(3,4)=-13/420*mu*l^2;
A(3,5)=mu/2*(-44/420*l^2-2*sin(z(16))*(-0.25/3*l^2*z(18)+l*z(19)+..
0.5*l*z(20))+..
4*l^2*cos(z(16)));

A(3,6)=mu/2*(16/3*l^3+a2);
A(3,7)=mu*(0.2/3*l^3+(8/420*l^3*z(17)-3/420*l^3*..
z(18)+13/420*l^2*..
z(20))*z(13));

A(3,8)=mu*(-0.4/3*l^3+z(13)*(4/420*l^3*z(18)-3/420*l^3*..
z(17)-22/420*l^2*z(20)-13/420*..
l^2*z(19)));

A(3,9)=mu*(l^2+z(13)*(312/420*l*z(19)-13/420*l^2*z(18)+54/420*l*z(20)));
A(3,10)=mu*(0.85*l^2+z(13)*(156/420*l*z(20)+13/420*l^2*z(17)-..
22/420*l^2*z(18)+..
54/420*l*z(19)));

A(4,1)=mu*l^2;
A(4,2)=0;
A(4,3)=-13/420*mu*l^2;
A(4,4)=312/420*mu*l;
A(4,5)=54/420*mu*l;
```

```
A(4,6)=0;
A(5,1)=mu/2*(1.7*l^2+(-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*..
(-sin(z(16))-cos(z(16)..
)*z(13))+4*l^2*(cos(z(16))-sin(z(16))*z(13)))+2*l*2*mu*l;

A(5,2)=13/420*mu*l^2;
A(5,3)=mu/2*(-44/420*l^2-2*sin(z(16))*(-0.25/3*l^2*z(18)+..
l*z(19)+0.5*l*z(20))+4*l^2*..
cos(z(16)));
A(5,4)=54/420*mu*l;
A(5,5)=0.5*mu*312/420*l+2*mu*l;
A(5,6)=mu/2*((-0.5/3*l^2*z(18)+2*l*z(19)+l*z(20))*(-sin(z(16))-..
cos(z(16))*z(13))+4*l^2*..
(cos(z(16))-sin(z(16))*z(13)));
A(5,7)=0;
A(5,8)=-0.25/3*mu*l^2*(-sin(z(16))*z(13)+cos(z(16)));
A(5,9)=mu*l*(-sin(z(16))*z(13)+cos(z(16)));
A(5,10)=0.5*mu*l*(-sin(z(16))*z(13)+cos(z(16)));
A(6,1)=mu/2*(a84+a85+2*(1+z(13)^2)*a60);
A(6,2)=0;
A(6,3)=(8/3*l^3+a60)*mu;
A(6,4)=0;
A(6,5)=mu/2*a86;
A(6,6)=mu/2*(16/3*l^3*(1+z(13)^2)+2*(1+z(13)^2)*a60);
A(6,7)=0.2/3*mu*l^3*(1+z(13)^2);
A(6,8)=-0.4/3*mu*l^3*(1+z(13)^2);
A(6,9)=mu*l^2*(1+z(13)^2);
A(6,10)=0.85*mu*l^2*(1+z(13)^2);
A(7,1)=0.2/3*mu*l^3*(1+z(13)^2);
A(7,2)=0;
A(7,3)=mu*(0.2/3*l^3+z(13)*(8/420*l^3*z(17)-3/420*l^3*z(15))+..
```

```
13/420*l^2*z(20)));
A(7,6)=0.2/3*mu*l^3*(1+z(13)^2);
A(7,7)=8/420*mu*(1+z(13)^2)*l^3;
A(7,8)=-3/420*mu*l^3*(1+z(13)^2);
A(7,9)=0;
A(7,10)=13/420*mu*l^2*(1+z(13)^2);
A(8,1)=mu*(-0.4/3*l^3*(1+z(13)^2)-0.25/3*l^2*(2*l*cos(z(16))+..
sin(z(16))*z(15)-..
2*l*sin(z(16))*z(13)+cos(z(16))*z(13)*z(15)));
A(8,3)=mu*(-0.4/3*l^3+z(13)*(4/420*l^3*z(18)-3/420*l^3*z(17)-22/420*..
l^2*z(20)-13/420*l^2*z(19)));
A(8,5)=-0.25/3*mu*l^2*(-sin(z(16))*z(13)+cos(z(16)));
A(8,6)=-0.4/3*mu*l^3*(1+z(13)^2);
A(8,7)=-3/420*mu*l^3*(1+z(13)^2);
A(8,8)=4/420*mu*l^3*(1+z(13)^2);
A(8,9)=-13/420*mu*l^2*(1+z(13)^2);
A(8,10)=-22/420*mu*l^2*(1+z(13)^2);
A(9,1)=mu*(l^2*(1+z(13)^2)+l*(2*l*cos(z(16))+sin(z(16))*z(15)-2*l*..
sin(z(16))*z(13)+..
cos(z(16))*z(13)*z(15)));
A(9,3)=mu*(l^2+(312/420*l*z(19)-13/420*l^2*z(18)+54/420*l*z(20))*z(13));
A(9,5)=mu*(l*cos(z(16))-l*sin(z(16))*z(13));
A(9,6)=mu*l^2*(1+z(13)^2);
A(9,8)=mu*(-13/420*l^2*(1+z(13)^2));
A(9,9)=312/420*mu*l*(1+z(13)^2);
A(9,10)=54/420*mu*l*(1+z(13)^2);
A(10,1)=mu*(0.85*l^2*(1+z(13)^2)+0.5*l*(2*l*cos(z(16))+sin(z(16))*z(15)-
2*l..
*sin(z(16))*z(13)+cos(z(16))*z(13)*z(15)));
A(10,3)=mu*(0.85*l^2+(156/420*l*z(20)+13/420*l^2*z(17)-22/420*l^2*..
z(18)+54/420*l*z(19))*z(13));
```

```
A(10,5)=0.5*mu*l*(-sin(z(16))*z(13)+cos(z(16)));
A(10,6)=0.85*mu*l^2*(1+z(13)^2);
A(10,7)=mu*(1+z(13)^2)*13/420*l^2;
A(10,8)=-22/420*mu*l^2*(1+z(13)^2);
A(10,9)=54/420*mu*l*(1+z(13)^2);
A(10,10)=156/420*mu*l*(1+z(13)^2);
C=[0;0;0;0;0;0;0;0;0;0];
B(1,1)=mu/2*(2*z(1)*(a65+a66)+32/3*l^3*(z(1)+z(6))*z(3)*z(13)+..
a67*a68+a69*(a70+a71)+4*l^2*((2*z(1)+z(6))*a72+a73)+(a74+a75)*a76+
a60*a77)+a78+mu*(a79+a80*a81)+mu*g*(a82+a83);

B(2,1)=-mu*z(1)^2*(8/420*l^3*z(12)-3/420*l^3*..
z(13)+13/420*l^2*z(15))+E*I*(8/l*z(12)..
+2*z(13)/l-6/l^2*z(15));

B(3,1)=mu/2*(a3+a4+a5)+mu*(z(3)*a6+a7)-mu/2*(a8+a9+a10)-mu*(a11+a6*z(4))
..
+a12+a13;

B(4,1)=-mu*(z(1)^2*(312/420*l*z(14)-13/420*l^2*z(13)+54/420*l*z(15)))+
mu*g*l*cos(z(11))..
+E*I*(24/l^3*z(14)+6/l^2*z(13)-12/l^3*z(15));

B(5,1)=mu*(a14+a15)+mu*a16-mu/2*(a17+a18+a19)+a20+a21;
B(6,1)=mu/2*(a87+a88*a89+a90*(a91+a92)+4*l^2*(a93+a94)+(a95+a96)*a97+ .
a98*a50)+2*mu*z(3)*z(13)*a99-mu/2*(a100*a101+4*l^2*(a102+a103))-mu*a104*
..
a105+a106;

B(7,1)=mu*(a22+a23)+a24+a25+a26+a27;
B(8,1)=mu*(a28+a29+a30)+a31-mu/2*(-0.5/3*l^2*a32+a33)+a34+a35;
```

```
B(9,1)=mu*(a36+a37+a38+a39)+a40-mu/2*(2*1*(a41+a42)+a43)+a44+a45;
B(10,1)=mu*(a46+0.5*1*(a47+a48)+a49+a50)+a51-mu/2*(1*(a52+a53)+a54)+a55+
..
a56;

C=inv(A)*(C-B);
```

z1

```
z(1)=Y1(1);
z(2)=Y1(2);
z(3)=Y1(3);
z(4)=Y1(4);
z(5)=Y1(5);
z(6)=Y1(6);
z(7)=Y1(7);
z(8)=Y1(8);
z(9)=Y1(9);
z(10)=Y1(10);
z(11)=X1(1);
z(12)=X1(2);
z(13)=X1(3);
z(14)=X1(4);
z(15)=X1(5);
z(16)=X1(6);
z(17)=X1(7);
z(18)=X1(8);
z(19)=X1(9);
z(20)=X1(10);
```

z2

```
z(1)=Y1(1);
z(2)=Y1(2);
z(3)=Y1(3);
z(4)=Y1(4);
z(5)=Y1(5);
z(6)=Y1(6);
z(7)=Y1(7);
z(8)=Y1(8);
z(9)=Y1(9);
z(10)=Y1(10);
z(11)=X1(1);
z(12)=X1(2);
z(13)=X1(3);
z(14)=X1(4);
z(15)=X1(5);
z(16)=X1(6);
z(17)=X1(7);
z(18)=X1(8);
z(19)=X1(9);
z(20)=X1(10);
```

Z3

```
z(1)=Y3(1);
z(2)=Y3(2);
z(3)=Y3(3);
z(4)=Y3(4);
z(5)=Y3(5);
z(6)=Y3(6);
z(7)=Y3(7);
z(8)=Y3(8);
z(9)=Y3(9);
```

```
z(10)=Y3(10);
z(11)=X3(1);
z(12)=X3(2);
z(13)=X3(3);
z(14)=X3(4),
z(15)=X3(5);
z(16)=X3(6);
z(17)=X3(7);
z(18)=X3(8);
z(19)=X3(9);
z(20)=X3(10),
```

z4

```
z(1)=Y4(1);
z(2)=Y4(2);
z(3)=Y4(3);
z(4)=Y4(4);
z(5)=Y4(5);
z(6)=Y4(6);
z(7)=Y4(7);
z(8)=Y4(8);
z(9)=Y4(9);
z(10)=Y4(10);
z(11)=X4(1);
z(12)=X4(2);
z(13)=X4(3);
z(14)=X4(4);
z(15)=X4(5);
z(16)=X4(6);
z(17)=X4(7);
```

```
z(18)=X4(8);
z(19)=X4(9);
z(20)=X4(10);
```

PROGRAM FOR SIMULATION 2 IN MATLAB

```
g=9.8;
l=0.5;
I=5/10^9;
mu=5;
E=2*10^11;
x(1,1)=0;
y(1,1)=0;
x(2,1)=0;
y(2,1)=0;
x(3,1)=0;
y(3,1)=0;
x(4,1)=0;
y(4,1)=0;
x(5,1)=0;
y(5,1)=0;
t=0;
h=0.00045;
torque=35;
      for j=1:10000,

if t>=1 & t<=2,
torque=-35;
end
if t>2,
```

```
torque=0;
end
X1=x;
Y1=y;
t1;
w1;
F1=p;
        X2=x+Y1*h/2;
        Y2=y+F1*h/2;
        t2;
        w1;
        F2=p;
          X3=x+Y2*h/2;
          Y3=y+F2*h/2;
          t3;
          w1;
          F3=p;
            X4=x+Y3*h;
            Y4=y+F3*h;
            t4;
            w1;
            F4=p;
              t=t+h;
              x=x+h/6*(Y1+2*Y2+2*Y3+Y4);
              y=y+h/6*(F1+2*F2+2*F3+F4);
            fprintf(' out0.dat',' %g        %g   ' , t, x(1));
            fprintf(' out1.dat',' %g        %g   ' , t, x(2));
            fprintf(' out2.dat',' %g        %g   ',  t,x(3));
            fprintf(' out3.dat',' %g        %g   ',  t,x(4));
            fprintf(' out4.dat',' %g        %g   ',  t,x(5));
            fprintf(' out5.dat',' %g        %g  ',   t,y(1));
```

```
            fprintf(' out6.dat',' %g        %g  ',  t,y(2));
            fprintf(' out7.dat',' %g        %g ',   t,y(3)),
            fprintf(' out8.dat',' %g        %g ',   t,y(4));
            fprintf(' out9.dat',' %g        %g ',   t,y(5));
end
```

WI

```
a57=5.333333*l^3+0.038094*l^3*q(7)^2+0.0190476*l^3*q(8)^2+1.485714*l*
q(9)^2+0.742857*l*q(10)^2-0.0285714*l^3*q(7)*q(8);

a58=0.1238094*l^2*q(7)*q(10)-0.2095238*l^2*q(8)*q(10)-..
0.1238094*l^2*q(8)*q(9)+0.5142856*l*q(9)*q(10);

a65=0.0380952*l^3*q(2)*q(7)+0.0190476*l^3*q(8)*q(3)+..
1.485714*l*q(9)*q(4)+0.742857*l*q(5)*q(10)-0.0142857*l^3*(q(2)*q(8)+..
q(7)*q(3))+0.0619047*l^2*(q(2)*q(10)+q(7)*q(5));

a66=-0.1047619*l^2*(q(3)*q(10)+q(8)*q(5))-0.0619047*l^2*(q( )*..
q(9)+q(8)*q(4))+0.2571428*l*(q(4)*q(10)+q(9)*q(5));

A(1,1)=0.5*mu*(a57+a58);
A(1,2)=0.06666666667*mu*l^3;

A(1,3)=-0.1333333333*l^3*mu;
A(1,4)=mu*l^2;
A(1,5)=mu/2*(1.7*l^2);
A(2,1)=0.066666666667*mu*l^3;
A(2,2)=0.0190476*mu*l^3;
A(2,3)=-0.00714285*mu*l^3;
A(2,5)=0.0309523*mu*l^2;
```

```
A(2,4)=0;
A(3,1)=mu/2*(-0.266666667*l^3);
A(3,2)=-0.00714285*mu*l^3;
A(3,3)=mu/2*(0.0190476*l^3);
A(3,4)=-0.0309523*mu*l^2;
A(3,5)=mu/2*(-0.1047619*l^2);
A(4,1)=mu*l^2;
A(4,2)=0;
A(4,3)=-0.0309523*mu*l^2;
A(4,4)=0.742857*mu*l;
A(4,5)=0.1285714*mu*l;
A(5,1)=mu/2*(1.7*l^2);
A(5,2)=0.0309523*mu*l^2;
A(5,3)=mu/2*(-0.1047618*l^2);
A(5,4)=0.1285714*mu*l;
A(5,5)=0.5*mu*0.742857*l;

B(1,1)=0;
B(2,1)=-mu*q(1)^2*(0.0190476*l^3*q(7)-0.00714285*l^3*q(8)..
+0.0309523*l^2*q(10));
B(3,1)=-0.5*mu*q(1)^2*(0.0190476*l^3*q(8)-0.0142857*l^3*q(7)..
-0.1047618*l^2*..
q(10)-0.0619046*l^2*q(9));
B(4,1)=-mu*q(1)^2*(0.742857*l*q(9)-0.0309523*l^2*q(8)+..
0.1285*l*q(10));
B(5,1)=-mu*q(1)^2*(0.37*l*q(10)+0.031*l^2*q(7)-0.0525*..
l^2*q(8)+..
0.1285714*l*q(9));

C(1,1)=mu*g*(2*l^2*cos(q(6))+0.083*l^2*sin(q(6))*q(8)-l*..
sin(q(6))*q(9)-l/2*sin(q(6))*q(10));
```

```
C(2,1)=E*I*(8/l*q(7)+2*q(8)/l-6/l^2*q(10));
C(3,1)=-0.083*mu*g*l^2*cos(q(6))+E*I*(4/l*q(8)+2*q(7)/..
l+6/l^2*..
q(9)-6/l^2*q(10));
C(4,1)=mu*g*l*cos(q(6))+E*I*(24/l^3*q(9)+6/l^2*q(8)-..
12/l^3*q(10));
C(5,1)=mu*g*l*cos(q(6))/2+E*I*(12/l^3*q(10)-6/l^2*q(7)-..
6/l^2*..
q(8)-12/l^3*q(9));

D(1,1)=mu*q(1)*(a65+a66);
D(5,1)=0;

G(1,1)=torque;
G(5,1)=0;

p=inv(A)*(G-B-C-D);
```

T1

```
q(1,1)=y(1,1);
q(2,1)=y(2,1);
q(3,1)=y(3,1);
q(4,1)=y(4,1);
q(5,1)=y(5,1);
q(6,1)=x(1,1);
q(7,1)=x(2,1);
q(8,1)=x(3,1);
q(9,1)=x(4,1);
q(10,1)=x(5,1);
```