# A Finite-Element Approach to Control Link Mechanisms: Its Concept and Basic Simulation

D. Isobe
Institute of Engineering Mechanics and Systems
University of Tsukuba, Ibaraki, Japan

H. Takeuchi
Mechanical Engineering Laboratory, Agency of Industrial Science and Technology
Ministry of International Trade and Industry, Ibaraki, Japan

T. Ueda
Graduate School, University of Tsukuba

## Summary

In this study, link mechanisms are modeled using finite elements based upon the Shifted Integration technique. A single link structure of a pin joint and a rigid bar is expressed by two linear Timoshenko beam elements with their numerical integration points shifted to opposite ends of the link joint. The finite-element approach in control schemes may provide flexibility against sudden change in a hardware system, since the remodeling can be achieved easily by rewriting input data. This paper describes an application of the modeling to normal pin joint-rigid bar link mechanisms. A numerical scheme for obtaining joint torque curves in n-link mechanisms based upon Cartesian coordinates is derived, and a numerical test on an 8-link mechanism by the Finite Element Method is carried out. The obtained joint torque curves as well as the CPU time are compared to those obtained by the conventional Newton-Euler method.

## Introduction

The parallel control of robotic systems, which is an adaptive control scheme involving the reassembly of hardware systems into different configurations while still using the same software, has been progressively developed in recent years. The concept of robotic systems with high adaptability to their environment is called Parallel Robotics, and is expected to be applied in space, under the sea, in mining, and in construction, where environments tend to change rapidly. Hamlin and Sanderson [1], for example, designed a prototype robotic architecture, which is capable of reassembling into many different patterns of platforms consisting of linear actuators and multi-link spherical joints. However, software systems which can practically handle the change in the higher redundant architecture have not been established yet, since difficulties lies in obtaining the torque of each joint of such architecture by conventional schemes for solving the dynamics.

A conventional control system necessitates the change of dynamic equations depending on the shape of a system or the quantity of the linked members. On the other hand, the Finite Element Method (FEM), a widely used computational tool for analyzing structures, fluids, etc., is capable of expressing the behaviors of each discrete element, as well as the whole continuous system, by evaluating the dynamic equations derived from the energy principle. Isobe and Nakagawa [2] applied the FEM to a control system of connected piezoelectric actuators, and achieved good control not only of the actuator itself but also of the entire system. The FEM does not require reimplementation of dynamic equations in the software, and flexible control can be achieved simply by changing the input data in the case of a physical change in the hardware system. For simplicity and flexibility, this method may provide a suitable control system, particularly for higher redundant link mechanisms.

In this paper, a new type of control scheme for link mechanisms has been developed. Link mechanisms are modeled using linear Timoshenko beam elements based on the Shifted Integration (SI) technique [3], which was originally used in finite element analyses of framed structures. A numerical scheme for obtaining the joint torque in Cartesian coordinates is derived, and a numerical example for a normal pin joint-rigid bar link mechanism is shown.

## Finite-element modeling of a link mechanism

The Shifted Integration technique, which is applied in order to model link mechanisms in this paper, was originally developed as a finite element scheme for the analysis of framed structures. By considering the equivalence conditions between the strain energy approximations of a linear Timoshenko beam element and a physical model, the rigid-bodies spring model (RBSM), the relationship between the locations of a numerical integration point ($s_1$) and a plastic hinge ($r_1$) in the linear Timoshenko beam element $(-1 \leq r_1, s_1 \leq 1)$ is obtained [3]. Referring to Fig.1, it is expressed by the following equation:

$$s_1 = -r_1 \ or \ r_1 = -s_1 \tag{1}$$

where $s_1$ and $r_1$ are the positions of the numerical integration point in the finite element and the spring in the RBSM, respectively. Referring to the equation above, the rotational and shear spring placed at the left end ($r_1$= -1) of an element can be expressed by shifting a numerical integration point in the element to the right end ($s_1$=1). Various stiffness values of a link joint are then expressed by changing the stiffness of the spring (or the element). Figure 2 shows the general concept of modeling by the SI technique. As shown in the figure, a link mechanism formed by a motor and a link member can be modeled by placing a nodal point at the center of gravity, and by two Timoshenko beam elements with numerical integration points shifted to the opposite ends of the link joint. The elemental stiffness matrix is obtained using

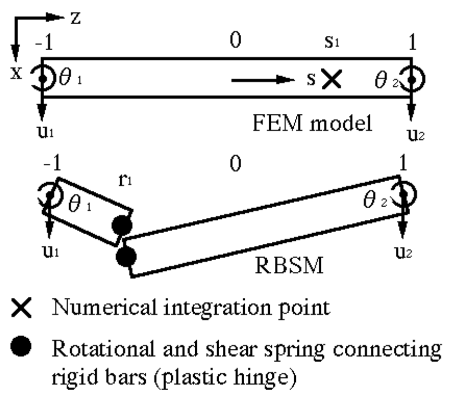


Fig.1 Linear Timoshenko beam element and its physical equivalent

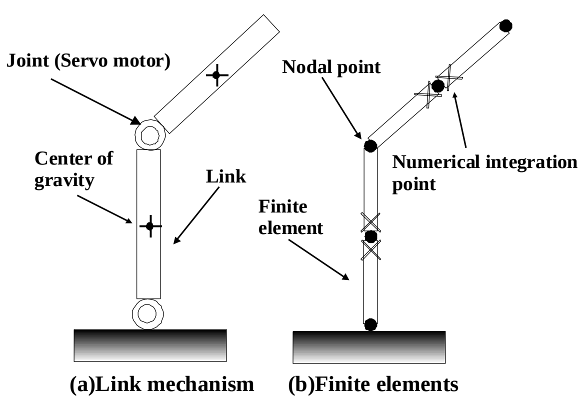


Fig.2 Modeling of link mechanism by Shifted Integration technique

$s_1$, $r_1$ and the normalized stiffness $C_{mot}$ of the spring, as shown below:

$$[K] = C_{mot} \int_V [B(s_1)]^T [D(r_1)][B(s_1)] dV \quad (2)$$

Various types of link joints (pin to rigid) can be expressed by varying $C_{mot}$ between 0 and 1. The value 0 is used in this paper to estimate the validity of the proposed scheme in controlling the normal pin joint-rigid bar link mechanisms. A lumped mass matrix is also defined using the location of the numerical integration point $s_1$. The diagonal components of the mass matrix are:

$$[M] = [m_1 \; m_1 \; m_1 \; \frac{m_1 l^2}{12} \; \frac{m_1 l^2}{12} \; t_1 \; m_2 \; m_2 \; m_2 \; \frac{m_2 l^2}{12} \; \frac{m_2 l^2}{12} \; t_2] \quad (3)$$

where

$$\begin{aligned} m_1 &= \rho A l(1 - s_1)/2, \quad m_2 = \rho A l(1 + s_1)/2 \\ t_1 &= \rho I_z l(1 - s_1)/2, \quad t_2 = \rho I_z l(1 + s_1)/2 \end{aligned} \quad (4)$$

and $\rho$, $A$, $l$, and $I_z$ are the density of the member, the cross-sectional area, the length of the element and the polar moment of area inertia, respectively. According to the matrix, the total mass of the element assembles at $r_1$=1 when the link joint is placed at $r_1$= -1 (thus $s_1$=1), and vice versa. A nodal point placed between two Timoshenko beam elements thus expresses the center of gravity in a link member (see Fig.2).

## Calculation of joint torque in n-link mechanism

Figure 3 shows the nodal forces (based on global coordinates) acting on the *i*-th link in an n-link mechanism, and the relationship between link numbers and nodal numbers. The joint torque $\tau_i$ required on the *i*-th link is determined by adding *i*+1-th joint torque $\tau_{i+1}$ to the sum of inertia moments acting in this link, and is expressed by nodal forces $F_{iCx}$ and $F_{iC\phi}$ based on elemental (or link) coordinates as follows:

$$\tau_i = l_{iC} F_{iCx} + l_i (\sum_{j=i+1}^{n} F_{jC})_x + F_{iC\phi} + \tau_{i+1} \quad (5)$$

where $l_{iC}$ is the length between the former joint and the center of gravity and $l_i$ is the link length, respectively. By arranging Eq. (5) into global coordinates and in matrix form, the joint torque vector is expressed as:

$$\{\tau^n\} = [L^n][T^n]\{P^n\} \quad (6)$$

where $\{P^n\}$ is a vector related to nodal force, defined as follows:

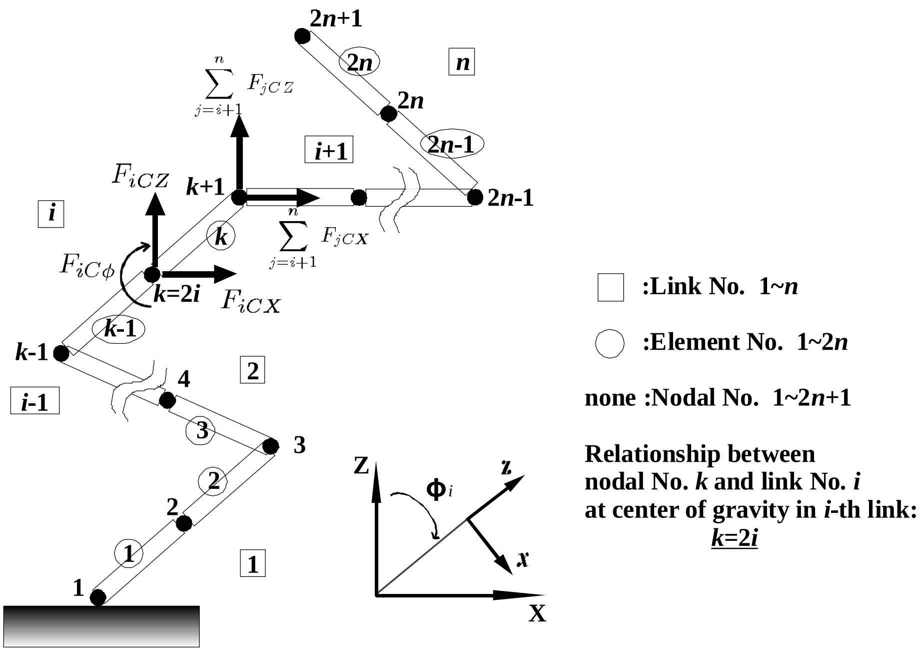


Fig.3 Nodal force acting on $i$-th link in n-link mechanism

$$\{P^n\} = \begin{Bmatrix} P_1 \\ P_2 \\ \cdot \\ \cdot \\ P_n \end{Bmatrix}, \quad where \quad \{P_i\} = \begin{Bmatrix} F_{iCX} \\ F_{iCZ} \\ \sum_{j=i+1}^{n} F_{jCX} \\ \sum_{j=i+1}^{n} F_{jCZ} \\ F_{iC\phi} \end{Bmatrix}, \quad (i = 1, \cdots, n) \tag{7}$$

Using the rotational angle $\phi$ between global and elemental (or link) coordinates, the transformation matrix $[T^n]$ is expressed as:

$$\left[\, T^n \,\right] = \begin{bmatrix} T_1 & & & & & & 0 \\ & T_2 & & & & & \\ & & T_3 & & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & 0 & & & & \cdot & \\ & & & & & & T_n \end{bmatrix}, \quad where \quad \left[\, T_i \,\right] = \begin{bmatrix} cos\phi_i & -sin\phi_i & 0 & 0 & 0 \\ sin\phi_i & cos\phi_i & 0 & 0 & 0 \\ 0 & 0 & cos\phi_i & -sin\phi_i & 0 \\ 0 & 0 & sin\phi_i & cos\phi_i & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

Conventional schemes require reiteration of the transformation between each link to calculate every joint torque in the link mechanism. However, the proposed scheme using FEM requires only one transformation between the global and elemental coordinates. This leads to a reduction of calculation time, particularly when the link number is not nominal. $[L^n]$ is a matrix related to member length, which is expressed as:

$$
\left[\, L^n \,\right] = \begin{bmatrix} L_1 & L_2 & L_3 & \cdot & \cdot & \cdot & L_n \\ & L_2 & L_3 & \cdot & \cdot & \cdot & L_n \\ & & L_3 & \cdot & \cdot & \cdot & L_n \\ & & & \cdot & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot & \cdot \\ & 0 & & & & \cdot & \cdot \\ & & & & & & L_n \end{bmatrix}, \quad where \quad \left[\, L_i \,\right] = \left[\, l_{iC} \quad 0 \quad l_i \quad 0 \quad 1 \,\right] \tag{9}
$$

Information on $i$+1~$n$ link is summed by multiplying the above matrix with vector $[T^n]\{P^n\}$, which is the nodal force vector transformed into elemental coordinates. On the other hand, a vector related to incremental nodal force acting on the $i$-th link is defined by using the nodal numbers:

$$
\{\Delta p_k\} = \left\{ \begin{array}{c} \Delta f_{kX} \\ \Delta f_{kZ} \\ \sum_{h=k+1}^{2n+1} \Delta f_{hX} \\ \sum_{h=k+1}^{2n+1} \Delta f_{hZ} \\ \Delta f_{k\phi} \end{array} \right\}, \quad (k = 2i, \quad i = 1, \cdots, n) \tag{10}
$$

Thus the vector related to nodal force acting on the $i$-th link at t+$\Delta$t is successively calculated by using the above vector:

$$
\{P_i\}_{t+\Delta t} = \{P_i\}_t + \{\Delta p_k\}, \quad (k = 2i) \tag{11}
$$

The successive values of n-link joint torque are then obtained by substituting Eq.(11) into Eq.(6). Newmark's $\beta$ method ($\delta$=1/2) is used for the time integration scheme to solve the incremental kinematic equation.

## Numerical example

The proposed scheme using FEM is applied to the joint torque calculation of an 8-link mechanism as an example, and the obtained torque curves as well as the CPU time is compared with those obtained by the conventional Newton-Euler method. Mathematica 4.0 is used not only for conducting dynamic equations but also for calculating the joint torque when applying the conventional scheme to this particular problem, since the implementation of the massive equations in other software programs is practically impossible. Microsoft's Fortran PowerStation 4.0 is used to compile the FEM program as well as the program for the Newton-Euler method in cases with fewer links. All calculations are carried out on a Dell Dimension XPS T600 (CPU: Intel Pentium III 600 MHz, RAM: 383MB) PC system.

Figures 4(a) and 4(b) show the torque curves obtained by both schemes when the target trajectory for 1.0 s motion is given to the 8-link mechanism (each link length: 20 cm, weight: 107.5 g, center of gravity at mid-point) as shown in the figure. Gravity is assumed to act vertically downward. Although the motion may produce various nonlinear forces such as the Coriolis force, the torque curves obtained by the FEM are in good agreement with those obtained by the Newton-Euler method. Evidently, the proposed scheme is capable of considering every component in the dynamics.

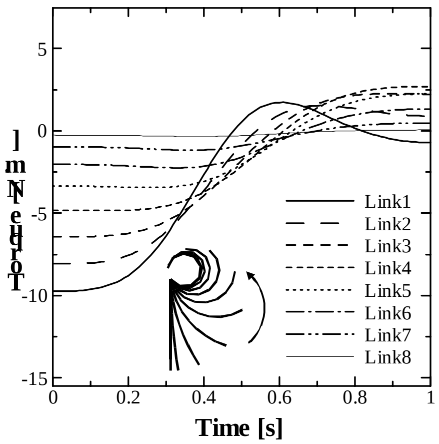


Fig.4 (a) Joint torque curves obtained by Newton-Euler method

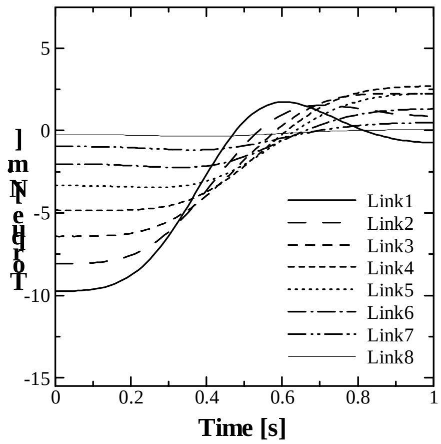


Fig.4 (b) Joint torque curves obtained by FEM

Comparisons between genuine CPU times for 1.0 s motion and time possession in the total process for a 5-link mechanism in both schemes (both compiled in Fortran) are shown in Figs. 5(a) and 5(b), respectively. Although the conventional scheme has a small advantage over the proposed scheme in terms of CPU time, its total process time, which includes the construction of dynamic equations and the implementation into another software program, is much longer than that of the proposed scheme. Conversely, the proposed scheme has no need to change the description in its program, and users are encouraged to change the system version by simply rewriting the input data of the numerical model. Therefore, the flexibility of the control system is increased and the total process time can be significantly reduced.

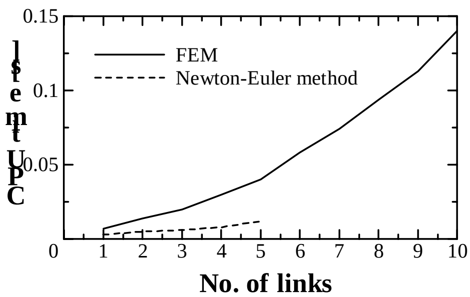


Fig.5 (a) CPU time against No. of links

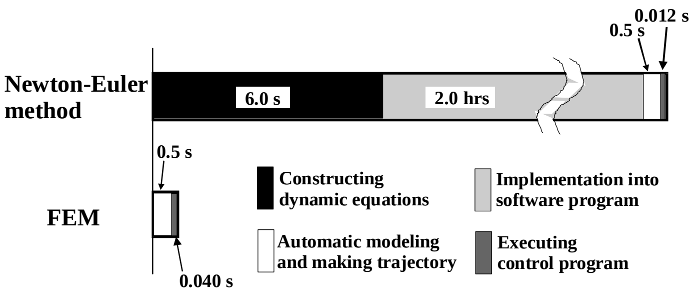


Fig.5 (b) Time possession in total process (5-link mechanism)

## Concluding Remarks

The approach to the control of link mechanisms using FEM may achieve real-time control of hyper-redundant manipulators that consist of 10, 20, or even more link members. It may also help in controlling the behaviors of flexible manipulators where the stiffness of the link member is an important factor.

## References


1. Hamlin, G. J. and Sanderson, A. C. (1997): “TETROBOT: A Modular Approach to Parallel Robotics”, *IEEE Robotics and Automation Magazine*, Vol. 4, No. 1, pp. 42-49.
2. Isobe, D. and Nakagawa, H. (1999): “A Parallel Control System for Continuous Architecture Using Finite Element Method, *Journal of Intelligent Material Systems and Structures,* Vol. 9, No. 12, pp.1038-1045.
3. Toi, Y. (1991): "Shifted Integration Technique in One-Dimensional Plastic Collapse Analysis Using Linear and Cubic Finite Elements", *International Journal for Numerical Methods in Engineering*, Vol. 31, pp.1537-1552.