1987 IEEE Conference on Robotics and Automation
March 30 - April 2, 1987
Raleigh, NC

# SYMBOLIC MODELING OF FLEXIBLE MANIPULATORS

Sabri Cetinkunt and Wayne J. Book

George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

## ABSTRACT

This paper presents a new systematic algorithm to symbolically derive the full nonlinear dynamic equations of motion of multi-link flexible manipulators. Lagrange's-Assumed modes method is the basis of the new algorithm and adapted in a way suitable for symbolic manipulation by digital computers. It is aaplied to model a two-link flexible arm via a commercially available symbolic manipulation program. The advantages of the algorithm and simulation results are discussed.

## I. Introduction :

Dynamics of a typical industrial manipulator, with six degrees of freedom, is governed by coupled highly nonlinear ordinary differential equations. These equations present a very complicated problem in control system design, mainly because the knowledge in nonlinear control system theory is very limited. Traditionally independent servo controllers are designed based on the assumption that nonlinear coupling terms are negligible. However, this assumption is reasonable and the control system performance may be satisfactory only if the speed of manipulator is "relatively slow". Increasing demand for higher industrial productivity requires manipulators that move faster and more accurately. As a result, the speed of manipulators must increase and the independent linear servo controllers, designed based on the slow motion dynamics, will perform unsatisfactorily.

In recent years there has been considerable progress in the adaptive control of robotic manipulators. Computed torque based methods are aimed at better performance by designing controllers based on more accurate models. Ultimately the performance and the capabilities of a system, i.e. maximum speeds etc., are limited by the initial design of the overall system. A control system at best can utilize these capabilities in an optimum manner. In other words no control law can make the system move at speed which can not be afforded by the existing actuators. Apparently one way of designing manipulators that can move faster is to increase the actuator sizes. However, since actuators themselves are carried by the other actuators, increasing size also increases the effective inertia resulting in a very massive structure. Thus this approach can be quickly self defeating and is not the ultimate answer. The next option is to design light weight systems. Light-weight systems could have the following advantages: higher speed of operation, less overall cost, less energy consumption, smaller actuator sizes, higher productivity. The drawback of such systems is the structural flexibility which deteriorates the accuracy and repeatability. Rigid body dynamic analysis will no longer be accurate and controllers based on this will not perform satisfactorily. Flexibility has to be included in the analysis.

### Background:

Modeling and control of a single link flexible arm [Fig.3] has been investigated by many authors [1,2,-3,4]. The system is essentially modeled as Bernoulli-Euler beam and vibration coordinates are approximated by a finite number of assumed mode shapes. This allows the application of the whole finite dimensional linear control theory to the problem.

The multi-link flexible manipulator [Fig.1, and 4] modelling and control problem has not been researched as much as single -link case. First of all, the modelling problem is not a trivial one. Due to coupling between links, large configuration changes, and high speeds, the system can no longer be accurately represented by simple beam equations. An accurate dynamic model of a light weight arm involves highly complicated algebraic manipulations and can become impossible to deal with by hand calculations. Moreover, the possiblity of making errors along the way is very high. Making some changes in an existing model also requires long algebraic calculations. There are two basic methods used in the modelling : 1. Lagrange's-Finite Element based methods, 2. Lagrange's- Assumed mode based methods. The end result of these methods are essentially the same. Many of the finite element based works on the analysis of closed chain mechanisms can be applied to the dynamic modelling of multi-link flexible arms [5,6].

In [7,8] the nominal joint variable time histories are assumed to be known and the small vibrational dynamic model of the manipulators and mechanisms about nominal motions are developed. In [9] this assumption is removed and full dynamic model is derived. The main advantages of this method are : a) very systematic, b) Can be applied to complex shaped systems, applicable to a very wide class of

This research was supported in part by NASA under grant NAG-1-623.

problems. The disadvatages are: a) requires a substantial amount of software organization, b) results in constrained model, c) does not give much insight to the dynamic structure of the system. Static deflection modes are included in the modes to improve the accuracy of models with limited number of mode shapes [6] . Usuro et.al. investigated the performance of LQR with prescribed degree of stability on a two-link planar arm by digital simulations [10].

The Lagrangian - Assumed modes method is used in the modeling of a two-link robotic manipulator in [11]. Distributed frequency domain analysis of non-planar manipulators using transfer-matrices has been developed in [12]. A recursive method using homogeneous transformation matrices to generate full coupled nonlinear dynamics of multi-link flexible manipulators is presented at [13].

It was experienced by the authors that the application of this technique to multi-link manipulators works well, but with an important drawback: **Algebraic complexity of intermediate steps.** When carried out by hand the length of expressions becomes very large and very time consuming. In addition, the possibility of making algebraic errors was quite high. On the other hand modelling method is easy to understand, recursive, does not require any dedicated special software and derives the full nonlinear dynamic model.

The symbolic manipulation programs are the answer to eliminate the major drawback of the method. Symbolic modelling allows one to model systems with large orders in a very short time, check the elements of the dynamic equations in explicit forms and manipulate them very conveniently.

The remaining part of this paper is organized as follows:
Section II summarizes the Lagrangian - Assumed Modes method. Section III presents a new algorithm which adapts this method to a form suitable for symbolic manipulation by digital computer. At section IV, the algorithm is applied to a two-link flexible arm example. Application details and simulation results are discussed.

## II. Lagrangian - Assumed Modes Method :

**Kinematics :** The first step in dynamic modelling of any mechanical system is to establish the kinematical relationships and be able to define fundamental vector quantities: position, velocity and acceleration. Consider the kinematic structure shown in [Fig.1] representing a manipulator with serial links and joints. Let the coordinate systems used for kinematics of the system be;

$O_0XYZ$ - Fixed to base ( Global Coordinate Frame )

$O_ixyz$ - Fixed to the base of the link i

$O'_ixyz$ - Fixed to the end of link i

If arms are rigid then $O'_i$ xyz coordinates are not needed . The position vector of any point on link i can be expressed with respect to $O_ixyz$ as ;

$$^{i}h(x_i) = [x_i,0,0,1]^T + [w_x(x_i,t),w_y(x_i,t),w_z(x_i,t),0]^T \quad (2.1)$$

where , $w_x(x_i,t)$, $w_y(x_i,t)$, $w_z(x_i,t)$ are displacements of the flexible arm due to flexibility in respective directions. The dependence of w's on the spatial coordinates makes the system infinite dimensional, leading to coupled Ordinary and Partial differential equations of motion. In general these are approximated by finite series consisting of spatial variable dependent functions multiplied by time-dependent generalized coordinates. Once the number of generalized coordinates to be used to represent the distributed flexibility of each link has been decided on, w's can be approximated as;

$$w_\beta(x_i,t)= \sum_{j=1}^{n_i} \phi_{\beta j}(x_i)\ \delta_j(t) \quad ; \beta : x,y,z \quad (2.2)$$

where $n_i$ is the number of assumed mode shapes used for link i for the $w_\beta$, $\phi_{\beta j}(x_i)$ are assumed mode shape functions from an admissable class, $\delta_j(t)$ are the generalized coordinates of approximation, $^{i}h(x_i)$ is uniquely defined. Next we need to be able to transfer this position vector with respect to global coordinate frame to obtain absolute position vector. Let $^{o}W_i$ be the homogeneous matrix transformation from moving coordinate frame $O_ixyz$ to fixed inertial frame $O_oXYZ$. Then the absolute position vector, [Fig.2]

$$^{o}h(x_i) = {^{o}W_i} \ . \ {^{i}h(x_i)} \quad (2.3)$$

It is clear that the transformation $^{o}W_i$ consists of two parts: joint variables and flexible deflections. More clearly, [Fig.1]

$$^{o}W_i = {^{o}W_{i-1}} \ . \ E_{i-1} . \ A_i \quad (2.4)$$

where
$A_i$ - the transformation between $O_ixyz$ and $O'_{i-1}xyz$ - join transformation
$E_{i-1}$ - the transformation from the end of the link coordinates to link base coordinates.
$^{o}W_{i-1}$ - the total transformation to the base coordinates.
The form of these transformation matrices are ;

$$^{j}W_i = \begin{bmatrix} ^{j}R_i & \begin{matrix} x_j \text{ component of } O_i \\ y_j \text{ component of } O_i \\ z_j \text{ component of } O_i \end{matrix} \\ 0^T & 1 \end{bmatrix} ;(2.5)$$

$^{j}R_i$ is (3x3) matrix of direction cosines, $0^T$ (1x3);

$$E_i = \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \sum_{j=1}^{n_i} \delta_{ij}(t) \begin{bmatrix} 0 & -\theta_{zij} & \theta_{yij} & x_{ij} \\ \theta_{zij} & 0 & -\theta_{xij} & y_{ij} \\ -\theta_{yij} & \theta_{xij} & 0 & z_{ij} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(2.6)

where $\theta_{\beta ij}$'s are rotation components of link i due to mode j, assuming small rotations due to flexible deflections, and $l_i$ is the length of the link i.

Once the kinematic description of the system is set up, the process of obtaining the equations of motion is as follows:

1. Pick generalized coordinates (natural choices are joint variables and a finite number of assumed modes series approximation for every flexible element)
2. Form the kinetic, and potential energy, and virtual work for the system
3. Take the necessary derivatives of the Lagrangian Equations and assemble the equations.

If system has $N_j$ number of joints with single degree of freedom and $N_l$ number of flexible elements with $n_i$ modal coordinate for each element, the dynamic model of the system will be governed by

$$N_j + \sum_{i=1}^{N_l} n_i \qquad (2.7)$$

set of coupled second order ordinary differential equations.

**III. Symbolic Implementation of Lagrangian- Assumed Modes Method:**

Although Lagrangian - Assumed modes method is theoretically very well understood and documented [13], it is not quite in a form suitable for symbolic implementation on a digital computer, i.e. insufficient memory problems are likely to occur. Here the modelling method is adapted to overcome this difficulty. Let us first specify some desired features of a modelling algorithm. First, the mode shapes and the mode shape dependent parameters should be easily varied by the analyst. The selection of "appropriate" or "best" mode shapes [Fig.5] for a given flexible system is not a clearly answered problem [12]. One should be able to easily simulate the effect of different mode shapes on the system behavior. For the case of a simple beam under bending vibrations the mode shapes effectively determine the natural frequencies of the system. Effective mass and spring matrix elements are functions of mode shapes as; (with simple boundary conditions )

$$m_{ij} = \int_0^{l_i} \rho\, A(x)\, \phi_i(x)\, \phi_j(x)\, dx \qquad (3.1)$$

$$k_{ij} = \int_0^{l_i} E\, I(x)\, \phi_i''(x)\, \phi_j''(x)\, dx \qquad (3.2)$$

If mode shapes are orthonormalized such that $m_{ij} = 1$ for i=j and 0 , for i≠j, then $k_{ij} = w_i^2$ for i=j, 0 for i≠j. The most accurate way is to update the mode shapes as the boundary conditions of the links vary as function of controller action.

Second, a recursive algorithm would be very desirable. For instance, when the number of modal coordinates increased or additional links included, the dynamic modelling process should not be repeated all over.

The equations governing the dynamics of the system are given by;

$$\frac{d}{dt}\frac{(\partial \Sigma KE)}{\partial \dot{q}_i} - \frac{(\partial \Sigma KE)}{\partial q_i} + \frac{(\partial \Sigma PE)}{\partial q_i} = Q_i \qquad (3.4)$$

where;

$$\Sigma KE = \sum_{i=1}^{N} (KE)_i \quad ; \quad \text{N: total number of discrete}$$

element in the system ( joints, links, payload ).

$$\Sigma PE = \sum_{i=1}^{N} ( PE)_{i\ \text{gravitational}} + (PE)_{i\ \text{elastic}} \qquad (3.5)$$

$q_i$'s are the generalized coordinates which are joint variables and flexible generalized coordinates of flexible elements.

Kinetic energies for rotary joints, if considered as mass with rotarty inertia about the axis of rotation

$$(KE)_{\text{joint } i} = 1/2\, m_i\, V_{gi}^2 + 1/2\, H_{gi} \cdot w_i \qquad (3.6)$$

where $m_i$ is the mass of joint i, $V_{gi}$ is the speed of joint i mass center, $H_{gi}$ is angular momentum vector of joint with respect to its center of mass, $w_i$ is the total angular velocity vector of the joint.

Kinetic energy of the flexible links;

$$( KE )_i = 1/2 \int_0^{l_i} \rho_i(x)\, ( \dot{\vec{r}}_i \cdot \dot{\vec{r}}_i )\, dx \qquad (3.7)$$

If all the modal coordinates and associated mode shapes were given, then the integration over the spatial variable could be evaluated. However since the mode shapes and dependent parameters are desired to be inputed by the user for analysis purposes, we identify all possible elements that are function of spatial variable and assign them parametric names. From (2.3)

$$^{o}\dot{h}_i(x) = {}^{o}\dot{W}_i\, {}^{i}h_i(x) + {}^{o}W_i\, {}^{i}\dot{h}_i(x) \qquad (3.8)$$

$$\dot{\vec{r}}_i \cdot \dot{\vec{r}}_i = {}^{o}\dot{h}_i^T(x) \cdot {}^{o}\dot{h}_i(x)$$

$$= {}^{i}h_i^T(x)\, {}^{o}\dot{W}_i^T\, {}^{o}\dot{W}_i\, {}^{i}h_i(x) + {}^{i}\dot{h}_i^T(x)\, {}^{o}W_i^T\, {}^{o}\dot{W}_i\, {}^{i}h_i(x) + {}^{i}h_i^T(x)\, {}^{o}\dot{W}_i^T\, {}^{o}W_i\, {}^{i}\dot{h}_i(x) + {}^{i}\dot{h}_i^T(x)\, {}^{o}W_i^T\, {}^{o}W_i\, {}^{i}\dot{h}_i(x) \qquad (3.9)$$

where;

$${}^{i}h_i^T(x) = [x + \Sigma\phi_{xij}(x)\, \delta_{xij}(t),\ \Sigma\phi_{yij}(x)\, \delta_{yij}(t),\ \Sigma\phi_{zij}(x)\, \delta_{zij}(t),\ 1\ ]$$

$${}^{i}\dot{h}_i^T(x) = [\ \Sigma\phi_{xij}(x)\, \dot{\delta}_{xij}(t),\ \Sigma\phi_{yij}(x)\, \dot{\delta}_{yij}(t),$$

$$\Sigma\phi_{zij}(x)\,\dot{\delta}_{zij}(t)\ ,\ 0\ ] \qquad (3.10)$$

Elements of the transformations ${}^{o}W_i$ and ${}^{o}\dot{W}_i$ are functions of the generalized coordinates and parameteres of the links k<i, such as { $\theta_i$, $\theta_i\theta_k(t)$, $\phi_{\beta kj}(l_k)$, $\delta_{\beta kj}(t)$, $\theta_k(t)$, where k=1,..i-1, $\beta$:x,y,z}, $l_k$ is the lenght of link k.

In general for serial link robotic manipulators, the kinetic energy of link i will have the following form ; (.) is used to indicate the possible existence of terms that are independent of spatial variable x . At this point, from symbolic modeling point of view it is not important what these (.) terms are. But what is important is to extract all the possible combination of spatial-variable dependent terms and replace them with symbolic names so that <u>the first</u> objective of the modeling is accomplished.

$$
\begin{aligned}
(K.E)_i &= (.)\textstyle\int \rho(x)\,dx + (.)\int \rho(x)\,x\,dx + (.)\int \rho(x)\,x^2\,dx \\
&+ \Sigma\,\Sigma \textstyle\int \rho(x)\,\phi_{\beta ij}(x)\,\phi_{\xi ik}(x)\,dx\ . \\
&[(.)\,\dot{\delta}_{\beta ij}\,\dot{\delta}_{\xi ik} + (.)\,\dot{\delta}_{\beta ij}\,\delta_{\xi ik} + (.)\,\delta_{\beta ij}\,\delta_{\xi ik}\ ] + \\
&\Sigma\,\Sigma \textstyle\int \rho(x)\,\phi_{\beta ij}(x)\,x\,dx\ [(.)\,\delta_{\beta ij} + (.)\,\dot{\delta}_{\beta ij}\ ] + \\
&\Sigma\,\Sigma \textstyle\int \rho(x)\,\phi_{\beta ij}(x)\,dx\ [(.)\,\delta_{\beta ij} + (.)\,\dot{\delta}_{\beta ij}\ ]
\end{aligned} \qquad (3.11)
$$

where; $\beta$ and $\xi$ : x,y,z , j=1,...$m_i$ . At the calculation of absolute velocity of differential element of a flexible member, the parameters which are function of spatial variable can be extracted and be given symbolic names by the symbolic manipulation program very easily. These parameters represent the elements in the dynamic model which are function of mode shapes, and link length, and mass distribution of the flexible element.

Replace in (3.9)

$$
\begin{aligned}
&nm\beta\xi ij \leftarrow \phi_{\beta ij}(x)\,\phi_{\xi ij}(x)\ ,\quad nw\beta ij \leftarrow \phi_{\beta ij}(x)\,x\ , \\
&nq\beta ij \leftarrow \phi_{\beta ij}(x) \\
&m_i \leftarrow 1\ .,\quad m_i\,l_i/2 \leftarrow x\ ,\quad J_{oi} \leftarrow x^2
\end{aligned} \qquad (3.12)
$$

and in the simulation level evaluate these terms by multiplying with $\rho(x)$ and integrating over the link length.

$$
\begin{aligned}
&nm\beta\xi ij = \textstyle\int \rho(x)\,\phi_{\beta ij}(x)\,\phi_{\xi ij}(x)\,dx\ , \\
&nw\beta ij = \textstyle\int \rho(x)\,\phi_{\beta ij}(x)\,x\,dx\ ,\quad nq\beta ij = \int \rho(x)\,\phi_{\beta ij}\,dx \\
&m_i = \textstyle\int \rho(x)\,dx\ ,\quad m_i\,l_i/2 = \int \rho(x)\,x\,dx\ , \\
&J_{oi} = \textstyle\int \rho(x)\,x^2\,dx\ ;
\end{aligned} \qquad (3.13)
$$

There are <u>six basic parameters</u> related to the inertia properties of the flexible element and with their use there is no longer spatial dependence in the kinetic energy expressions. With this approach one can see more explicitly the effect of mode shapes and system parameters on the dynamic model, leading to a better understanding of the dynamics, which is not offered by numerical or other modelling methods. Notice that if the mode shapes associated with a coordinate ( i.e. y ) are chosen to be orthonormal with respect to distributed mass and flexibility many of the above terms will be zero, such as nm$\beta\xi$ij = 1 if i=j, 0 if i≠j .

Similiarly for the elastic potential energy of the link i ( gravitational potential energy is omited here to save space )

$$
\begin{aligned}
(P.E.)_i = 1/2\ \Sigma\textstyle\int\ &(EI_y(\phi''_{yij}(x)\,\phi''_{yik}(x)\,\delta_{yij}(t)\,\delta_{yik}(t)) \\
&EI_z\ (\ \phi''_{zij}(x)\,\phi''_{zik}(x)\,\delta_{zij}(t)\,\delta_{zik}(t)\ ) + \\
EA(x)\ (\ &\phi'_{xij}(x)\,\phi'_{xik}(x)\,\delta_{xij}(t)\,\delta_{xik}(t)\ ))dx\ ;
\end{aligned} \qquad (3.14)
$$

Similiarly

$$k\beta ijk = \int_o^{li} EI_\beta(x)\,\phi''_{\beta ij}(x)\,\phi''_{\beta ik}(x)\,dx\ ;$$

$$\beta\ :\ y,z\ \text{and}\ j,k=1,n_i$$

$$kxijk = \int_o^{li} EA(x)\,\phi'_{xij}(x)\,\phi'_{xik}(x)\,dx\ ;$$

$$
\begin{aligned}
(P.E.)_i = 1/2\ \Sigma\,\Sigma\ [\ &k\beta ijk\ \delta_{\beta ij}(t)\,\delta_{\beta ik}(t) + \\
&kxijk\ \delta_{xij}(t)\,\delta_{xik}(t)\ ]
\end{aligned} \qquad ;\ (3.15)
$$

Now the next important topic is the development of a recursive method which will not run into memory problems as the system dimension get large. Moreover once a model is developed, some variations of the model should be possible without repeating the whole modelling process. As the system dimension gets larger, carrying out the derivations using total energy expressions can easily run into memory problems. Thus

$$\frac{d}{dt}\frac{\partial}{\partial \dot{q}_i}(\ \Sigma\ KE_j) - \frac{\partial}{\partial q_i}(\ \Sigma\ KE_j\ ) + \frac{\partial}{\partial q_i}(\Sigma PE_j\ ) = Q_i \qquad (3.16)$$

$$\Sigma\ (\ \frac{d}{dt}\frac{\partial}{\partial \dot{q}_i}(KE_j) - \frac{\partial}{\partial q_i}(KE_j) + \frac{\partial}{\partial q_i}(PE_j)) = Q_i \qquad (3.17)$$

Due to serial nature of manipulator arm ;

$$\frac{\partial}{\partial \dot{q}_i}(KE_j) = \frac{\partial}{\partial q_i}(KE_j) = \frac{\partial}{\partial q_i}(PE_j) = 0.\quad \text{for } i > j \qquad (3.18)$$

The equations of motion of the system are found to be;

$$\sum_{j=1}^{N}(\frac{d}{dt}\frac{\partial}{\partial q_i}(KE_j) - \frac{\partial}{\partial q_i}(KE_j) + \frac{\partial}{\partial q_i}(PE_j)) = Q_i\ ;$$

$$\text{for } i=1 \text{ to } j\ ; \qquad (3.19)$$

**Algorithm :**

For $j = 1$ to N
  For i=1, to j
    Find and store $KE_j$ , $PE_j$  (3.11) and (3.15)

$$\frac{d}{dt}\left(\frac{\partial}{\partial \dot{q}_i}(KE_j)\right), \quad \frac{\partial}{\partial q_i}(KE_j), \quad \frac{\partial}{\partial q_i}(PE_j)$$

  Next i
Next j

Given all the non-zero derivatives substitute these to equation (3.19) and assemble the equations in a convenient form for simulations and analysis purposes. After the equations are assembled, it is very easy to program them in one of the standard scientific programming languages using the capabilities of the commercial symbolic manipulation packages.

Let us assume that after modelling a manipulator , it is desired to add another link to the model with $n_i$ degrees of freedom. Based on the above algorithm one must evaluate ;

For i=1, to $N + n_i$

$$\frac{d}{dt}\frac{\partial}{\partial \dot{q}_i}(KE_{N+1}), \quad \frac{\partial}{\partial q_i}(KE_{N+1}), \quad \frac{\partial}{\partial q_i}(PE_{N+1}) \tag{3.20}$$

Next i

Let us assume previous model was assembled in the form;

$$[\,M\,]\ \ddot{q} - f = Q \tag{3.21}$$

where the inertia matrix dimension is (NxN) , q , f, Q vector dimensions are (Nx1), N is the total number of generalized coordinates up to that point.

The result of additional link contribution is of form

$$\begin{bmatrix} m_{nn} & m_{nn+1} \\ m_{nn+1} & m_{n+1} \end{bmatrix} \begin{bmatrix} q_{nn+1} \\ q_{n+1} \end{bmatrix} + \begin{bmatrix} f_{nn+1} \\ f_{n+1} \end{bmatrix} = \begin{bmatrix} Q_{nn+1} \\ Q_{n+1} \end{bmatrix} \tag{3.22}$$

where the inertia matrix is of dimension $(N+n_i)$ x $(N+n_i)$ and the vector quantities are of $(N+n_i)$ x 1 . Partition of the equation (3.22) is made such that it would clearly reflect the increase in the dimension of the system compared to one step before. Total equation of motion is obtained by the addition of (3.22) to (3.21), where (3.21) is extended to (3.22) dimensions by addition of zeros at appropriate dimensions.

The implementation adapted here has the following advantages: a) Memory problems are not likely to occur, b) all unnecessary derivatives avoided, c) It is recursive, and d) Mode shape and dependent parameters can be easily varied.

**VI.Application and Discussion of Simulation Results:**

Here the described modeling method is applied to a two-link planar flexible arm, with rotary joints and payload. Two mode shapes for each link are considered to represent the structural flexibilities. As noted earlier, mode shapes can be input into the simulation program and the effect of different mode shapes on the dynamic response and the accuracy of modes can be checked. Joints and payload are considered as mass with rotary inertia. These inertial parameters can be set to zero as well [Fig.4]

System input parameters for simulation ;

Joint 1 mass and rotary inertia about its center of mass ; $mj_1$ , $jj_1$ and similiarly for joint 2 ; $mj_2$, $jj_2$ , and for payload ; $m_p$ , $j_p$
For link 1 and 2 ; mass per unit length, link lengths, flexural rigidity constants .

$\rho A_1$ , $\rho A_2$ , $l_1$, $l_2$ , $EI_1$ , $EI_2$

Assumed mode shapes and gravity vector ;
$\phi_{11}(x)$ , $\phi_{12}(x)$ , $\phi_{21}(x)$ , $\phi_{22}(x)$ ; $g_x$ , $g_y$ , $g_z$
Initialization procedures

Time independent parameters are calculated at the initialization of the program only once per session. If mode shapes are up dated as function of changing boundary conditions, than these parameters need to be reevaluated. These parameters are ;

nm11, nm12, nm21, nm22, nw11, nw12, nw21, nw22,
nq11, nq12, nq21, nq22, kw11, kw12, kw21, kw22

$\phi_{11}(l_1)$ , $\phi_{12}(l_1)$ , $\phi_{21}(l_1)$, $\phi_{22}(l_2)$

$$\frac{\partial}{\partial x}(\phi_{11})\Big| , \quad \frac{\partial}{\partial x}(\phi_{12})\Big|_{x=l1} , \quad \frac{\partial}{\partial x}(\phi_{21})\Big| , \quad \frac{\partial}{\partial x}(\phi_{22})\Big|_{x=l2}$$

Here the objectives are as follows :
1. Verify that the model generated by the above algorithm is correct.
2. Analyze the effect of different mode shapes on the dynamic characteristics of the system.

1. Model verification will be done by comparing the response of flexible arm model with those of rigid arm, which has the same corrosponding parameters. Cleary as the flexural rigidity, EI(x), of the links increases, joint variable responses of flexible model should converge to those of rigid model response. Figures (6a) and (6c) clearly shows that joint variable responses converges to those of rigid arm case, as flexural rigidity, EI , of links is increased. Same test simulation is done with clamped-clamped mode shapes for the first link. Only one case result is given at figure (6d) since it is sufficient to illustrate the point here. For this case, when EI is set to 100 Nt $m^2$, the joint variable response were almost the same as rigid case. The reason for faster convergence for the clamped-clamped case than the clamped-free case is that clamped-clamped mode shapes result in a stiffer

system. However, clamped-free case is a more accurate prediction of the system response than the clamped-clamped case, as discussed below. In addition to that, as EI(x) increases the frequencies associated with structural flexiblity should increase, for the simple beam case natural frequencies are fuctions of EI as ;

$$w_i = (\gamma_i /1)^2 \ (EI/\rho A)^{1/2} \qquad (4.1)$$

Even though in two link arm case, we are considering here, (4.1) does not hold exactly, it is still valid in principle and gives a quantitative idea about what to expect. Given the fact that for these simulation conditions, nonlinear effects should not be very important, Rayleigh's energy principle also supports this expectation. Figures (7) - (8) confirm these expectations.

2. Modelling method cleary reveals that mode shapes are important parameters of the system dynamics (e.g. Eqn (3.12)). What assumed mode shapes should be used and would it make an important difference in the system characteristics? Theoretically, the only constraint on the assumed mode shapes is that they must satisfy the geometric boundary conditions, but not neccessarily the natural boundary conditions nor the governing differential equations. The governing differential equations and natural boundary conditions are results of the functional variation of the Hamiltonian and are approximately satisfied in any case. The controlled end of each link, driven by a high gain feedback controller, behaves more like a clamped end [1]. The other end condition of the intermediate links should be approximated by a mass with rotary inertia due to other links of the serial structure and payload. However, for different structures and even for diffrent payloads the resultant simple beam analysis will give different mode shapes. Given the fact that these are natural boundary conditions and will be approximately satisfied even if assumed mode shapes do not satisfy them, a clamped-free simple beam mode shape would be an appropriate choice for the assumed modes used in the model. For curiosity, model is also simulated for clamped-clamped mode shapes for the first link. Clamped-clamped case results in a stiffer system. As a result, joint variable response converges to rigid arm response faster than clamped-free case as function of flexural rigidity, frequency of flexible vibrations are higher than those of clamped-free case for the same parameters and conditions.

## V.Conclusion :

From the modeling technique point of view, it has been shown that Lagrangian - Assumed modes method can be effectively used for multi-link flexible arms. The availability of general purpose symbolic manipulation programs overcomes the algebraic complexity of derivation steps, and allows the researchers to obtain more complete models in very short time, in spite of their complexity. A new systematic algorithm based on Lagrangian-Assumed mode method is presented suitable for symbolic manipulation by digital computers. The algorithm is applied to a two link flexible arm. Simulation results are discussed and shown that the method worked very well for this example case.

## VI. References :


1. Hastings, G.G., Book, W.J., "Verification of a Linear Dynamic Model for Flexible Robotic Manipulators", IEEE Control Systems Magazine, IEEE Control Systems Society, to appear.
2. Alberts, T.A, Hastings, G.G., Book, W.J., Dickerson, S.L., "Experiments in Optimal Control of a Flexible arm with Passive Damping", VPI & SU/AIAA Symposium on Dynamics and Control of Large Flexible Structures, Blackburg, VA, June, 1985.
3. Cannon, R.H.Jr., Schmitz, E., "Initial Experiments on the End-Point Control of a Flexible One-Link Robot" , The International Journal of Robotics Research, Vol.3, No.3, Fall 1984, pp. 62-75.
4. Balas, M.J., "Active Control of Flexible Systems", Journal of Optimization Theory and Applications, Vol. 25, No. 3, July 1978, pp.415-436.
5. Midha, A., Erdman, A.G., Frorib, D.A., "Finite Element Approach to Mathematical Modelling of High-Speed Elastic Linkages", Mechanism and Machine Theory, Vol.13, 1978, pp. 603-618.
6. Yoo, W.S., Haug, E.J., "Dynamics of Flexible Mechanical Systems", Third Army Conference on Applied Mathematics and Computing, May 13-16, 1985, Atlanta, Georgia.
7. Sunada, W., Dubowsky, S., "The Application of Finite Element Methods to the Dynamic Analysis of Flexible Linkage Systems", Journal of Mechanical Design, Vol. 103, 1983, pp. 643-651.
8. Sunada, W., Dubowsky, S., "On the Dynamic Analysis and Behavior of Industrial Robotic Manipulators with Elastic Members", Transactions of ASME, J. Mech.,Trans., Automation and Design , Vol 105, 1983, pp. 42-51.
9. Shabana, A.A., Wehage, R.A., "A coordinate Reduction Technique for Dynamic Analysis of Spatial Substructures with Large Angular Rotations", J.Struct. Mech., 1983, pp. 401-431.
10. Usoro, P.B., Nadira, R.,Mahil, S.S., "Advanced COntrol of Flexible Manipulators", Phase I Final report, NSF Award Number ECS-8260419, April 1983.
11. Book, W.J., Maizza-Netto, O., Whitney , D.E., "Feedback Control of Two Beam, Two Joint Systems with Distributed Flexibility", ASME Journal of Dynamic Systems, Measurement, and Control 97G, Dec. 1975 .
12. Book, W.J., Majette, M., "Controller Design for Flexible, Distributed Parameter Mechanical Arms Via Combined State Space and Frequency Domain Techniques", Journal of Dynamic Systems, Measurement, and Control, Vol.105, Dec. 1985, pp.245-254.
13. Book, W.J., " Recursive Lagrangian Dynamics of Flexible Manipulator Arms ", International Journal of Robotic Research, Vol. 3, No.3, Fall 1984, pp.87-101.
14. SMP Reference Manual, Inference Corporation , 1985.

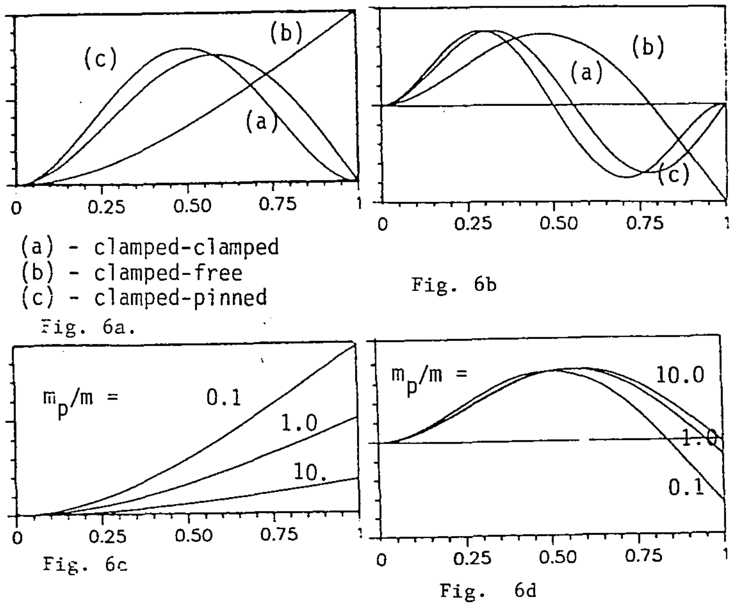


Fig. 6.a,b,c,d - First and Second mode shapes

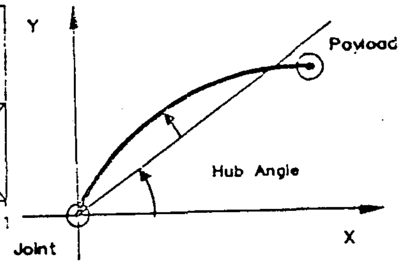


Fig. 3 One Link Arm

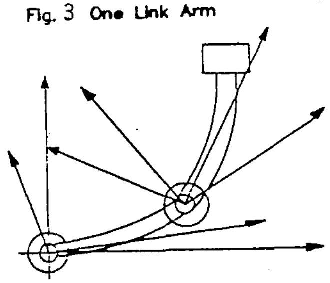

Fig. 4 Two Link Arm Example

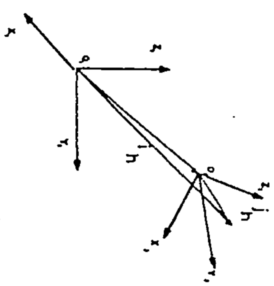

Fig. 2 Transformation Matrices

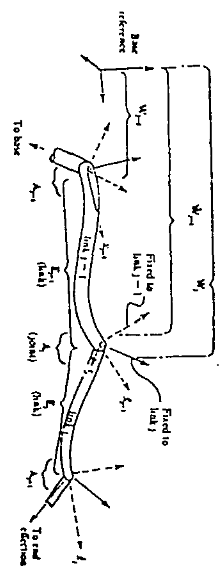

Fig. 1 A Flexible serial structure

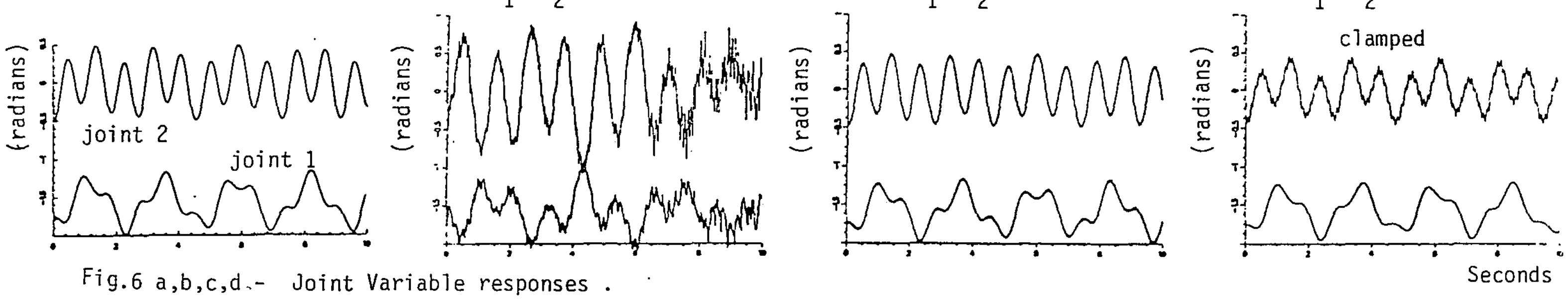


Fig.6 a,b,c,d.- Joint Variable responses .

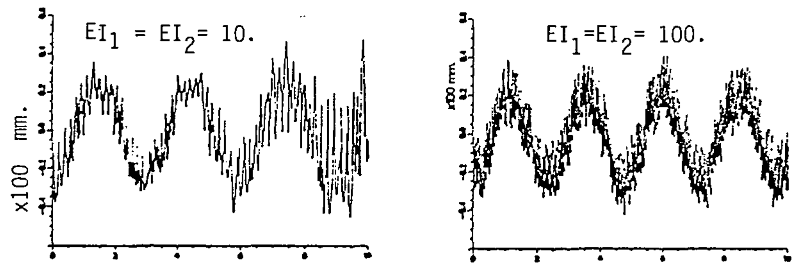


Fig.7 a,b - Link 1, generalized flexible coordinate 1

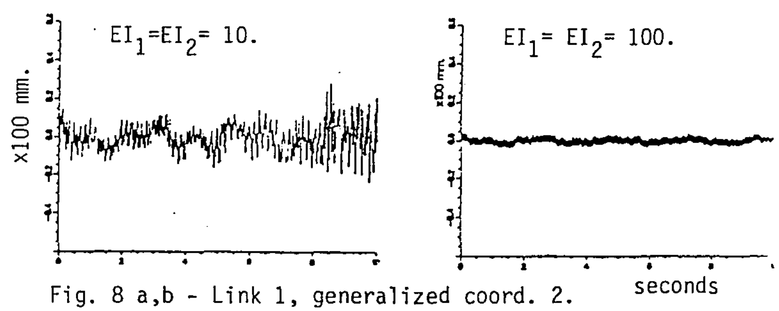


Fig. 8 a,b - Link 1, generalized coord. 2.