KSME Journal, Vol. 8, No. 2, pp. 115~126, 1994

Evaluation of Loop Constraints for Kinematic and Dynamic Modeling of General Closed-Chain Robotic Systems

Hee-Jun Kang* and R. A. Freeman**

(Received January 6, 1993)

In this article, the two algorithms evaluating loop constraints are presented for kinematic and dynamic modeling of general closed-chain robotic systems in terms of a system minimum set of coordinates. These procedures are based on higher order kinematic relationships between hypothetically open chain reference coordinates(system Lagrangian coordinates) and a set of independent closed-chain coordinates(system generalized coordinates). These relationships, along with principle of virtual work, allow for the determination of a system generalized coordinate based dynamic model in terms of the system Lagrangian coordinate based dynamic model in terms of the system Lagrangian coordinate based dynamic model(s). The proposed algorithms for determining/evaluating these relationships, both numerically and symbolically, are investigated and discussed with respect to their relative computational merits.

Key Words: Closed Link Mechanism, Tree Structured Mechanism, Loop Constraints, System Generalized Coordinate Based Dynamic Model.

Nomenclature

The notational scheme utilized in this paper underscores the transfer of system dependence concept central to the modeling algorithm presented herein. This is accomplished through a graphically descriptive form whereby parameters currently being considered dependent are obviously distinct from those currently being considered independent. The distinction basically results by employing subscripts when referring to independent parameters and allowing superscripts for dependent parametes. Refer to Figs. 1 and 2 when considering the following notation.

u : Common(end-effector) coordinate set of the given mechanism

ϕ : Lagrangian coordinate set of the given mechanism

$\phi_a = a$: Independent joint coordinate set of the given mechanism

$\phi_p = p$: Dependent joint coordinate set of the given mechanism

${}_r\phi$: Lagrangian coordinate set of r^{th} chain

${}_r\phi_a$: Independent joint coordinate set of r^{th} chain

${}_r\phi_p$: Dependent joint coordinate set of r^{th} chain

$[{}_rG_\phi^u]$: Open-chain jacobian of u in terms of ${}_r\phi$

${}_rg_j$: j^{th} column vector of $[{}_rG_\phi^u]$

$[{}_rG_a^u]$: Sub-matrix of $[{}_rG_\phi^u]$ made up of the columns corresponding to ${}_r\phi_a$

$[{}_rG_p^u]$: Sub-matrix of $[{}_rG_\phi^u]$ made up of the columns corresponding to ${}_r\phi_p$

N_L : Number of open-chain paths

N_a : Number of independent joints of the given mechanism

N_p : Number of independent joints of the given mechanism

N_j : Number of open-chain

N_o : Dimension of end-effector space of the given mechanism

*Assistant Professor, Department of Control and Instrumentation Engineering, University of Ulsan, Korea

**Assistant Professor, Department of Mechanical Engineering, The University of Texas, Tx., U. S. A.

1. Introduction

Closed link mechanisms have been widely used

in both industrial and research applications, because they are capable of moving faster and more precisely, and having more structural rigidity than still prevalent serial type robots. Cincinnati Milacron T^3, Bendix MA 510 and semi-direct drive manipulators designed by Asada (1983) contain single closed kinematic chain, while spherical shoulder(Cox, 1981) and Stewart Platform(Stewart, 1965) include multiple closed kinematic chains in their kinematic structure. In this paper, we address three different mehtods which embed the holonomic constraints caused by the closed kinematic chains into the equations of motion. They lead to deriving the kinematic and dynamic modeling parametes in terms of the generalized coordinate of the constrained robotic manipulators(system independent coordinate), not the generalized coordinates of their tree structured manipulators(Lagrangian coordinates of the system). The proposing procedures allow the inverse and forward dynamics of the robotic manipulators with single or multiple closed kinematic chain.

Dynamic modeling of closed chain mechanisms is commonly accomplished by applying the principle of virtual work to the model(s) of a corresponding tree structured(open chain) mechanism, which is obtained by hypothetically cutting joints(or links) until no closed kinematic chains exits. Utilizing this basic approach, Wittenburg(1977) derived the dynamic equation of closed-chain mechanism based on the motion of tree structured mechanism by using Lagrange multipliers and finally removing them. The motion equations he got, are expressed in system Lagrangian coordinate set. Integration routine might violate both the velocity loop constraints and position loop constraints. As an attempt to remedy this, he uses constraint stabilization method obtained from Baumgarte's idea(1972). Luh and Zheng(1985) formalized this methodology for inverse dynamic algorithm of closed-chain mechanism with Lagrange multiplier by replacing the constraints with the effect of unknown additional joint torques/forces on tree structured mechanism. Nakamura and Ghodoussi(1988) also obtained the inverse dynamic algorithm without Lagrange multiplier by using d'Alembert principle. Wittenburg(1983) refined his methodology to get the motion equations of closed chain mechanism in terms of the minimum actuating generalized coordinate set of the system. He assumed that system dependent coordinates can be explicitly written in independent coordinates and then symbolically differentiate the former with the latter to obtain constrained kinematic modeling parameters. Otherwise, numerical differentiation process is necessary. But, this assumption may not be true for a complicate closed-chain mechanism and numerical differentiation is liable to be a error source of the motion of system. Murray and Lovell(1989) integrated the whole idea to the inverse and forward dynamic algorithm in terms of minimum actuating generalized coordinate set of closed-chain mechanism. Walker(1985) also independently proposed a unified approach for inverse and forward dynamics of closed-chain systems. However, they might follow the constraint embedding procedure of Wittenburg (1983).

As seen in above literature, a widely employed approach to the dynamic modeling of general closed-chain mechanism can be summarized as the following three steps :

1. the dynamic modeling of a corresponding tree structured mechanism, which is obtained by hypothetically cutting joints(or links) until no closed kinematic chains exist ;

2. the evaluation of higher order kinematic relationships between a generalized coordinate set of the tree structured model(s) and a generalized coordinate set of the closed-chain mechanism ;

3. the dynamic modeling of the closed-chain system via the results of step 1 and step 2 and the principle of vitrual work.

In this article, we will concentrate on step 2 of the above procedure, rather than on step 1 and step 3. However, we will first discuss step 1, which is related to a judicial selection of computationally efficeint dynamic algorithm (Luh, et al., 1980 ; Walker and Orin, 1982) for open-chain linkage. Next, step 3 will be addressed to show which kinematic relationships need to be evaluated in step 2 to satisfy our purpose. Then,

three different approaches to obtain these relationships are presented. The first approach, based on position constraint differentiation, is discussed, since it is a commonly used method in robotics and multibody dynamic(Wittenburg, 1983 ; Paul, 1975). The second approach is based on the modification of higher order(velocity and acceleration) loop constraints. The third approach is obtained from the application of a set isomorphic transformation(Freeman and Tesar, 1988) to the above dynamic modeling prodedure. The proposed second and third approaches are different from the first approach in that they do not require differentiation. Those approaches, which embed the kinematic constraints into the equations of motion of closed-chain mechanism, can be used with computationally efficient open-chain dynamic algorithm selected in step 1(e. g, Recursive Newton-Euler algorithm) to obtain an efficient algorithm for the given closed-chain system.

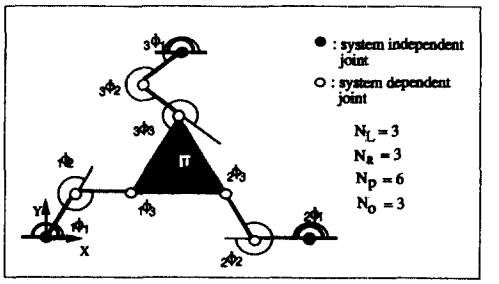

Fig. 1 3DOF planar shoulder

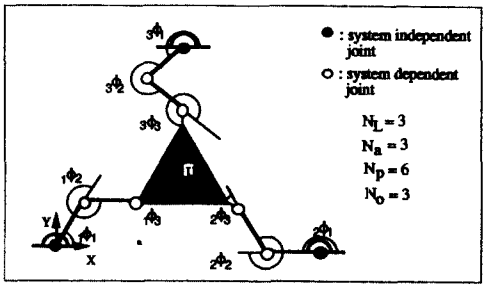

Fig. 2 The open tree structured mechanism corresponding to Fig. 1

2. Dynamic Modeling Algorithm

In this section, we briefly address the result format of the open-chain model of step 1 and develop the transformation equations of step 3 to show which kinematic relationships need to be evaluated(step 2) for the basic algorithm. This method is applied to the open-chain mechanism obtained by hypothetically cutting joints of the closed-chain mechanism. Here, these cut joint coordinates are still employed to describe a chosen intermediate coordinate set, often the end-effector coordinates($e.g.,u$) for parallel manipulators. Figure 2 shows the open tree structured mechanism of Fig. 1.

2.1 Open-chain modeling

Here, only the resulting format of the higher-order kinematics for the r^{th} open-chain is given. The problem of position analyzing is not addressable(except in an iterative, or differential displacement). Adopting the standard Jacobian $[{}_rG_a^u]$ representation for the velocity of a vector of N_o intermediate coordinates, u, in terms of a set of N_j independent coordinates, ${}_r\dot{\phi}$, of the r^{th} open-chain, one has

$$\dot{u} = [{}_rG_\phi^u]\,{}_r\dot{\phi} \tag{1}$$

Here,

$$[{}_rG_\phi^u] = \left[\frac{\partial u}{\partial {}_r\phi_1}\ \frac{\partial u}{\partial {}_r\phi_2}\ \cdots\ \frac{\partial u}{\partial {}_r\phi_{N_j}}\right] = [{}_rg_1^u\ {}_rg_2^u\ \cdots\ {}_rg_{N_j}^u] \tag{2}$$

is the Jacobian relating the coordinate u and ${}_r\phi$ with the n^{th} column ${}_rg_n^u$ being of dimension $N_o \times 1$. Generally, the acceleration vector, $\ddot{u}$, of a set of N_o dependent parameters, u, is obtained from the differentiation of Eq. (1) with respect to($w.r.t$) time as

$$\ddot{u} = [{}_rG_\phi^u]\,{}_r\ddot{\phi} + ([{}_r\dot{G}_\phi^u])\,{}_r\phi = [{}_rG_\phi{}^u]\,{}_r\ddot{\phi} + H({}_r\phi, {}_r\phi) \tag{3}$$

where $H({}_r\phi, {}_r\dot{\phi})$ is a centripetal and Coriolis acceleration vector of dimension $N_o \times 1$. The required inertial torque, ${}_rT_\phi$, of the r^{th} open-chain can be described in terms of position and time derivatives of its joint parameters, or in joint space with the general form as

$${}_rT_\phi = [{}_rI_{\phi\phi}^*]\,{}_r\ddot{\phi} + C({}_r\phi, {}_r\phi) \tag{4}$$

where $[{}_rI_{\phi\phi}^*]$ is the $N_j \times N_j$ inertia matrix of the

r^{th} open-chain, with $C({}_r\phi, {}_r\dot{\phi})$ being an $N_j \times 1$ vector of centrifugal, Coriolis and gravity terms. The entire open-chain dynamic of the given closed-chain mechanism can be expressed is an augmented form as follws

$$T_\phi = [I^*_{\phi\phi}]\ddot{\phi} + C(\phi, \dot{\phi}) \tag{5}$$

where the r^{th} diagonal submatrix of $[I^*_{\phi\phi}]$ is $[{}_rI^*_{\phi\phi}]$, with the r^{th} element of T_ϕ and $C(\phi, \dot{\phi})$ being ${}_rT_\phi$ and $C({}_r\phi, {}_r\dot{\phi})$, respectively.

2.2 Dynamic modeling of general closed-chain system

In this section, the principle of virtual work is employed to show the relationship between the open chain modeling parameters and the modeling parameters of the closed-chain mechanism. By using the principle of virtual work, the open chain dynamics can be directly incorporated into the closed chain dynamics according to

$$T_\phi \cdot \delta\phi = T_a \cdot \delta\phi_a \tag{6}$$

where T_a is the required inertial torque of the closed chain mechanism, corresponding to ϕ_a.

Assuming that we have the kinematic models(which will be developed in the next section in three different ways), which relate the Lagrangian coordinate set(ϕ) to the independent coordinate set(ϕ_a) of the given mechanism,

$$\dot{\phi} = [G^\phi_a]\dot{\phi}_a \tag{7}$$

$$\ddot{\phi} = [G^\phi_a]\ddot{\phi}_a + ([\dot{G}^\phi_a])\dot{\phi}_a \tag{8}$$

the total system dynamic is obtained as follows

$$\begin{aligned} T_a &= [G^\phi_a]^T T_\phi \\ &= [G^\phi_a]^T\{[I^*_{\phi\phi}]\ddot{\phi} + C(\phi, \dot{\phi})\} \end{aligned} \tag{9}$$

In a compact form, the dynamic model will be expressed in terms of a minimum set of coordinates(ϕ_a) as

$$\begin{aligned} T_a &= [G^\phi_a]^T[I^*_{\phi\phi}][G^\phi_a]\ddot{\phi}_a + [G^\phi_a]^T[I^*_{\phi\phi}]([\dot{G}^\phi_a])\dot{\phi}_a \\ &\quad + [G^\phi_a]^T C(\phi, \dot{\phi}) \\ &= [I^*_{aa}]\ddot{\phi}_a + C_a(\phi, \dot{\phi}) \end{aligned} \tag{10}$$

where

$$[I^*_{aa}] = [G^\phi_a]^T[I^*_{\phi\phi}][G^\phi_a] \tag{11}$$

$$\begin{aligned} C_a(\phi, \dot{\phi}) &= [G^\phi_a]^T([\dot{G}^\phi_a])\dot{\phi}_a \\ &\quad + [G^\phi_a]^T C(\phi, \dot{\phi}) \end{aligned} \tag{12}$$

Eqs. (9) and (10) show what needs to be evaluated in step 2 for the inverse dynamic algorithm($[G^\phi_a]$) and for the forward dynamic algorithm(both $[G^\phi_a]$ and $([\dot{G}^\phi_a]\dot{\phi}_a)$). And also, the modeling parameters obtained from the open chain kinematics, dynamics and the kinematic constraints of the given system result in Eqs. (11) and (12) being decoupled, allowing them to be independently evaluated for each chain. Hence, Eqs. (11) and (12) become

$$[I^*_{aa}] = \sum_{r=1}^{N_L} [{}_rG_a]^T[{}_rI^*_{\phi\phi}][{}_rG_a] \tag{13}$$

$$\begin{aligned} C_a(\phi, \dot{\phi}) &= \sum_{r=1}^{N_L}\{[{}_rG_a]^T[{}_rI^*_{\phi\phi}]([{}_r\dot{G}_a]\dot{\phi}_a) \\ &\quad + [{}_rG_a]^T C({}_r\phi, {}_r\dot{\phi})\} \end{aligned} \tag{14}$$

where $[{}_rG_a]$ and $([{}_r\dot{G}_a]\dot{\phi}_a)$ will be shown in Eqs. (48) and (51), which are the submatrix/subvector of $[G^\phi_a]$ and $([\dot{G}^\phi_a]\dot{\phi}_a)$ corresponding to r^{th} open chain, respectively.

Equations (13) and (14) show the algorithm parallelism which will allow synchronous parallel computation of each open-chain modeling. In fact, the total amount of computational effort of the closed-chain system is largely dependent on the computation required for the open-chain dynamic model.

3. Evaluation of Loop Constraints for Modeling of Closed Chain System

As previously mentioned, three different methods to obtain the kinematic model(both $[G^\phi_a]$ and $[\dot{G}^\phi_a]\dot{\phi}_a$) are discussed. The first approach, called position constraint differentiation method, will be discussed, since it is a commonly used method in robotics and multibody dynamics, Then, the two proposed approaches will be discussed, which are mainly different from the first approach in that they do not require differentiation.

3.1 Position constraint differentiation method

Here, a system is assumed to have closed kinematic chains. The holonomic constraint equations are expressed in terms of system Lagrangian coordinates, or sets of dependent and independent coordinates, as follows

$$\begin{aligned} &f(\phi) = f(\phi_a, \phi_p) = 0 \\ &\text{with } \phi = [\phi_a^T : \phi_p^T]^T \end{aligned} \tag{15}$$

Jacobian matrix, which relates the dependent

coordinates to the independent coordinates, can be obtained by total differentiation of Eq. (15) as

$$\left[\frac{\partial f}{\partial a}\right]\dot{\phi}_a+\left[\frac{\partial f}{\partial p}\right]\dot{\phi}_p=0 \tag{16}$$

where p and a imply ϕ_p and ϕ_a, respectively, and $[\partial f/\partial a]$ is an $N_p\times N_a$ matrix whose i^{th} row and j^{th} column element is $\partial f_i/\partial a_j$ and $[\partial f/\partial p]$ is an $N_p\times N_p$ matrix with $\partial f_i/\partial p_j$ as its i^{th} row and j^{th} column element. Proceeding furhter by solving Eq. (16) for $\dot{\phi}_p$, we have

$$\dot{\phi}_p=-\left[\frac{\partial f}{\partial p}\right]^{-1}\left[\frac{\partial f}{\partial a}\right]\dot{\phi}_a \tag{17}$$

where the nonsingularity of matrix $[\partial f/\partial p]$ is assumed. Now, we define the first order internal kinematic influence coefficient(IKIC) matrix of a closed-chain system as

$$[G_a^p]=-\left[\frac{\partial f}{\partial p}\right]^{-1}\left[\frac{\partial f}{\partial a}\right] \tag{18}$$

where p and q imply ϕ_p and ϕ_a, respectively. Using this definition, Eq. (17) can be written as

$$\dot{\phi}_p=[G_a^p]\dot{\phi}_a \tag{19}$$

The direct differentiation of Eq. (19) $w.r.t.$ time gives

$$\ddot{\phi}_p=[G_a^p]\ddot{\phi}_a+([\dot{G}_a^p])\dot{\phi}_a \tag{20}$$

Note that the definitions of the matrices $[G_a^p]$ and $([\dot{G}_a^p])$ given in Eqs. (18) and (20), respectively, involve explicit partial differentiation or implicit numerical differentiation of the constraints of Eq. (15) $w.r.t.$ their arguments/time up to the second order. Actual evaluation of these matrices can be accomplished either by straghtforward differentiation of the positional constraint equations according to the definition, which may be a simple task for planar closed kinematic chains, or by starting directly from the velocity and acceleraton constraint of the closed kinematic chain(see the following two methods). The latter approach is appealing for the case of spatially closed kinematic chains, considering the algebraic complexity of the positional constraint equations.

3.2 Higher order loop constraint method

The higher order($i.e.$, velocity and acceleration) constraint equations obtained from the direct differentiation of the position constraint equations seen in the prior section can also be obtained directly at the velocity and acceleration level by using a common intermediate coordinate set. For example, the end-effector coordinate set of a parallel manipulator(Fig. 1) can be a convenient common intermediate coordinate set. This method uses the predetermined open tree structured kinematic modeling parameters without considering the position constraint equations for the constraint determination. The velocity and acceleration vector of the end-effector can be expressed directly in terms of the joint velocities and accelerations of the r^{th} chain of the open tree structured according to Eqs. (1) and (3), as

$$\dot{u}=[{}_rG_\phi^u]{}_r\dot{\phi} \tag{21}$$

$$\ddot{u}=[{}_rG_\phi^u]{}_r\ddot{\phi}+H({}_r\phi, {}_r\dot{\phi}),\ r=1, 2, \cdots, N_L \tag{22}$$

Eq. (21) implies that there are $N_o\times(N_L-1)$ algebraic equations relating one of the joint velocity sets to each of the other sets. This can be expressed as

$$[{}_1G_\phi^u]{}_1\dot{\phi}=[{}_rG_\phi^u]{}_r\dot{\phi},\quad r=2, 3, \cdots, N_L \tag{23}$$

To show the column vectors of the Jacobian matrices, Eq. (23) can be rewritten in detail as

$$[{}_1g_1 : {}_1g_2 \cdots {}_1g_{Nj}]\begin{bmatrix}{}_1\dot{\phi}_1\\ {}_1\dot{\phi}_2\\ \vdots\\ {}_1\dot{\phi}_{Nj}\end{bmatrix}=[{}_rg_1 : {}_rg_2 \cdots {}_rg_{Nj}]\begin{bmatrix}{}_r\dot{\phi}_1\\ {}_r\dot{\phi}_2\\ \vdots\\ {}_r\dot{\phi}_{Nj}\end{bmatrix},\quad r=2, 3, \cdots, N_L \tag{24}$$

Eq. (24) can be expressed as a linear relationship between the column vectors of the Jacobians

$$\sum_{k=1}^{Nj}{}_1g_k\ {}_1\dot{\phi}_k=\sum_{k=1}^{Nj}{}_rg_k\ {}_r\dot{\phi}_k,\ r=2, 3, \cdots, N_L \tag{25}$$

The linear relations of Eq. (25) can be rearranged and regrouped according to the independent and dependent coordinate velocity sets of each chain as

$$[{}_1G_a^u]{}_1\dot{\phi}_a+[{}_1G_p^u]{}_1\dot{\phi}_p=[{}_rG_a^u]{}_r\dot{\phi}_a+[{}_rG_p^u]{}_r\dot{\phi}_p,\quad r=2, 3, \cdots, N_L \tag{26}$$

The (N_L-1) matrix equations of (26) can be augmented into single matrix equations as shown below

$$\begin{bmatrix}[{}_1G_p^u] & -[{}_2G_p^u] & [0] & \cdots & [0]\\ [{}_1G_p^u] & [0] & -[{}_3G_p^u] & \cdots & [0]\\ \vdots & & & & \vdots\\ [{}_1G_p^u] & [0] & [0] & \cdots & -[{}_{N_L}G_p^u]\end{bmatrix}\begin{bmatrix}{}_1\dot{\phi}_p\\ {}_2\dot{\phi}_p\\ \vdots\\ {}_{N_L}\dot{\phi}_p\end{bmatrix}$$

$$= \begin{bmatrix} -[{}_1G_a^u] & [{}_2G_a^u] & [0] & \cdots & [0] \\ -[{}_1G_a^u] & [0] & [{}_3G_a^u] & \cdots & [0] \\ \vdots & & & & \vdots \\ -[{}_1G_a^u] & [0] & [0] & \cdots & [{}_{N_L}G_a^u] \end{bmatrix} \begin{bmatrix} {}_1\dot{\phi}_a \\ {}_2\dot{\phi}_a \\ \vdots \\ {}_{N_L}\dot{\phi}_a \end{bmatrix} \quad (27)$$

Now, Eq. (27) can be expressed simply as

$$[A]\dot{\phi}_p = [B]\dot{\phi}_a \quad (28)$$

whrere $[A]$ is an $N_p \times N_p$ matrix and $[B]$ is an $N_p \times N_a$ matrix, since $N_p = N_o \times (N_L - 1)$. See Appendix.

Direct inversion of matrix $[A]$, which is assumed to be a nonsingular, gives

$$\dot{\phi}_p = [G_a^p]\dot{\phi}_a,\ [G_a^p] = [A]^{-1}[B] \quad (29)$$

where $[G_a^p]$ is the first order IKIC matrix of the given closed-chain system required for both inverse and forward dynamics.

To obtain the second-order kinematic relationship between the dependent and independent coordinate sets, required for the forward dynamics, Eq. (22) can be rewritten as

$$[{}_1G_a^u]{}_1\ddot{\phi}_a + [{}_1G_p^u]{}_1\ddot{\phi}_p = [{}_rG_a^u]{}_r\ddot{\phi}_a + [{}_rG_p^u]{}_r\ddot{\phi}_p + H({}_r\phi, {}_r\dot{\phi}) - H({}_1\phi, {}_1\dot{\phi}),\ r = 2, 3, \cdots, N_L \quad (30)$$

The $(N_L - 1)$ matrix equations can be rearranged and regrouped as

$$\begin{bmatrix} [{}_1G_p^u] & -[{}_2G_p^u] & [0] & \cdots & [0] \\ [{}_1G_p^u] & [0] & -[{}_3G_p^u] & \cdots & [0] \\ \vdots & & & & \vdots \\ [{}_1G_p^u] & [0] & [0] & \cdots & -[{}_{N_L}G_p^u] \end{bmatrix} \begin{bmatrix} {}_1\ddot{\phi}_p \\ {}_2\ddot{\phi}_p \\ \vdots \\ {}_{N_L}\ddot{\phi}_p \end{bmatrix}$$
$$= \begin{bmatrix} -[{}_1G_a^u] & [{}_2G_a^u] & [0] & \cdots & [0] \\ -[{}_1G_a^u] & [0] & [{}_3G_a^u] & \cdots & [0] \\ \vdots & & & & \vdots \\ -[{}_1G_a^u] & [0] & [0] & \cdots & [{}_{N_L}G_a^u] \end{bmatrix} \begin{bmatrix} {}_1\ddot{\phi}_a \\ {}_2\ddot{\phi}_a \\ \vdots \\ {}_{N_L}\ddot{\phi}_a \end{bmatrix}$$
$$+ \begin{bmatrix} H({}_2\phi, {}_2\dot{\phi}) - H({}_1\phi, {}_1\dot{\phi}) \\ H({}_3\phi, {}_3\dot{\phi}) - H({}_1\phi, {}_1\dot{\phi}) \\ \vdots \\ H({}_{N_L}\phi, {}_{N_L}\dot{\phi}) - H({}_1\phi, {}_1\dot{\phi}) \end{bmatrix} \quad (31)$$

A notationally simple expression of Eq. (31) is

$$[A]\ddot{\phi}_p = [B]\ddot{\phi}_a + Q \quad (32)$$

Now, solving for the dependent parameters accelerations, we have

$$\ddot{\phi}_p = [A]^{-1}[B]\ddot{\phi}_a + [A]^{-1}Q = [G_a^p]\ddot{\phi}_a + ([\dot{G}_a^p]\dot{\phi}_a) \quad (33)$$

3.3 Intermediate coordinate transfer approach

Another method for evaluating kinematic constraints is accomplished based on the application of isomorphic transformation of kinematic models(Freeman and Tesar, 1988) to the presenting dynamic modeling procedure, and is particularly effective for the multiple nonredundant, closed chain situation. Recalling Eqs. (21) and (22), the higher order inverse kinematics of each open chain are

$${}_r\dot{\phi} = [{}_rG_\phi^u]^{-1}\dot{u} = [{}_rG_u^\phi]\dot{u} \quad (34)$$

and

$${}_r\ddot{\phi} = [{}_rG_\phi^u]^{-1}(\ddot{u} - H({}_r\phi, {}_r\dot{\phi})) = [{}^rG_u^\phi]\ddot{u} + {}_rV \quad (35)$$

where

$${}_rV = -[{}_rG_u^\phi]H({}_r\phi, {}_r\dot{\phi}) \quad (36)$$

Now, we select the kinematic influence coefficients relating the independent coordinate set(ϕ_a), and the dependent coordinate set(ϕ_p) to the common coordinate(u)(from the transferred models in Eqs. (34) and (36))

$$[G_u^a] = \begin{bmatrix} [{}_1G_u^\phi]_a; \\ [{}_2G_u^\phi]_a; \\ \vdots \\ [{}_{N_L}G_u^\phi]_a; \end{bmatrix} (N_a \times N_o),$$
$$[G_u^p] = \begin{bmatrix} [{}_1G_u^\phi]_p; \\ [{}_2G_u^\phi]_p; \\ \vdots \\ [{}_{N_L}G_u^\phi]_p; \end{bmatrix} (N_p \times N_o) \quad (37)$$

and

$$V_a = \begin{bmatrix} {}_1V_a \\ {}_2V_a \\ \vdots \\ {}_{N_L}V_a \end{bmatrix} (N_a \times 1),\ V_p = \begin{bmatrix} {}_1V_p \\ {}_2V_p \\ \vdots \\ {}_{N_L}V_p \end{bmatrix} (N_p \times 1) \quad (38)$$

where the subscripts, a and p, imply the independent and coordinate sets for each open chain, respectively.

Recalling Eqs. (34) and (35) and the following forward kinematic relations,

$$\dot{u} = [G_a^u]\dot{\phi}_a \quad (39)$$
$$\ddot{u} = [G_a^u]\ddot{\phi}_a + H(\phi_a, \dot{\phi}_a) \quad (40)$$

with

$$[\dot{G}_a^u] = [G_u^a]^{-1} \text{ and } H(\phi_a, \dot{\phi}_a) = -[G_u^a]^{-1}V_a \quad (41)$$

Now, the velocity of the dependent coordinate set will be expressed in terms of the independent coordinate set as

$$\dot{\phi}_p=[G_u^p]\dot{u}=[G_u^p][G_u^a]^{-1}\dot{\phi}_a=[G_a^p]\dot{\phi}_a \tag{42}$$

since, by definition, $N_o=N_a$ for the nonredundant situation, and the acceleration of the dependent coordinate set will also be expressed as

$$\ddot{\phi}_p=[G_u^p]\ddot{u}+V_p=[G_u^p][G_u^a]^{-1}\ddot{\phi}_a -[G_u^p][G_u^a]^{-1}V_a+V_p \tag{43}$$

Thus, the final acceleration equations is formed as follows

$$\ddot{\phi}_p=[G_a^p]\ddot{\phi}_a+([\dot{G}_a^p])\dot{\phi}_a \tag{44}$$

The final corresponding transferred model($[G_a^p]$, $([\dot{G}_a^p])\dot{\phi}_a$) is obtained as

$$[G_a^p]=[G_u^p][G_u^p]^{-1} \tag{45}$$

and

$$([\dot{G}_a^p])\dot{\phi}_a)=V_p-[G_a^p]V_a \tag{46}$$

The kinematic modeling parameters($[G_a^p]$, $([\dot{G}_a^p])\dot{\phi}_a)$) obtained from above three different methods can be augmented to describe the kinematic modeling parameters ($[G_a^\phi]$, $([\dot{G}_a^\phi])\dot{\phi}_a)$) required for the dynamic evaluation in Eqs. (11) and (12).

For the first order kinematic modeling parameters($[G_a^\phi]$), the joint velocities of each open-chain are first expressed, in terms of velocity of the generalized coordinate set, as

$$_r\dot{\phi}=\begin{bmatrix} {}_r\dot{\phi}_a \\ {}_r\dot{\phi}_p \end{bmatrix}=\begin{bmatrix} [0]_1[0]_2\cdots[I]_r\cdots[0]_{N_a} \\ [G_a^p]_r; \end{bmatrix}\dot{\phi}_a =[{}_rG_a]\dot{\phi}_a,\ r=1,2,3,\cdots,N_L. \tag{47}$$

where$[{}_rG_a]$ is the first order IKIC matrix relating the joint parameters of the r^{th} chain to the independent joint parameters of the total system. $[I]_r$ is the identity matrix, which is located in r^{th} column block and has dimension of $\dim({}_r\phi_a)\times\dim({}_r\phi_a)$. $[G_a^p]_r$; is the r^{th} row of the first-order IKIC matrix relating the dependent coordinate set of the r^{th} chain to the independent coordinate set of the total system and is of dimension$(N_j-\dim({}_r\phi_a))\times N_a$.

The matrices given by Eq. (47) can be directly augmented to describe all joint velocities in terms of the velocity of the independent coordinate set as

$$\dot{\phi}=\begin{bmatrix} [{}_1G_a] \\ [{}_2G_a] \\ \vdots \\ [{}_{N_L}G_a] \end{bmatrix}\dot{\phi}_a=[G_a^\phi]\dot{\phi}_a \tag{48}$$

For the second order modeling parameters($[\dot{G}_a^\phi]\dot{\phi}_a$), the joint accelerations of each open chain are expressed, in terms of the velocity and acceleration of the generalized coordinate set, as

$$_r\ddot{p}=\begin{bmatrix} {}_r\ddot{\phi}_a \\ {}_r\ddot{\phi}_p \end{bmatrix}=[{}_rG_a]\ddot{\phi}_a+([{}_r\dot{G}_a]\dot{\phi}_a) =[{}_rG_a]\ddot{\phi}_a+\begin{bmatrix} 0 \\ \cdots\cdots \\ ([{}_r\dot{G}_a^p]\dot{\phi}_a) \end{bmatrix},\quad r=1,2,\cdots,N_L \tag{49}$$

where $([{}_r\dot{G}_a^p]\dot{\phi}_a)$ is the second order kinematic relationship between the joint parameters of the r^{th} chain and the independent joint parameters of the total system. 0 is the null vector, and its dimension is $\dim({}_r\phi_a)\times N_a\times N_a$. $([{}_r\dot{G}_a^p]\dot{\phi}_a)$ is the second order kinematic relationship between the dependent coordinate of the r^{th} chain to the independent coordinate set of the total system and is of dimension $(N_j-\dim({}_r\phi_a))\times N_a\times N_a$.

The matrices seen in Eq. (49) can be directly augmented to describe all joint accelerations in termas of velocity and acceleration of the minimum actuating coordinate set as

$$\ddot{\phi}=\begin{bmatrix} \ddot{\phi}_a \\ \ddot{\phi}_p \end{bmatrix}=[G_a^\phi]\ddot{\phi}_a+([\dot{G}_a^\phi]\dot{\phi}_a) \tag{50}$$

where

$$([\dot{G}_a^\phi]\dot{\phi}_a)=\begin{bmatrix} ([{}_1\dot{G}_a]\dot{\phi}_a) \\ ([{}_2\dot{G}_a]\dot{\phi}_a) \\ \cdots\cdots \\ ([{}_{N_L}\dot{G}_a]\dot{\phi}_a) \end{bmatrix} \tag{51}$$

The algorithm presented in last Secs. (3.2) and (3.3), yielding$[G_a^\phi]$ and $([\dot{G}_a^\phi]\dot{\phi}_a)$ can be applied to general robotic mechanisms, including ones having several internal modular structures. This requires judicial placement of the common intermediate coordinate set.

4. Dynamic Simulation of Closed Chain Robotic Systems

In this section, some additional modeling

parameters required for the simulation of the closed-chain robotic systems are obtained, which are based on the previously computed modeling parameters in last sections. Then, the overall procedures of both inverse and forward dynamic simulations are given here for the summary of the proposed modeling methodology.

4.1 Higher order forward kinematics

For any controlled motion of the end-effector of a given robotic system, it is required to have an explicit expression of the controlled variables(the output motion) in terms of the controlling variables(the actuating joint torques). This is accomplished by the external kinematic modeling parameters which address the kinematic relationship between the end-effector motion and the generalized coordinate set of the given system. Having the constrained joint relationships of Sec. 3 makes the higher order forward kinematics straightforward by using the open chain kinematic modeling parameters. The 1^{st} open-chain is usually taken to describe the end-effector motion, where the 1^{st} open-chain runs from the robot base to the robot end-effector. The velocities of the end-effector are directly obtained by inserting Eq. (47) into Eq. (21)

$$\dot{u} = [{}_1G^u_\phi]\,{}_1\dot{\phi} = [{}_1G^u_\phi][{}_1G_a]\,\dot{\phi}_a = [G^u_a]\,\dot{\phi}_a \tag{52}$$

where first order external KIC matrix is

$$[G^u_\phi] = [{}_1G^u_\phi][{}_1G_a] \tag{53}$$

The accelerations are obtained by inserting Eqs. (49) and (52) into Eq. (22) as

$$\begin{aligned}\ddot{u} &= [{}_1G^u_\phi]([{}_1G_a]\,\ddot{\phi}_a + [{}_1\dot{G}_a]\,\dot{\phi}_a) + H({}_1\phi, {}_1\dot{\phi}) \\ &= [G^u_a]\,\ddot{\phi}_a + H_a\end{aligned} \tag{54}$$

where the second order external kinematic modeling parameters is

$$H_a = [{}_1G^u_\phi] \cdot [{}_1G_a]\,\dot{\phi}_a + H({}_1\phi, {}_1\dot{\phi}) \tag{55}$$

These external kinematic modeling parameters shoule be evaluated for both forward and inverse dynamic analysis.

4.2 Determination of underdetermined torque sets

When a system is driven by the actuation of more joints then there are kinematic freedoms, it is called a redundantly actuated(mathematically, under-determined) system. Multiple cooperating manipulators and multi-fingered hands might be included in this class of mechanism. The problem of specifying 'optimal' torque sets for redundantly actuated system is another active research area considering force redundancy as well as motion redundancy. Here, the conventional joint torque minimization is applied to distribute the required loads among the redundantly actuated joints.

Once the minimum actuating joint torque set is evaluated, using the method of the previous sections, the underdetermined torque set corresponding to a selected underdetermined joint set ϕ_s is obtained by using the principle of virtual work, as

$$T_{\phi_s} \cdot \delta\phi_s = T_a \cdot \delta\phi_a \tag{56}$$

Selecting the first order internal KIC matrix from Eq. (48) gives the relationship as

$$[G^s_a]^T T_s = T_a \tag{57}$$

In order to optimize the selected joint torques, the local joint torque minimization scheme was performed by means of the criterion

$$\min(T_s - (T_s)_n)^T [W](T_s - (T_s)_n)$$

where the desired torque vector $(T_s)_n$ is equal to $(T_{min} + T_{max})/2$, and T_{min} and T_{max} are the lower limit and upper limit of joint torque vectors, respectively, and $[W]$ is a weighting matrix determined by capacity of each joint actuator.

This is a straightforward least square problem with the solution given by

$$\begin{aligned}T_s = &([G^s_a]^T)^+ T_a \\ &+ ([I] - ([G^s_a]^T)^+ [G^s_a]^T)(T_s)_n\end{aligned} \tag{58}$$

where the weighted pseudo inverse of $([G^s_a]^T)$ is as follows

$$([G^s_a]^T)^+ = [W]^{-1}[G^s_a]([G^s_a]^T[W]^{-1}[G^s_a])^{-1} \tag{59}$$

The distributed torque set causes the same motion with generally lower input load magnitudes avoiding the torque limit of each joint than required of a minimum actuating torque set.

4.3 Prodedures of dynamic simulation

Inverse dynamic simulation of given object motion trajectory is executed according to the following steps:

(1) Computes the open-chain modeling parameters based on the current ϕ_i and $\dot{\phi}_i$ ($[{}_rG^u_\phi]_i$, $H({}_r\phi_i, {}_r\dot{\phi}_i)$, $[{}_rI^*_{\phi\phi}]_i$, $Q({}_r\phi_i, {}_r\dot{\phi}_i)$).

(2) Computes the kinematic constraint modeling parameters

($[G_a^p]_i$, $([\dot{G}_a^p]\dot{\phi}_a)_i$, $[G_a^q]_i$, $[\dot{G}_a^q]\dot{\phi}_a)_i$, $[G_a^u]_i$ and $H_a(\phi_i, \dot{\phi}_i)$).

(3) Computes closed-chain dynamic modeling parameters and the required torque as

$(T_a)_i=[I_{aa}^*]_i(\ddot{\phi}_a)_i+Q_a(\phi_i, \dot{\phi}_i)$.

(4) Computes the next minimum set of accelerations$(\ddot{\phi}_a)_{i+1}$ from the inverse kinematic using $[G_a^u]_i$ and $H_a(\phi_i, \dot{\phi}_i)$.

(5) Computes the next minimum set of velocities $(\dot{\phi}_a)_{i+1}$ from the internal kinematic relationship as $(\dot{\phi}_a)_{i+1}=(\phi_a)_i+(\dot{\phi}_a)_i dt$.

(6) Computes the next minimum set of velocities $(\dot{\phi}_p)_{i+1}$ from the internal kinematic relationship as $(\dot{\phi}_p)_i=[G_a^p]_i(\dot{\phi}_a)_i$.

(7) Computes the next all joint positions ϕ_{i+1} from the direct integration as $\phi_{i+1}=\phi_i+\dot{\phi}_i$ dt.

Forward dynamic simulation of given torque history is executed according to the following steps :

(1) Same as 1) and 2) of inverse dynamic.

(2) Computes the next minimum set of accelerations$(\ddot{\phi}_a)_{i+1}$ from the closed-chain dynamic as $(\ddot{\phi}_a)_i=[I_{aa}^*]_i^{-1}\{(T_a)_i-Q_a(\phi_i, \dot{\phi}_i)\}$.

(3) Same as 5) and 6) of inverse dynamics.

5. Discussion

We presented two method to determine$[G_a^q]$ and $([\dot{G}_a^q]\dot{\phi}_a)$ used to obtain the dynamic modeling parameters for general closed-chain system, with position constraint differentation method widely used in robotics. The coefficient $[G_a^q]$ is used to transfer the generalized loads required by the tree structured motion to the independent actuation coordinate of the closed-chain system(inverse dynamic), and the coefficient$([\dot{G}_a^q]\dot{\phi}_a)$ is necessary to express the constrained motion in terms of the generalized coordinate of the system(forward dynamic).

For comparison of the methods, initial computational assessment has been performed for the determination of those modeling parameters require to constrain(transfer) the open chain dynamic models to the desired closed chain actuation set(ϕ_a. The computation counts for each major step are given in Table 1. The mechanisms are considered "fully parallel," since $N_a=N_o=N_j=N_L$. The example mechanisms are a 3 DOF Shoulder mechanism and a 6 DOF Stewart Plaform. The computation numbers(shown in Table 1) for the Intermediate Coordinate Transfer Method(ICTM) are significantly less than those for the Higher Order Constraint Method(HOCM) in both cases. This is due that the computation numbers for the HOCM are generally dependent on the higher order polynomial of N_p, while those for the ICTM are generally dependent on the higher order polynomial of N_j, where N_p is much bigger than N_j for the 6 DOF Stewart Platform. And also, the ICTM naturally allows for kinematic models($[G_a^u]$ and $[\dot{G}_a^q]\dot{\phi}_a$) which are indispensable for the control of the closed chain system. This assessment was not performed in detail for the position constraint differentiation method(PCDM) due to its direct dependence on the differentiation of the holonomic constraint equations. It should be noted, however, that, after the first differentiation, the determination of the first order KIC is the same as for the HOCM. This indicates that the computation numbers for the PCDM are also dependent on the higher order polynomial of N_p and that the differentiation steps are in relative excess.

In order to verify the above observation, symbolic generation of $[G_a^q]$ has been performed by the three different methods. The PCDM involves explicit differentiation of position constraint equations, while the other two methods involve symbolic generation of the first order KIC matrix($[G_\phi^u]$) of each open chain. For comparison, the three methods are applied to the planar shoulder mechanism seen in Fig. 1, using Mathematica on an IBM PC. Again, the ICTM is the most efficient both in manipulation time and ease of reducibility into the final form(given in Kang and Freeman, 1990-b). This is expected in that the ICTM involves multiple inversions of matrices of the dimension of the task coordinate set, while the other two methods involve the single inversion of a generally larger matrix of the dimension of the dependent coordinate set, as seen in Table 2. In addition, symbolic inversion is

Table 1 Number of multiplications and additions required to generate the kinematic modeling parameters($[G_a^\phi]$ and $([\dot{G}_a^\phi]\dot{\phi}_a)$)***

Higher order loop constraint method			Intermediate coordinate transfer method		
Majou steps	Mults.	Adds.	Major steps	Mults.	Adds.
Eq. (29) $[A]^{-1}$**	N_P^3	$\frac{1}{2}N_P^3$	Eq. (3-20) $[{}_rG_\phi^u]^{-1}$	$N_L N_j^{3*}$	$\frac{1}{2}N_L N_j^{3*}$
Eq. (29) $[A]^{-1}[B]$	$N_P^2 N_a$	$N_P N_a(N_P-1)$	Eq. (3-27) $[G_u^a]^{-1}$	N_j^3	$\frac{1}{2}N_j^3$
Eq. (32) Q		$N_P(N_L-1)$	Eq. (3-28) $[G_u^p][G_u^a]-1$	$N_p N_j^2$	$N_p N_j(N_j-1)$
Eq. (33) $[A]^{-1}Q$	N_p^2	$N_P(N_p-1)$	Eq. (3-32) $([\dot{G}_a^\phi]\dot{\phi}_a)$	$N_p N_a$	$N_p N_a$
3 DOF Shoulder	360	240		180/126*	108/81*
6 DOF Stewart platform	33300	19740		2772/1692	1836/1296*

3 DOF Shoulder($N_p=6$, $N_a=N_o=N_j=N_L=3$)

6 DOF Stewart Platform($N_p=30$, $N_a=N_o=N_j=N_L=6$)

* Algorithm Parallelism in this method allows synchronous parallel computation with suitable hardware. N_L is equal to 1 with respect to actual elapsed time.

** The inversion of matrix with rank n is assumed to require n^3 Multiplications and $\frac{1}{2}n^3$ Additions from [Strang, 85].

*** The computation number shown in this table does not take into account known reduction rules(symmetry, multiplication by 0).

much more dependent on matrix size than is numerical inversion, due to the lengthy trigonometric nature of each matrix element. Also, note that the apparent possibility of algorithmic singularity at a place of no geometric sigularity from the inversion between the open tree structured coordinate set of each leg and the common intermediate coordinate set is deceptive. Although these algorithmic singularities may occur during numerical evaluation, they do not occur during symbolic evaluation(This was proved from comparison with the result of position constraint differentiation method ; the final denominator is the same).

When closed-chain systems form spatially complicated architectures, symbolic manipulation may not be forthcoming, and, even though possible, it is often the case that numerical evaluation is more suitable(efficient) than symbolic evaluation. As discussed before, the second and third approaches were shown to satisfy the need to evaluate $[G_a^\phi]$ and $([\dot{G}_a^\phi]\dot{\phi}_a)$ without numerical differentiation which can be a main source of simulation error.

Next, consider constraint violation which often arises in multibody motion simulation. When the system equations are described in terms of more coordinate than the minimum required for the closed-chain system, integration procedure involve both velocity and position constraint stabilization method(Baumgrrte, 1972), by the determination of the dependent joint velocities

Table 2 Matrix inversions required for the evaluation of $[G_d^*]$

	3 DOF shoulder		6 DOF stewart platform	
	Size	Number of inversion	Size	Number of inversion
PCDM & HOCM	6×6	1	30×30	1
ICTM	3×3	4	6×6	7

and displacement from Eqs. (19) and (15) after the integration of a minimum set of equations. This is one of the main reasons for obtaining the dynamic equations of motion in terms of a minimum coordinate set for closed-chain mechanisms. Also since, when a system has a complicated geometric structure, the joint displacement parameters determine procedure(Eq. (15)) involves a numerical routine, it is questionable as to how critical the constraint violation of the second and third approaches is, when Eq. (15) is replaced with the integration of the joint speeds obtained via Eq. (19). Therefore, in some motion simulation applications, we feel that our proposed approaches can justifiably allow the elimination of the difficult task of deriving the position constraint equations(Eq. (15)) of the complicated closed-chain system.

6. Conclusion

The two algorithm evaluating loop constraints are presented for kinematic and dynamic modeling of general closed-chain robotic systems in terms of a system minimum set of coordinates. These procedures are based on higher order kinematic relationships between hypothetically open chain reference coordinates(system Lagrangian coordinate) and a set of independent closed-chain coordinates(system generalized coordinates). Those algorithm with the widley used position constraint differentiation method are investigated together, both symbolically and numerically, and discussed with respect to their relative computational merits. Finally, the algorithm s presented herein have been implemented in a general dynamic modeling package entitled "Manipulator Analysis Package(MAP)"(Kang and Freeman, 1990-a). We are also investigating computational efficiency of our constraint embedding approaches with respect to different general classes of mechanical architectures.

References

Asada, H. and Youcef-Toumi, K., 1983, "Analysis and Design of Semi-direct-drive Robot Arms," in Proc. 1983 American Control Conf., June 22-24, San Francisco, CA, pp. 757~764.

Baumgarte, J., 1972, "Stabilization of Constraint and Integrals of Motion," Comp. Meth. Appl. Mech. Eng, Vol. 1, pp. 1~16.

Cox, D. J., 1981, "The Dynamic Modeling and Command Signal Formulation for Parallel Multi -Parameter Robotic Devices," Masters Thesis, Univ. of Florida.

Freeman, R. A. and Tesar, D., 1988, "Dynamic Modeling of Serial and Parallel Mechanisms Robotic Systems, Part I -Methodology, Part II -Applications," Trends and Developments in Mechanisms, Machines, and Robotics, 20th Biennial Mechanisms Conf., Kissimmee, Florida.

Hollerbach, J. M., 1980, "A Recursive Formulation of Manipulator Dynamics and A Comparative Study of Dynamic Formulation Complexity," IEEE Trans. Syst. Man. Cybern., Vol. SMC -10, pp. 730~736.

Kang, H. J. and Freeman, R. A., 1990-a, "An Interactive Software Package(MAP) for the Dynamic Modeling and Simulation of Parallel Robotic System Including Redundancy," ASME Comp. Eng. Conf., Aug., Vol. 3, pp. 1142 ~1149.

Kang, H. J. and Freeman, R. A., 1990-b, "Evaluation on Loop Constraint of General Closed-Chain Linkages," 2ndInt. Workshop on Advances in Robot Kinematic, Linz, Austria.

Luh, J. Y. S., Walker, M. W. and Paul, R. P., 1980, "On Line Computational Scheme for Mechanical Manipulators," J. Dyn. Syst. Meas. Con., Vol. 102, No. 4, pp. 69~76.

Luh, J. Y. S. and Zheng, Y. F., 1985, "Computation of Input Generalized Forces for Robots

with Closed Kinematic Chain Mechanisms," IEEE J. Robotic and Automation, Vol. RA-1, No. 2, pp. 95~103.

Murray, J. J. and Lovell, G. H., 1989, "Dynamic Modeling of Closed-Chain Robotic Manipulators and Implications for Trajectory Control," IEEE Int. Conf. Robotics and Automation, pp. 1354~1360.

Paul, B., 1975, "Analytical Dynamics of Mechanisms-A Computer Oriented View," Mechanism and Machine Theory, Vol 10, pp. 481~507.

Walker,M. W. and Orin, D. E, 1982, "Efficient Dynamic Computer Simulation of Robotic Mechanism," ASME J. Dyn. Syst. Meas. Cont., Vol. 104, No. 3., pp. 205~211.

Stewart, D., 1965, "A Platform with Six Degrees of Freedom," Proc. Institution of Mechanical Engineers, Vol. 180, Part 1, No. 15, pp. 371~386.

Walker, M. W., 1985, "A Unified Approach to Manipulator Modeling," IEEE Conf. on Robotics and Automation, Vol. 2, pp. 729~736.

Wittenburg, J., 1977, *Dynamic of System of Rigid Bodies*, Stuttgart, FRG : B. G. Teubner.

Wittenburg, J., 1983, "Analytical Method in Mechanical System Dynamic," NATO ASI Series, Series F ; Computer and System Science, Vol. 9, pp. 89~127.

Appendix 1

Consider a(completely) closed-chain manipulator system with total number of joints, bodies and branches given by N_{tj}, N_{tb} and N_L respectively. They must satisfy the relation

$$N_{tj}-N_{tb}=N_L-2 \tag{A.1}$$

This is the case for most of parallel mechanisms and multiple robots manipulating a common object. To prove $N_p=N_o\times(N_{tb}-1)$ in Eq. (27), the system mobility, M, is employed as

$$\begin{aligned} M&=N_a=N_{tj}-N_p \\ &=N_o\times(N_{tb}-1)-N_{tj}\times(N_o-1) \end{aligned} \tag{A.2}$$

Solving Eq. (A.2) for N_p, gives, when combined with Eq. (A.1),

$$N_p=N_o\times(N_{tj}+1N_{tb})=N_o\times(N_L-1) \tag{A.3}$$