Inverse Dynamics Calculation of Underactuated Link Systems Using Parallel Solution Scheme

Daigoro Isobe[1], Kouji Yamanaka[2] and Yuto Kitamura[3]
[1] Dept. of Engineering Mechanics and Energy, University of Tsukuba, Japan
isobe@kz.tsukuba.ac.jp
[2] Kyushu Railway Company, Japan
[3] Graduate School, University of Tsukuba, Japan
e0411288@edu.esys.tsukuba.ac.jp

Abstract- **A general-purpose solution scheme of inverse dynamics for link systems was developed on the basis of a finite element approach. It can not only deal with different types of configurations, such as open-loop, closed-loop, or multibranch link systems, but also with the elasticity of constituted links without the need to revise any part of the scheme. The main objective of this study is to extend the use of the scheme by applying the inverse dynamics calculations to several types of underactuated link systems. A solution scheme of kinematics is also developed on the basis of the finite element method in order to calculate target trajectories that compensate for the inertial forces acting in the systems. The obtained trajectories are fed into the inverse dynamics calculation using the parallel solution scheme. The torque values are verified through comparison with the input moment used in finite element analyses, and the validity of the parallel solution scheme is confirmed.**

I. INTRODUCTION

Dynamic equations used for feed-forward control of robotic mechanisms include interdependent variables between the constituting links, since they are normally evaluated in relative polar coordinates and in the dimension of torque. Accordingly, it will become highly complicated to derive inverse dynamics of closed-loop link systems, continuously transforming systems, or flexible link systems. Nakamura and Ghodoussi proposed a systematic computational scheme of the inverse dynamics specified for closed-loop link systems, derived using d'Alembert's principle [1]. Sugimoto derived an equation of motion for closed-loop link systems by motor algebra [2]. Nakamura and Yamane developed a computational algorithm for the inverse and forward dynamics of open and closed kinematic chains, which can be applied seamlessly to the motion of any rigid link system without switching algorithms [3]. However, these methods are specific for rigid-body link systems, and users occasionally must revise the methods or dynamic equations when the forms of the system, elasticity of the links, or characteristics of the joints (active or passive) are changed.

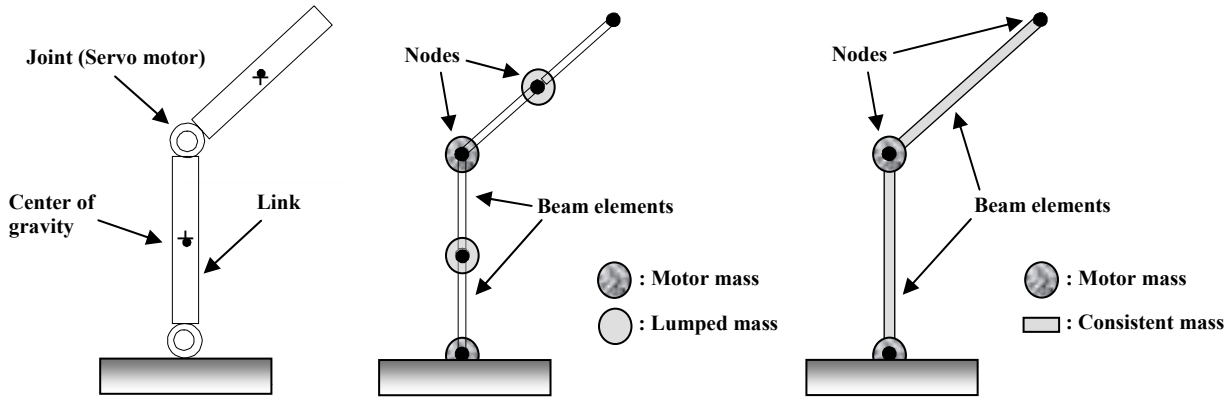
Isobe, on the other hand, developed a completely new solution scheme for inverse dynamics, called the parallel solution scheme [4]. The scheme was developed using a finite element approach, handling the entire system as a continuum. By taking advantage of the natural characteristics of the finite element method (FEM), i.e., the capability of expressing the behavior of each discrete element as well as that of the entire continuous system, local information, such as nodal forces and displacements, can be calculated in parallel. The analyzed model is evaluated in absolute Cartesian coordinates with the equation of motion expressed in the dimension of force. The inverse dynamics is calculated using a matrix-form relation to the nodal forces obtained by the finite element calculation. The matrix-form equations are divided individually into terms of force, transformation between coordinates, and length, which makes the scheme potentially higher in applicability and expansibility. The scheme can not only deal with different types of configurations such as open-loop, closed-loop, or multibranch link systems, but also with the elasticity of constituent links without the need to revise any part of the scheme.

There are two different versions of the parallel solution scheme for calculating the inverse dynamics of link systems. One version uses two beam element subdivisions per link to explicitly express the center of gravity of links, and the other uses only one beam element per link with consistent mass distribution. The former version is easier to compare with conventional dynamic equations since it explicitly expresses the center of gravity. We implemented the linear Timoshenko beam elements in modeling with the former version [4]. The latter, on the other hand, has the merits of high accuracy, short calculation time and stability, particularly for flexible models [5]. Therefore, the latter version with Bernoulli-Euler beam elements for the modeling, is described and used in this paper. It is also extended to underactuated link systems [6,7] by developing a kinematics solution scheme for the systems, which is a revised version of the one used for flexible models. Some numerical examples of underactuated systems are shown, and the results are verified by comparison with the results obtained by finite element analyses.

II. PARALLEL SOLUTION SCHEME OF INVERSE DYNAMICS

In the parallel solution scheme, a link system consisting of motor joints and links, as shown in Fig. 1(a), is modeled and subdivided using finite elements. One method of modeling the system is to subdivide a link into two beam elements, as shown in Fig. 1(b), with an intermediate node explicitly

978-1-4244-1766-7/08/$25.00 ©2008 IEEE



(a) Link system (b) Lumped mass distribution (c) Consistent mass distribution

Fig. 1. Modeling of link systems using finite elements.

expressing the center of gravity [4]. Another method is to substitute a link with a single beam element, as shown in Fig. 1(c), assuming a consistent mass distribution along the link [5]. The type of modeling shown in Fig. 1(c) is described and used throughout this paper.

The order of the displacement function used for a Bernoulli-Euler beam element is higher than that of a linear Timoshenko beam element, and thus, the former can express deformation more accurately with fewer elements. The consistent mass matrix of the beam element is formulated in the same manner as the displacement function and does not require an expression for the center of gravity. Note that, since the deformation of the element is defined using a higher-order function, it requires only one-element subdivision per member for cases of infinitesimal deformation.

Figure 2 shows the nodal forces (based on global coordinates) acting on the *i*th link in a three-dimensional open-loop *n*-link system with a consistent mass distribution. The joint torque τ_{ix} required around the x-elemental axis on the *i*th link is determined by adding the *i+1*th joint torque

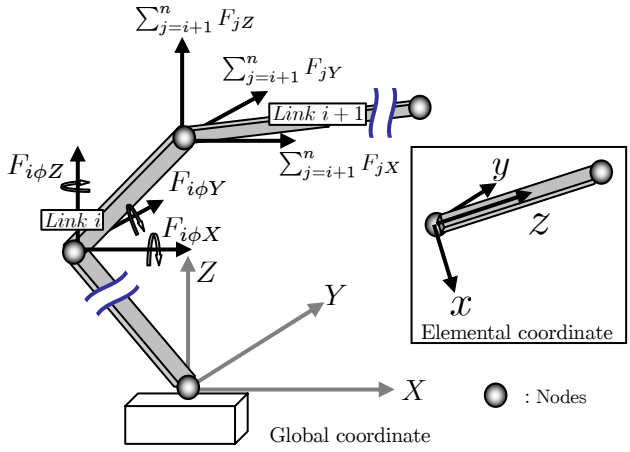


Fig. 2. Nodal forces acting in a link system with consistent mass distribution.

$\tau_{(i+1)x}$ to the sum of the moments of inertia acting on this link, and is expressed by the nodal forces F_{iy} and $F_{i\varphi x}$ based on elemental (or link) coordinates as

$$\tau_{ix} = l_i(\sum_{j=i+1}^{n} F_j)_y + F_{i\phi x} + \tau_{(i+1)x}. \quad (1)$$

By considering the other components around the y- and z-axes and arranging them into global coordinates (X, Y, Z) in matrix form, the joint torque vector is expressed as

$$\{\tau^n\} = [L^n][T^n]\{P^n\}, \quad (2)$$

where $\{P^n\}$ is a *6n×1* vector related to nodal force and is defined as

$$\{P^n\} = \begin{Bmatrix} P_1 \\ P_2 \\ \cdot \\ \cdot \\ P_n \end{Bmatrix}, \quad where \quad \{P_i\} = \begin{Bmatrix} \sum_{j=i+1}^{n} F_{jX} \\ \sum_{j=i+1}^{n} F_{jY} \\ \sum_{j=i+1}^{n} F_{jZ} \\ F_{i\phi X} \\ F_{i\phi Y} \\ F_{i\phi Z} \end{Bmatrix}, (i = 1, \cdots, n) \quad (3)$$

$[T^n]$ is a *6n×6n* transformation matrix and is defined as

$$[T^n] = [h^n][T^n_{GE}], \quad (4)$$

where $[h^n]$ is a correction matrix between the *x-y* and *z-x* coordinate systems, which simply inverts the signs of the components in the *y*-axis direction. $[T^n_{GE}]$ is a transformation matrix between the global and elemental coordinates and is expressed as

$$\left[\ T_{GE}^{n}\ \right]=\begin{bmatrix} T_1 & & & & & \\ & T_2 & & & 0 & \\ & & T_3 & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & 0 & & & \cdot & \\ & & & & & T_n \end{bmatrix}, \tag{5}$$

where

$$\left[\ T_i\ \right]=\begin{bmatrix} A_i & 0 \\ 0 & A_i \end{bmatrix} \tag{6a}$$

and

$$\left[\ A_i\ \right]=\begin{bmatrix} cos\phi_{iXx} & cos\phi_{iYx} & cos\phi_{iZx} \\ cos\phi_{iXy} & cos\phi_{iYy} & cos\phi_{iZy} \\ cos\phi_{iXz} & cos\phi_{iYz} & cos\phi_{iZz} \end{bmatrix}, \tag{6b}$$

where φ_{iXx}, for example, represents the rotational angle between the X (global) and x (elemental) coordinates.

$[L^n]$ is a $3n\times6n$ matrix related to link length and is expressed as

$$[L^n]=[T_\Lambda^n][\Lambda^n], \tag{7}$$

where $[T^n_\Lambda]$ is a transformation matrix between each elemental coordinate, and $[\Lambda^n]$ is a matrix expressed as

$$\left[\ \Lambda^n\ \right]=\begin{bmatrix} \Lambda_1 & & & & & \\ & \Lambda_2 & & & 0 & \\ & & \Lambda_3 & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & 0 & & & \cdot & \\ & & & & & \Lambda_n \end{bmatrix}, \tag{8}$$

where

$$\left[\ \Lambda_i\ \right]=\begin{bmatrix} 0 & l_i & 0 & 1 & 0 & 0 \\ l_i & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{9}$$

III. Kinematics Calculation for Underactuated Models

To calculate the inverse dynamics for underactuated link systems, information on target trajectories that compensate for the inertial forces acting at the links, as well as on the stiffnesses (or flexibilities) of the links and nonactuated joints are required. Therefore, a solution scheme of kinematics, that was developed on the basis of the FEM for flexible models [5], is revised and applied in this study.

When inertia caused by the overall system motion is taken into account, the equation of motion at time $t+\Delta t$ is derived from the principle of virtual work as

$$[M]\{\ddot{u}_m\}_{t+\Delta t}+[M]\{\ddot{u}_d\}_{t+\Delta t}+[C]\{\dot{u}_d\}_{t+\Delta t}+[K]\{\Delta u_d\}=\{F\}_{t+\Delta t}-\{R\}_t, \tag{10}$$

where $[M]$ is the total mass matrix, $[C]$ is the total damping matrix, $[K]$ is the total stiffness matrix, $\{F\}$ is the external force vector, and $\{R\}$ is the internal force vector. The vectors $\{\ddot{u}_m\}$ and $\{\ddot{u}_d\}$ denote the acceleration vectors for the overall system motion components and material deformation components, respectively, $\{\dot{u}_d\}$ denotes the velocity vector for material deformation components, and $\{\Delta u_d\}$ is the incremental material deformation vector at time $t+\Delta t$. By applying Newmark's β method (δ, β: integration parameters) as a time integration scheme, the velocity and acceleration vectors are calculated as

$$\{\dot{u}_m\}_{t+\Delta t}=\frac{\delta}{\beta\Delta t}\{\Delta u_m\}-(\frac{\delta}{\beta}-1)\{\dot{u}_m\}_t-(\frac{\delta}{2\beta}-1)\{\ddot{u}_m\}_t\Delta t, \tag{11a}$$

$$\{\dot{u}_d\}_{t+\Delta t}=\frac{\delta}{\beta\Delta t}\{\Delta u_d\}-(\frac{\delta}{\beta}-1)\{\dot{u}_d\}_t-(\frac{\delta}{2\beta}-1)\{\ddot{u}_d\}_t\Delta t, \tag{11b}$$

$$\{\ddot{u}_m\}_{t+\Delta t}=\frac{1}{\beta\Delta t^2}\{\Delta u_m\}-\frac{1}{\beta\Delta t}\{\dot{u}_m\}_t-(\frac{1}{2\beta}-1)\{\ddot{u}_m\}_t, \tag{11c}$$

$$\{\ddot{u}_d\}_{t+\Delta t}=\frac{1}{\beta\Delta t^2}\{\Delta u_d\}-\frac{1}{\beta\Delta t}\{\dot{u}_d\}_t-(\frac{1}{2\beta}-1)\{\ddot{u}_d\}_t, \tag{11d}$$

where $\{\dot{u}_m\}$ is the velocity vector for the overall system motion components and $\{\Delta u_m\}$ is the incremental motion vector at time $t+\Delta t$. Substituting (11) into (10) yields

$$\begin{aligned}&([K]+\frac{1}{\beta\Delta t^2}[M]+\frac{\delta}{\beta\Delta t}[C])\{\Delta u_d\}=\\&\{F\}_{t+\Delta t}-\{R\}_t+[M](\frac{1}{\beta\Delta t}\{\dot{u}_d\}_t+(\frac{1}{2\beta}-1)\{\ddot{u}_d\}_t)\\&-[M](\frac{1}{\beta\Delta t^2}\{\Delta u_m\}-\frac{1}{\beta\Delta t}\{\dot{u}_m\}_t-(\frac{1}{2\beta}-1)\{\ddot{u}_m\}_t)\\&+[C]((\frac{\delta}{\beta}-1)\{\dot{u}_d\}_t+(\frac{\delta}{2\beta}-1)\{\ddot{u}_d\}_t\Delta t)\end{aligned}. \tag{12}$$

By applying $\{\Delta u_m\}$ as the input trajectory (normally an initial trajectory for a rigid model) in a time integration loop of (12) and using the vectors of (11) at time t, $\{\Delta u_d\}$ at each time step can be successively obtained. The displacement vector $\{u_m\}$ for the overall system motion components and the displacement vector $\{u_d\}$ for the material deformation components are calculated incrementally as

$$\{u_m\}_{t+\Delta t}=\{u_m\}_t+\{\Delta u_m\}, \tag{13a}$$

$$\{u_d\}_{t+\Delta t} = \{u_d\}_t + \{\Delta u_d\}. \quad (13b)$$

The total displacement $\{u\}$ is obtained by summing both the overall system motion and material deformation components as

$$\{u\}_{t+\Delta t} = \{u_m\}_{t+\Delta t} + \{u_d\}_{t+\Delta t}. \quad (14)$$

The final target trajectory considering the effects of stiffness and damping is obtained using (14).

The calculation process described above can be commonly used for flexible and underactuated models. However, some special treatments should be carried out for the latter model. First, we use an elemental stiffness matrix with only axial components for nonactuated links, in order to exclude all the transmission of bending moments at the nonactuated joints. Second, the input trajectory must be successively revised during the calculation process to avoid any accumulation of numerical errors due to large deformation.

The trajectory revision algorithm developed in this study is explained using a simple example of a two-joint underactuated link system. Let the initial trajectory without considering any effect be a horizontal 2-second motion, as shown in Fig. 3. J1 is an active joint, whereas J2 is a nonactuated joint. The parameters of the links are as follows: material of the links SUS430; flexural stiffness 20.84 Nm^2; link lengths 0.25 m; link mass 9.75 $\times 10^{-2}$ kg; surplus mass at the tip 0.1 kg. If only the bending moment effect is excluded from the nonactuated link, and no revision of the input trajectory is performed, the final trajectory will become as shown in Fig. 4. It is clear that the coordinates are not successfully transformed during the motion, since some lengthening of links can be observed in the figure. In order to solve this problem, the concept of the successive revision of

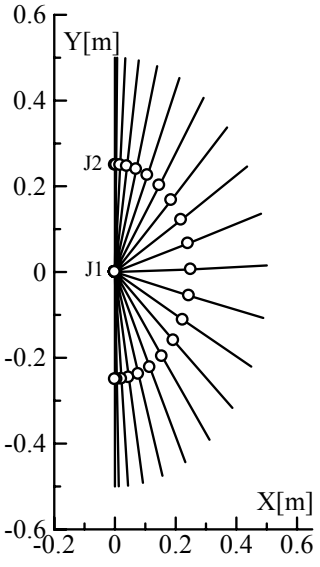


Fig. 3. Initial trajectory.

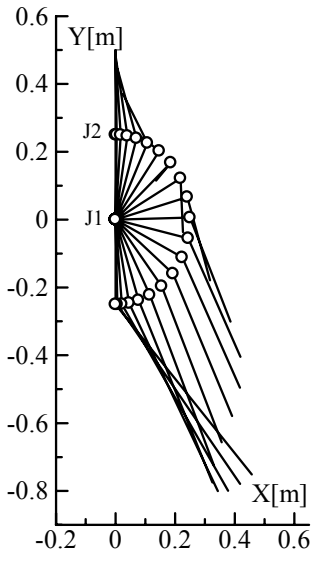


Fig. 4. Final trajectory without performing successive revision.

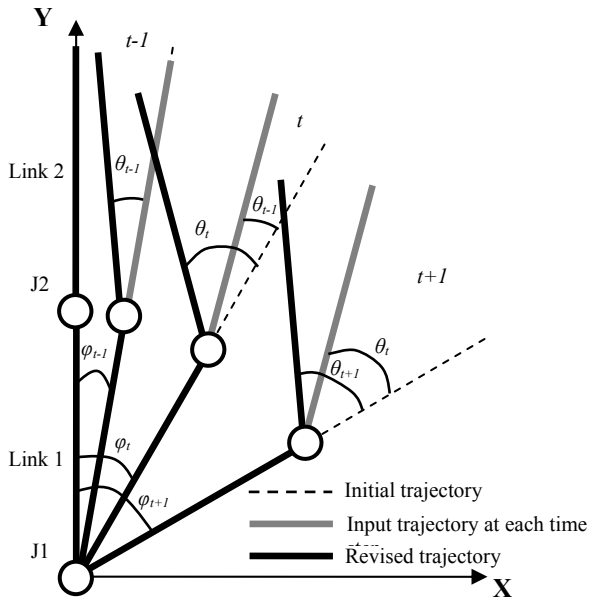


Fig. 5. Concept of successive trajectory revision algorithm for underactuated link systems.

input trajectories, as shown in Fig. 5, is developed. The broken lines in the figure are the initial trajectory, whereas the gray lines are the input trajectory at each time step and dark lines are the revised trajectories at the end of each time step. φ is the rotational angle of the active joint (J1, in this case), which will not be revised by the algorithm, and θ is the rotational angle of the nonactuated joint (J2, in this case), which will be calculated using the deformation obtained from (13b). Suppose the posture at time step *t-1* is represented by the dark line shown in the figure, with the rotational angle of J1 being φ_{t-1}. The input trajectory at the next time step *t* is calculated only by rotating the active joint to φ_t, whereas the overall posture is fixed as it is. The deformations of the link system at time step *t* are calculated starting from this posture (gray line at time step *t*). The calculated result at the end of time step *t* will become as the dark line (with the rotational angle of J1 being φ_t), and the revisions proceed successively to the next time step *t+1* in the same manner. In general *n*-link cases, the input trajectory can be revised by applying the above algorithm from the base joint to the end joint successively at each time step.

IV. Numerical Examples

Some numerical examples on underactuated link systems are shown in this section. Figure 6(a) shows the calculated trajectory of a two-joint underactuated model using the algorithm described in the previous section. The initial trajectory is given as shown in Fig. 3, and the same link parameters as those in the previous section are used. The integration parameters for Newmark's method are given as δ=1/2 and β=1/4 in this example. The time increment for the calculation is set to 10 ms. Note that the trajectory in Fig. 6(a) is improved compared with that in Fig. 4. Figure 6(b) shows the torque curves calculated using data in Fig. 6(a) as input

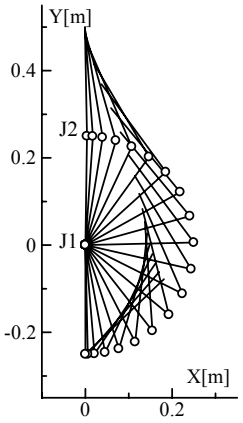


Fig. 6(a). Calculated trajectory for a two-joint underactuated link system considering the stiffness of Link 1.

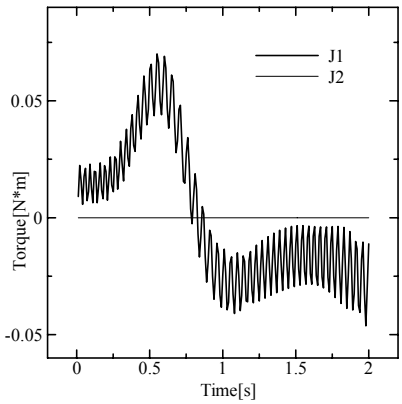


Fig. 6(b). Torque curves for a two-joint underactuated link system considering the stiffness of Link 1.

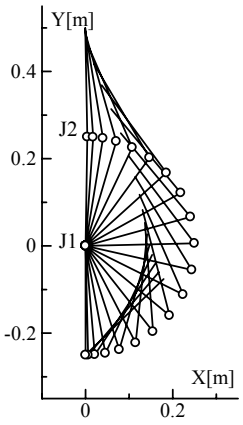


Fig. 7(a). Calculated trajectory for a two-joint underactuated link system without considering the stiffness of Link 1.

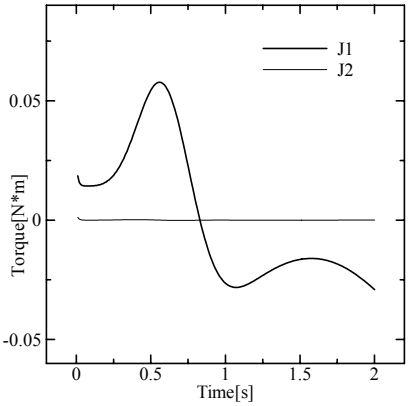


Fig. 7(b). Torque curves for a two-joint underactuated link system without considering the stiffness of Link 1.

for the parallel solution scheme described in Section III. The torque curve for J2 retains the value of 0 throughout the motion, which is proof that the joint is successfully expressed as a nonactuated joint. The vibration seen in the torque curve for J1 is proof that Link 1 has stiffness (or flexibility). It vibrates with a natural period in the range of 0.025 s ~ 0.04 s (note that the natural period changes with the posture), which matches the theoretical value for a two-link free vibration beam, 0.031 s. The vibration, however, generally exerts an adverse effect on the stability in actual control. Therefore, the property of Link 1 is idealized to that of a rigid link by neglecting the velocity and the acceleration of material deformation components in (12), $\{\dot{u}_d\}$ and $\{\ddot{u}_d\}$, and the effect is verified. Figure 7(a) shows the calculated trajectory and Fig. 7(b) shows the torque curves obtained by applying the parallel solution scheme to this case. Although there is no visual difference between Fig. 7(a) and Fig. 6(a), the vibration in the torque curve for J1 is clearly reduced in Fig. 7(b). The slight rise observed near the starting point in Fig. 7(b) is due to both the initial velocity and acceleration being given as those of a rigid link system without a nonactuated joint, and the modeling error arising as a result of not considering the stiffness of Link 1.

Next, the calculated torque curves are verified by a different process. A bending moment is applied to base joint J1 of the two-link system and the motion is calculated by a mechanical analysis with the FEM. Then, the obtained trajectory is fed into the parallel solution scheme and the calculated torque curves are compared with the initial input moment. Although the calculated torques include the effects of internal forces occurring in the link system even though the input moment does not, the two curves should match well if the motion is not too fast. The model described previously is used in this analysis. The time increment for the calculation is set to 10 ms. Integration parameters for Newmark's method are given as δ=5/6 and β=4/9 in this example, to apply numerical dissipation and to avoid the occurrence of unnecessary high-order noise in the results [8]. Figure 8(a) shows the trajectory when an input moment was fed into the finite element analysis, and Fig. 8(b) shows the input moment and the calculated torque values. The two curves are in good agreement and again, it can be observed that no torque is required at the nonactuated joint, J2. Slight overshoots and undershoots observed in the calculated torque curve of J1 oroginate from the consideration of internal forces in the link system.

Lastly, a numerical example for an eight-joint underactuated link system is shown. The parameters of the links are as follows: material of the links SUS430; flexural stiffness 20.84 Nm^2; link lengths 0.1 m; link mass 4.95 $\times 10^{-2}$ kg; surplus mass at the tip 0.01 kg. The time increment for the calculation is set to 5 ms and numerical dissipation is considered (δ=5/6 and β=4/9). The joints are numbered from

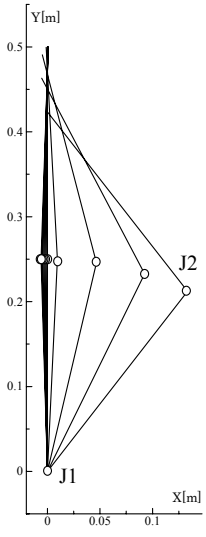


Fig. 8(a). Trajectory for a two-joint underactuated link system obtained by finite element analysis.

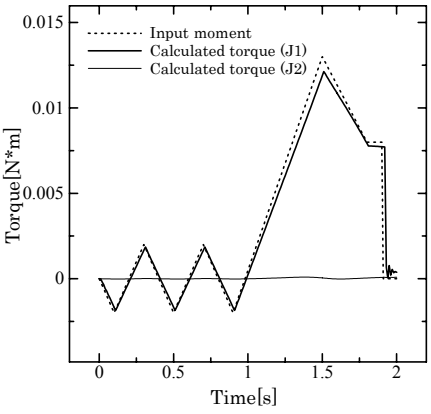


Fig. 8(b). Input moment and torque curves for a two-joint underactuated link system. Input moment is applied only to J1.

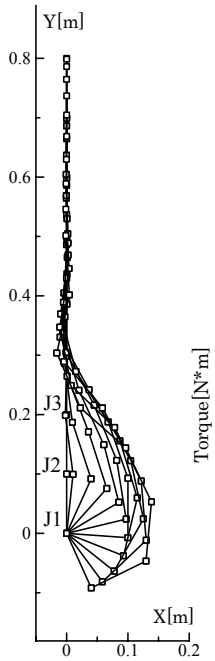


Fig. 9(a). Trajectory for an eight-joint underactuated link system calculated by finite element analysis.

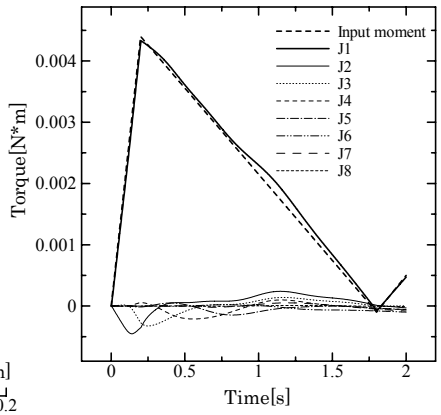


Fig. 9(b). Input moment and torque curves for an eight-joint underactuated link system. Input moment is applied only to J1.

the base joint as J1, J2, ⋯, J8. J1 is an active joint, and the other joints are all nonactuated joints. Figure 9(a) shows the calculated trajectory and Fig. 9(b) shows the input moment and torque curves. Although the required torque for J1 is very small in this case, the calculated torque matches the input moment, and the torques for other nonactuated joints maintain values nearly equal to 0.

V. Conclusion

The parallel solution scheme of inverse dynamics using high-order beam elements for modeling the link systems was developed, and the use of the scheme was extended to underactuated systems in this work. There are no special requirements when applying the scheme to the systems, except for the calculation of the correct trajectories. The trajectories can be calculated using the successive revision algorithm described in this paper, and basically, the use of the parallel solution scheme is not limited by the constituent number of links. Some feed-forward control experiments of underactuated link systems are now in progress.

Acknowledgment

This work was supported by Grant-in-Aid for Scientific Research (C) No.19560250, MEXT and the Electro-Mechanic Technology Advancing Foundation.

References

[1] Y. Nakamura and M. Ghodoussi, "Dynamics computation of closed-link robot mechanisms with nonredundant and redundant actuators," *IEEE Trans. Robotics and Automation*, vol. 5, no. 3, pp. 294-302, 1989.

[2] K. Sugimoto, "Dynamic analysis of closed loop mechanisms on the basis vectors of passive joint axes," *J. Robotic Systems*, vol. 20, no. 8, pp. 501-508, 2003.

[3] Y. Nakamura and K. Yamane, "Dynamics computation of structure-varying kinematic chains and its application to human figures," *IEEE Trans. Robotics and Automation*, vol. 16, no. 2, pp. 124-134, 2000.

[4] D. Isobe, "A unified solution scheme for inverse dynamics," *Advanced Robotics*, vol. 18, no. 9, pp. 859-880, 2004.

[5] D. Isobe and A. Kato, "Feed-forward control of flexible link systems using parallel solution scheme," *Int. J. Robotics and Automation*, 2008, in press.

[6] A. Jain and G. Rodriquez, "An analysis of the kinematics and dynamics of underactuated manipulators," *IEEE Trans. Robotics and Automation*, vol. 9, no. 4, pp. 411-422, 1993.

[7] T. Suzuki and Y. Nakamura, "Control of manipulators with free-joints via the averaging method," *IEEE Int. Conf. Robotics and Automation*, pp. 2998-3005, 1997.

[8] K.J. Bathe, *Finite Element Procedures*, Prentice Hall, 1996.