공학 연구논문집 제24권 제2호 pp. 95~108, 울산대학교
Journal of Engineering Research Vol. 24, No. 2, pp. 95~108, 1993.
University of Ulsan.

로보트 매니퓰레이터의 KIC-Based 동력학적 모델링의 개선 방안 연구

강희준
제어계측공학과

〈요 약〉

본 논문은 기존의 KICs(Kinematic Influence Coefficients)에 근거한 로보트 매니퓰레이터의 동력학적 모델링 방법과 그 적용의 장점을 소개하고, 이 방법의 단점인 계산의 비효율성을 극복하기 위한 개선 방안을 제시한다. 제시된 방법은 Generalized Augmented Body 개념을 기초로 하고, 기존의 방법에 비교하여 동력학적 모델을 구하는데 필요한 계산의 갯수(Computational Counts)의 44% 감소를 이루었다.

Computationally Efficient KIC-Based Dynamic Modeling of Robot Manipulators

Hee-Jun Kang
Department of Control and Instrumentation Engineering

<Abstract>

In this paper, the current KIC(Kinematic Influence Coefficients) based dynamic modeling algorithm of robot manipulators is modified to overcome its computational inefficiency. The modification is made by the introduction of the generalized body concept. This modified algorithm allows a 44% computational saving over the current optimized method.

1. Introduction

Three main methodologies, categorized as the Recursive Newton-Euler method[1-2], the Lagrange-Euler method[3-4] and Lagrange' s form of the Generalized Principle of d' Alembert [5-7] have been extensively investigated for the dynamic modeling of robots. All these efforts have contributed to the progress in dynamics for robot control and design. When an algorithm is evaluated, it can be discussed with regard to efficiency and efficacy: efficiency of kinematic and dynamic transformations from the standpoint of real-time control, and efficacy in terms of improvement in control and design. From this point of view, the Recursive Newton-Euler Method might be said to have more computational efficiency and less efficacy for inverse dynamics than the other two methods. Because this algorithm results in fundamental information only(mainly joint torques corresponding to the manipulator dynamics), it is difficult to apply it to sophisticated robot manipulator control systems which utilize explicit dynamic models. Moreover, this algorithm must be modified to generate the required dynamic models for forward dynamic simulation, which is indispensable to design optimization of robotic systems.

The particular algorithm[8] which will be the basis forlater development, has good efficacy rather than computational efficiency, even though its real time computation capability has been proved through parallel pipelined programing running on an AP 500 Array Processor[9]. The algorithm generates the necessary modeling parameters useful for the improvement of both control and design and requires more arithmetic operations as a trade-off between efficacy and efficiency. This paper proposes the modification of this particular algorithm to increase the computational efficiency without sacrificing the efficacy provided by the resulting explicit dynamic models. Sections 2. reviews this particular algorithm. Section 3. shows the mathematical and physical properties of the resulting dynamic models and determines the number of the independent model elements for efficient evaluation of the complete dynamic model. Finally, Section 4. proposes a modified algorithm which only generates the independent modeling elements and then constructs the full dynamic model. This is accomplished by considering the properties of the resulting dynamic model and by the introduction of generalized augmented body concept. The partial momenta of the generalized augmented body due only to its corresponding joint velocity are considered. The time variation effects of the momenta result in the required independent dynamic model elements. The computational efficiency of this modified algorithm is estimated as O (N^3), with 3362 arithmetic operations when N=6. This algorithm allows a 44 % computational saving over the current optimized one (O(N^4), 5995 when N=6)[10].

2. Current KIC Based Modeling Method

2.1 Kinematic Influence Coefficients (KICs)

The main idea of Kinematic Influence Coefficients(KICs) is placed on the separation of time dependent and position dependent functions. The purely position dependent KICs will be the fundamental tools to lead closed-form dynamic models, which are in turn purely position dependent (actually, time independent). The first order and second order KICs will be shown to represent velocities and accelerations of a moving robot. The velocity of the end-effector of robot can generally be obtained by direct differentiation of its position and orientation functions with respect to time. However, the direct differentiation of the finite orientation angles with respect to time is problematic due to the spatial geometry of a robot. This is because the rate of change of the joint angles do not represent the finite orientation angles by integration over time. Therefore, classical vector mechanics is utilized to represent the velocity of the considered body by adding its partial velocities[11].

Here, only the result format of the higher-order kinematics for a general serial manipulator is given by using KICs. For the velocity of a vector of p dependent (output) parameters in terms of a set of m independent input coordinates $\ddot{\phi}$, one has

$$\underline{\dot{u}} = [G_{\phi}^{u}]\,\underline{\dot{\phi}} \qquad (1)$$

Here,

$$[G_{\phi}^{u}] = [\frac{\partial u}{\partial \phi_1}\ \frac{\partial u}{\partial \phi_2} \cdots \frac{\partial u}{\partial \phi_m}] = [g_1^u g_2^u \cdots\cdots g_m^u] \qquad (2)$$

is the first order KICs relating the coordinates u and ϕ with the n[th] column g_u being of dimension p x 1. Having stated the first-order kinematics in a fairly common form as Jacobian, the second-order kinematics are presented in a less common form. Here, a particular matrix formulation is chosen in which the non-linear, velocity related components are expressed in terms of a three-dimensional coefficient array $[H_{\phi\phi}^{u}]$, (consisting of position dependent second-order partial derivatives) operated on quadratically in a "plane by plane" sense. Generally, the acceleration vector ü of a set of p dependent parameters u is represented in terms of the m independent coordinates ϕ as

$$\ddot{u} = [G_{\phi}^{u}]\,\underline{\ddot{\phi}} + \underline{\dot{\phi}}^{T}\,[H_{\phi\phi}^{u}]\,\underline{\dot{\phi}} \qquad (3)$$

where the second-order influence coefficient array $[H_{\phi\phi}^{u}]$ with the dimension of p x m x m is defined as

$$([\dot{G}_{\phi}^{u}])\,\underline{\dot{\phi}} = \underline{\dot{\phi}}^{T}\,[H_{\phi\phi}^{u}]\,\underline{\dot{\phi}} = \begin{bmatrix} \underline{\dot{\phi}}^{T}\ [{}^{1}H_{\phi\phi}^{u}]\ \underline{\dot{\phi}} \\ \underline{\dot{\phi}}^{T}\ [{}^{2}H_{\phi\phi}^{u}]\ \underline{\dot{\phi}} \\ \underline{\dot{\phi}}^{T}\ [{}^{p}H_{\phi\phi}^{u}]\ \underline{\dot{\phi}} \end{bmatrix} \qquad (4)$$

In representing the centripetal, and Coriolis acceleration of the link, the second order KICs seem to be computationally redundant in the sense of the acceleration evaluation. However, the sacrifice in computational efficiency is compensated for with improved efficacy in several applications. For example, this matrix is essential in explaining the antagonistic stiffness effect which occurs in redundant actuation involving structures with nonlinear geometry[12] and their applications[13]. The matrix is also utilized in design optimization [14] and in the formulation of an integrable pseudoinverse for redundant manipulator control[15], etc. See Appendix 1 for specifics of KIC parameters.

2.2 Dynamics of Serial Manipulators

The dynamic controlling equations for general serial manipulators will now be given using the generalized principle of d' Alembert (i.e., the virtual work of the d' Alembert loads) to transfer the system dependence from the specified 6m Cartesian based local link coordinates to the m generalized parameters $\underline{\phi}$. The inertia force $\underline{f}^k$ due to the centroidal acceleration ${}^k\underline{a}^c$ of the mass M_k of link k can be expressed by using Newton' s equations as

$$\underline{f}^k = M_k\, {}^k\underline{a}^c \tag{5}$$

The inertia moment m^k can be expressed in a modified Euler format (the benefit of this will become apparent momentarily) as

$$\underline{m}^k = [II^k]\,\underline{\alpha}^k + \underline{\omega}^k \times [II^k]\,\underline{\omega}^k \tag{6}$$

where $[II^k] = [I^{kx} : I^{ky} : I^{kz}]$; 3 x 3 globally referenced inertia matrix.

From Eqs.(5) and (6) and the principle of virtual work, the generalized inertial loads $\underline{T}^I_\phi$ of an m-link chain as referenced to the m relative joint parameters (ϕ) are given by

$$\underline{T}^I_\phi = \sum_{k=1}^{m} ([{}^kG^c_\phi]^T \underline{f}^k + [G^k_\phi]^T \underline{m}^k) \tag{7}$$

with $[{}^kG^c_\phi]$ and $[G^k_\phi]$ the associated centroidal and rotational first order KICs of link jk, respectively(See Appendix 1). In order to maintain the particular form of Eqs.(3) and (6), the first order KIC matrix transposes in Eq.(7) must be brought inside the quadratic forms of Eqs.(5) and (6). This is accomplished by using generalized scalar product (o) [6]. Now, with the generalized scalar product (o) and the influence coefficient representation of the link kinematics, the driving inertia torques become

$$\underline{T}^I_\phi = [I^*_{\phi\phi}]\ddot{\underline{\phi}} + \dot{\underline{\phi}}^T [P^*_{\phi\phi\phi}]\dot{\underline{\phi}} \tag{8}$$

where the mxm joint-referenced effective inertia matrix is

$$[I^*_{\phi\phi}] = \sum_{k=1}^{m} \{M_k [{}^kG^c_\phi]^T [{}^kG^c_\phi] + [G^k_\phi]^T [II^k] [G^k_\phi]\} \tag{9}$$

and the mxmxm inertial power array (operated on in the same "plane by plane" manner as in Eqs.(3) and (4) for parameter accelerations (ü)) is

$$[P^*_{\phi\phi\phi*}] = \sum_{k=1}^{n} \{M_k([{}^kG^c_\phi]^T \circ [{}^kH^c_{\phi\phi}] + (([G^k_\phi]^T [II^k]) \circ [H^k_{\phi\phi}]) + ([G^k_\phi]^T \times [G^k_\phi])^T [II^k] [G^k_\phi]\} \tag{10}$$

This concludes the modeling of serial manipulators. For a more general treatment of dynamic modeling, including external loads, gravity, springs, and viscous friction, see [11].

3. Properties of the Dynamic Models

The dynamic models, seen in Eqs. (9) and (10), exhibit some useful mathematical properties which can be utilized to reduce computational complexity and also illustrate physical meaning. Recalling their evaluation equations, these properties are listed as follows:

Effective Inertia Matrix ($[I^*]$)

. Positive definite (P1)

. $[I^*]_{j;k} = [I^*]_{k;j}$ (symmetric) (P2)

First Term of Effective Inertia Power Matrix ($[\overset{*}{P_1}]$)

. $[\overset{*}{P_1}]_{i;j;k} = [\overset{*}{P_1}]_{i;k;j}$ (symmetric in the i^{th} plane) (P3)

. $[\overset{*}{P_1}]_{i;j;k} = -[\overset{*}{P_1}]_{j;k;i}$ for $j \le k < i$ (P4)

. $[\overset{*}{P_1}]_{i;j;k} = 0$ for ($i = k$ and $j \le i$) or ($i = j$ and $k \le i$) (P5)

Second Term of Effective Inertia Power Matrix ($[\overset{*}{P_2}]$)

. $[\overset{*}{P_2}]_{i;j;k} = 0$ for $j \le k$ (P6)

Third Term of Equivalent Inertia Power Matrix ($[\overset{*}{P_3}]$)

. $[\overset{*}{P_3}]_{i;j;k} = -[\overset{*}{P_3}]_{j;i;k}$ (skew symmetric in the k^{th} plane) (P7)

. $[\overset{*}{P_3}]_{i;j;k} = 0$ for $i = j$. (P8)

Dependency between $[\overset{*}{P_2}]$ and $[\overset{*}{P_3}]$

. $[\overset{*}{P_3}]_{i;j;k} = -[\overset{*}{P_2}]_{k;i;j}$ for $k < j$ and $i < j$ (P9)

. $[\overset{*}{P_3}]_{i;j;k} = -[\overset{*}{P_2}]_{k;i;j}$ for $j < k < i$ (P10)

The mathematical properties of the dynamic models might allow the considerable computational saving, if the dynamic algorithm is able to fully utilize them. For example, the number of model parameters of the inertia matrix($[I^*]$) and the first term of the inertia power matrix ($[\overset{*}{P_1}]$) are n^2 and n^3, respectively, for an n DOF serial manipulator. Full utilization of the properties of $[I^*]$ and $[\overset{*}{P_1}]$ reduces those numbers to $\frac{1}{2}(n^2 + n)$ for $[I^*]$ and $\frac{1}{3}(n^3 - n)$ for $[\overset{*}{P_1}]$, respectively. When $n = 6$, the numbers are such that 36 -> 21 for $[I^*]$, and 216 -> 70 for $[\overset{*}{P_1}]$. The reduced numbers can be the minimum number of model elements to be evaluated for $[I^*]$ and $[\overset{*}{P_1}]$. However, these elements might not fully represent the dynamic interactions between the joints, occurring due to joint motions of the manipulator, since the above counts don' t include the zero dynamic interactions of Property (P5).

Knowing the number of necessary model elements which fully describe the dynamic interactions between the joints of the manipulator give more insight into the physical nature of manipulator dynamics. Including the model elements resulting in the zero net dynamic interactions, which are consistent with Properties (P3) and (P4), yields the number $\frac{1}{6}(2n^3 + 3n^2 + n)$ for $[\overset{*}{P_1}]$. When $n = 6$, the number becomes 216 -> 91 for $[\overset{*}{P_1}]$. Under the same consideration, the model elements of $[\overset{*}{P_2}]$ and $[\overset{*}{P_2}]$ were counted. The number of model elements are also $\frac{1}{6}$

$(2n^3 + 3n^2 + n)$ for both model parameters. Therefore, the computation count numbers are

. $\frac{1}{2}(n^2+n)$ for $[I^*]$

. $\frac{1}{6}(2n^3+3n^2+n)$ each, for $[P_1^*]$, $[P_2^*]$ and $[P_3^*]$

The independent dynamic model elements are defined as the necessary model elements to fully represent the dynamic interactions between the joints, due to the joint motions of the manipulator. The arrangement of the independent model elements for $[I^*]$ and $[P^*]$ can be graphically viewed in Figs. 1 and 2, respectively. Existence of the independent dynamic model elements for $[I^*]$ and $[P^*]$ will aid in modifying the current dynamic algorithm to increase computational efficiency. The following sections focus on efficient generation of the independent dynamic model elements by considering the generalized augmented body.

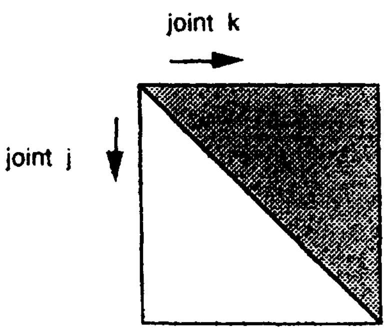

Fig. 1 The Independent Modeling Elements for $[I^*]$

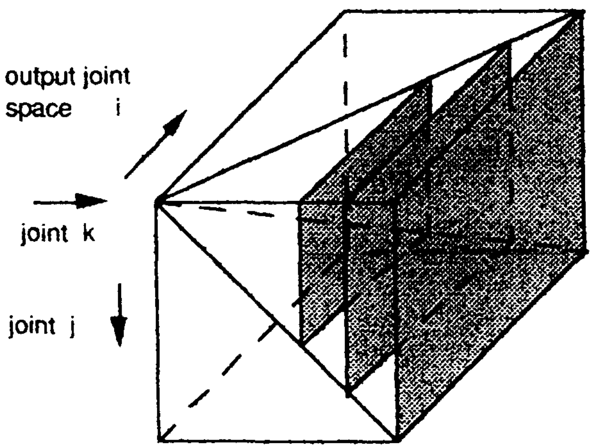

Fig. 2 The Indepent Modeling Elements for each for $[P_1^*]$, $[P_2^*]$ and $[P_3^*]$

4. Computationally Efficient Modified Dynamic Algorithm

4.1 General Description of the Modified Dynamic Algorithm

In section 3, the resulting dynamic models is shown to be constructed in terms of their independent elements, which represent the physical dynamic interactions between joints due to the joint motions of the manipulator. In order to find the actual dynamic quantities (mass and inertia tensor) related to each independent model element, the system momenta are considered.

Employing the first order translational KIC matrix, the total linear momentum of the manipulator is

$$\underline{P} = \sum_{k=1}^{n} m_k {}^k\underline{V}^c = \sum_{k=1}^{n} m_k [{}^kG_\phi^c] \dot{\underline{\phi}} \qquad (11)$$

Expanding this equation, in terms of the column elements of the KIC matrix, yields

$$\underline{P} = m_1 g_1^{c1} \dot{\phi}_1 + m_2 (g_1^{c2} \dot{\phi}_1 + g_2^{c2} \dot{\phi}_2) + \ldots\ldots + m_n (\sum_{j=1}^{n} g_j^{cn} \dot{\phi}_j) \qquad (12)$$

By selecting the coefficients corresponding to each joint velocity, Eq. (12) becomes

$$\underline{P} = (\sum_{j=1}^{n} m_j g_1^{cj}) \dot{\phi}_1 + (\sum_{j=2}^{n} m_j g_2^{cj}) \dot{\phi}_2 + \ldots\ldots + m_n g_n^{cn} \dot{\phi}_n \qquad (13)$$

Assuming that joint k is a revolute joint, the partial linear momentum, $\underline{P}^k$, due to k[th] joint velocity can be written as

$$\underline{P}^k = (\sum_{j=k}^{n} m_j g_k^{cj}) \dot{\phi}_k = S_k \times (\sum_{j=k}^{n} m_j \underline{r}_k^{cj}) \dot{\phi}_k = M_k (S_k \times \underline{r}_k^{c*}) \dot{\phi}_k = M_k g_k^{c*} \dot{\phi}_i \qquad (14)$$

where

$$M_k = \sum_{j=k}^{n} m_j, \ \underline{r}_k^{c*} = \frac{1}{M_k} (\sum_{j=k}^{n} m_j \underline{r}_k^{cj}) \text{ and } g_k^{c*} = S_k \times \underline{r}_k^{c*} \qquad (15)$$

Therefore, the total linear momentum of the manipulator can be expressed as

$$\underline{P} = \sum_{k=1}^{n} \underline{P}^k = \sum_{k=1}^{n} M_k g_k^{c*} \dot{\phi}_k \qquad (16)$$

Considering that the inertia matrix is origin dependent and then applying the parallel axes theorem, the same approach for total angular momentum of the manipulator yields

$$\underline{L} = \sum_{k=1}^{n} \underline{L}^k = \sum_{k=1}^{n} [III^k] g_k \dot{\phi}_k \qquad (17)$$

where

$$[III^k] = \sum_{j=k}^{n} \{[II^j] + m_k([I] \underline{r}_j^T \underline{r}_j - \underline{r}_j \underline{r}_j^T)\} \qquad (18)$$

with [I] being 3 by 3 identity matrix.

Reviewing Eqs. (15) and (18), the total momenta of the manipulator can be expressed as the sum of the momenta of the generalized augmented bodies due to only its own joint velocity. Following Eqs. (15) and (18), the generalized augmented bodies are constructed for each link from manipulator tip to mani pulator base. For more understanding, the k[th] generalized augmented body is graphically shown in Fig.3.

Practically, it is impossible to directly generate the complete dynamic model([I*] and [P*]) by considering the time rate of change of each partial momenta of the generalized augmented bodies, since the partial momenta are the composition of the different momentum vectors described in different moving coordinate frames. Considering all these effects, the description of the system momenta, seen in Eqs.(16) and (17), would be back to that of the system momenta based on each link of the manipulator. However, the main purpose

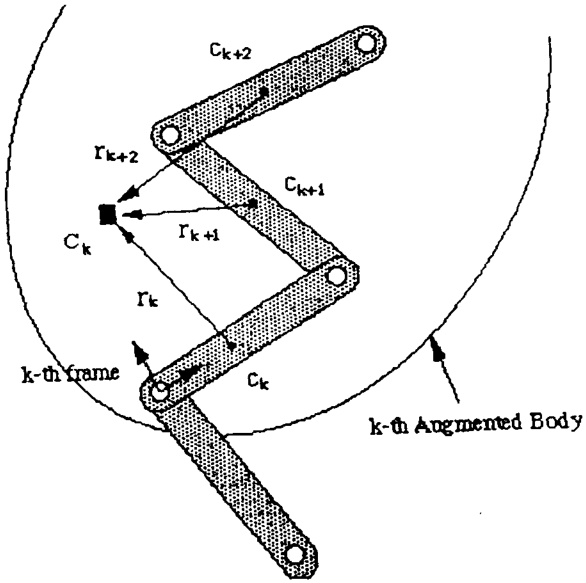

Fig. 3 The K^{th} Generalized Augmented Body of the Manipulator

of this section is the generation of the independent model elements and then the construction of the full dynamic model by using their inherent properties (P1)-(P10). Assuming that the joints inside the generalized augmented body are fixed, consider how time variation of the partial momenta affect the downstream joints of the manipulator. Recalling Figs. 2 and 3, the time variations of the nth partial momenta leads to the generation of the independent model elements in the last column of [I$^{\cdot}$] and the last plane of [P$^{\cdot}$]. This is easily checked using the previous algorithm, because the effect of the generalized augmented body are not accounted for there. Then, one can think of the next augmented body as the last link of (n-1) DOF manipulator. The consideration result in the independent model elements on the (n-1)th column and plane of the dynamic models. This process, preceding from the distal link to the proximal link of the manipulator, can generate all the independent dynamic model elements required to construct the complete dynamic model([I$^{\cdot}$] and [P$^{\cdot}$]).

The time rate of change of the momenta in Eqs. (14) and (17) can be written as

$$\dot{\underline{P}}^{k} = M_k g_k^{c^*} \ddot{\phi}_k + \underline{\omega}^k \times M_k g_k^{c^*} \dot{\phi}_k \tag{19}$$

and

$$\begin{aligned} \dot{\underline{L}}^{k} &= [III^k] \frac{d}{dt}(g_k \dot{\phi}_k) + \underline{\omega}^k \times ([III^k] g_k \dot{\phi}_k) \\ &= [III^k] g_k \ddot{\phi}_k + [III^k] (\underline{\omega}^k \times g_k) \dot{\phi}_k + \underline{\omega}^k \times ([III^k] \\ &\quad g_k \dot{\phi}_k) \end{aligned} \tag{20}$$

Employing the generalized principle of d' Alembert, as in Eq.(2-33), the effective load due to the k^{th} augmented body can be expressed as

$$\underline{T}^I_k = [{}^kG^{c*}_\phi]^T \dot{\underline{P}}^k + [G^k_\phi]^T \dot{\underline{L}}^k$$
$$= (M_k[{}^kG^{c*}_\phi]^T g^{c*}_k + [G^k_\phi]^T [III^k] g_k)\ddot{\phi}_k + M_k[{}^kG^{c*}_\phi]^T(\underline{\omega}^k x\, g^{c*}_k)\dot{\phi}_k$$
$$+ [III^k][G^k_\phi]^T (\underline{\omega}^k x\, g^{c*}_k)\dot{\phi}_k + [G^k_\phi]^T(\underline{\omega}^k x\, ([III^k] g_k \dot{\phi}_k)) \quad (21)$$

Recalling the definition of the second order KICs, this equation becomes

$$\underline{T}^I_k = (M_k[{}^kG^{c*}_\phi]^T g^{c*}_k + [G^k_\phi]^T [III^k] g_k)\ddot{\phi}_k + \dot{\underline{\phi}}^T (M_k[G^k_\phi]^T [H^{c*}_{\phi\phi}]_{;;k})\dot{\phi}_k$$
$$+ \dot{\underline{\phi}}^T([G^k_\phi]^T [III^k][H^k_{\phi\phi}]_{;;k})\dot{\phi}_k + \dot{\underline{\phi}}^T([G^k_\phi]^T x\, [G^k_\phi])^T [III^k] g_k)\dot{\phi}_k \quad (22)$$

The model elements in the above equation represent the k^{th} column and plane of the independent modeling parameters, seen in Figs 1 and 2, of $[I^*]$ and $[P^*]$, respectively. The evaluation forms of the independent model elements are summarized as

$$[I^*_1]_{i;k} = M_k (g^{c*}_k)^T g^{c*}_k \qquad \text{for } i \le k \quad (23)$$

$$[I^*_2]_{i;k} = g^T_j [III^k] g_k \qquad \text{for } i \le k \quad (24)$$

$$[P^*_1]_{i;j;k} = M_k (g^{c*}_1)^T [H^{c*}_{\phi\phi}]_{;j;k} \qquad \text{for } i, j \le k \quad (25)$$

$$[P^*_2]_{i;j;k} = g^T_1 [III^k][H^k_{\phi\phi}]_{;j;k} \qquad \text{for } i, j \le k \quad (26)$$

$$[P^*_3]_{i;j;k} = [[H^k_{\phi\phi}] - [H^k_{\phi\phi}]^T]_{;i;j} [III^k] g_k \qquad \text{for } i, j \le k \quad (27)$$

The rest of the model elements for $[I^*]$ and $[P^*]$ are evaluated by utilizing the mathematical properties (P2)-(P10), seen in Section 3. One thing to note is that the numerical values of the dynamic model components, $[I^*_1]$, $[I^*_2]$, $[P^*_1]$, $[P^*_2]$ and $[P^*_3]$ matrices, obtained from the modified algorithm, are different from those obtained from the previous algorithm, while the complete dynamic model($[I^*]$ and $[P^*]$ matrices) are identical. This is mainly due to the transfer of each link' n mass quantities into the inertia quantities of the generalized augmented body.

Before describing the overall procedure and actual commutational count of this new algorithm, comparison of the new evaluation equations with Eqs. (9) and (10) gives an overall view of the computational saving afforded by the augmented body algorithm. The previous algorithm requires full generation of the first order and second order translational KICs of the center of mass of each link, with the final model elements then obtained by addition of the effective model elements of each link. This addition is one of major contributors to the computational count of the old algorithm. The modified algorithm requires only vector and plane generation of the first order and second order translational KICs, respectively, of the center of mass of the corresponding generalized augmented body. The use of the generalized augmented body eliminates the addition of the effective model elements of each link. However, the modified algorithm requires the generation of mass, location of mass center and inertia matrix of each generalized augmented body. Their evaluation, seen in Eqs. (15) and (18), can be obtained in recursive evaluation form as

$$M_k = m_k + M_{k+1},\ \underline{r}_k^{c*} = \frac{1}{M_k}(\underline{r}_k^{ck} + \underline{r}_{k+1}^{c*} + \underline{O}_k^{k+1}) \tag{28}$$

and

$$\begin{aligned}[III^k] = [II^k] &+ m_k\{[I](\underline{r}_k^{c*} - \underline{r}_k^{ck})^T(\underline{r}_k^{c*} - \underline{r}_k^{ck}) - (\underline{r}_k^{c*} - \underline{r}_k^{ck})(\underline{r}_k^{c*} - \underline{r}_k^{ck})^T\} \\ &+ [III^{k+1}] + M_{k+1}\{[I](\underline{r}_k^{c*} - \underline{r}_{k+1}^{c*} - \underline{O}_k^{k+1})^T(\underline{r}_k^{c*} - \underline{r}_{k+1}^{c*} - \underline{O}_k^{k+1}) \\ &- (\underline{r}_k^{c*} - \underline{r}_{k+1}^{c*} - \underline{O}_k^{k+1})(\underline{r}_k^{c*} - \underline{r}_{k+1}^{c*} - \underline{O}_k^{k+1})^T\}\end{aligned} \tag{29}$$

where the notation employed is made clear with the help of Fig. 4. Hence, considerable saving in computation is expected based on this overall argument without a detailed accounting.

4.2 Computational Count of the Modified Algorithm

In this subsection, the steps and their computational requirements of the modified algorithm are described for computer implementation. For additional saving, all necessary model components are described with respect to the first joint coordinate frame. For general n revolute serial manipulator, the procedure for generation of the [I*] and [P*] matrices are given as follows:

STEP 1) Compute the rotational matrices, R^1, R^2, ..., R^n, the origin of each joint, $\underline{O}^1$, ..., $\underline{O}^n$, and the location of mass center of each link, $\underline{C}^1$, ..., $\underline{C}^n$.

Multiplications(M) : 51n-87 Additions(A) : 39n-66

STEP 2) Compute the location of center of mass and inertia elements of the generalized augmented bodies, $\underline{r}_k^{c*}$, Mk and [III^k] using Eqs.(28) and (29).

M : 84n-84 A : 70n-70

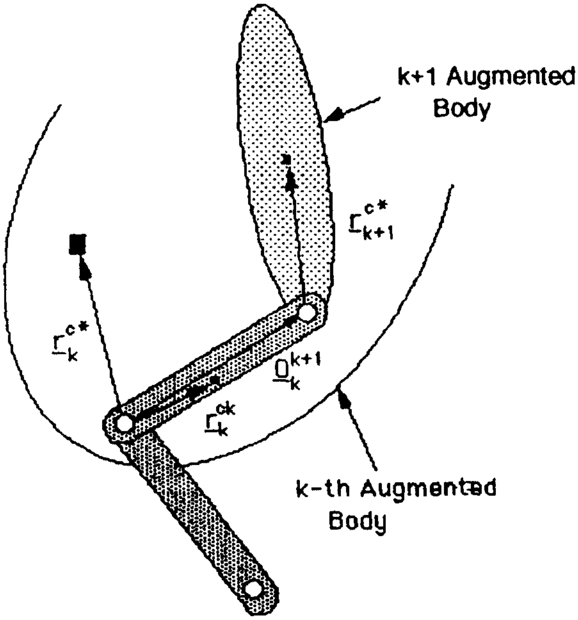

Fig. 4 The Generalized Augmented Body for the Recursive Evaluation

STEP 3) Compute the first order KICs, g_k^{c*} and g_k.

M : $3n^2 - 3n$ A : $2n^2 + 4n$

STEP 4) Compute the $[I^*]$ matrix using Eq. (22).

M : $\frac{7}{2}n^2 + \frac{25}{2}n$ A : $\frac{5}{2}n^2 + \frac{17}{2}n$

The number of computation for the independent modeling elements of the $[I^*]$ matrix

M : $\frac{13}{2}n^2 + \frac{289}{2}n - 171$ A : $\frac{9}{2}n^2 + \frac{243}{2}n - 136$

STEP 5) Compute the second order KICs, $[H^{c*}_{\phi\phi;;k}]$ and $[H^{c*}_{\phi\phi;;k}]$.

M : $6n^2 - 12n$ A : $3n^2 - 3n - 2$

STEP 6) Compute the independent modeling elements of $[P_1^*]$, $[P_2^*]$ and $[P_3^*]$.

M : $3n^3 - \frac{3}{2}n2 - \frac{7}{2}n$ A : $2n^3 - 3n$

STEP 7) Generate [P] from the independent modeling elements.

M : 0 A : $\frac{11}{6}n^3 - n^2 + \frac{2}{3}n$

The number of computation for $[I^*]$ and $[P^*]$

M : $3n^3 + 11n^2 + 129n - 171$ (1647 for n=6)

A : $\frac{23}{6}n^3 + \frac{13}{2}n^2 + \frac{691}{6}n - 138$ (1715 for n=6)

5. Conclusion

In this paper, the modification of the particular algorithm[8] has been performed by utilizing the properties of the dynamic models and the generalized augmented body. For general 6 dof serial manipulators, the resulting algorithm($O(n^3)$) has 1647 multi-plications and 1715 additions. Compared with the optimal computational count of the previous algorithm (5995 total operations for n=6[10]), the new algorithm(3363 total operations for n=6) results in a 44% computational saving and thus is more appropriate for real time computation. This algorithm is also comparable to the most efficient algorithm in the literature($O(n^3)$, 2528 total operations)[4] which evaluates the position dependent explicit dynamic model in the form as $[A]\ddot{\underline{\phi}} + [B]\dot{\underline{\phi}}\dot{\underline{\phi}} + [C]\dot{\underline{\phi}}^2$. The increase in computational count of the proposed algorithm might be due to the generation of the three dimensional modeling parameter($[H^u_{\phi\phi}]$ and $[P^*]$). The practical applications of $[P^*]$ are made for determining optimal actuator size under constant operational velocity of the robot tip[16], while the applications of $[H^u_{\phi\phi}]$ is mentioned before[12-14]. These might not be possible using Li' s algorithm.

Appendix 1 Kinematic Influence Coefficents of Serial Manipulator

Translational First Order KIC		
k^{th} Joint Type	Conditions	Value
Revolute(R)	$k \le n$	$\underline{S}^k \times (\underline{P} - \underline{R}^k)$
Prismatic(P)	$k \le n$	$\underline{S}^k$
R, P	$k > n$	0

Rotational First Order KIC		
k^{th} Joint Type	Conditions	Value
R	$k \le n$	$\underline{S}^k$
P	$k \le n$	0
R, P	$k > n$	0

	Translational Second Order KIC		
k^{th} Joint Type	k^{th} Joint Type	Conditions	Value
R	R	$j \le k \le n$	$\underline{S}^j \times (\underline{S}^k \times (\underline{P} - \underline{O}^k))$
R	R	$k \le j \le n$	$\underline{S}^k \times (\underline{S}^j \times (\underline{P} - \underline{O}^j))$
P	R	$j \le k \le n$	$\underline{S}^j \times \underline{S}^k$
R	P	$k \le j \le n$	$\underline{S}^k \times \underline{S}^j$
P	R	$j \le k \le n$	$\underline{0}$
R	P	$k \le j \le n$	$\underline{0}$
R, P	R, P	(j or k) > n	$\underline{0}$
P	P	All j, k	$\underline{0}$

	Rotational First Order KIC		
k^{th} Joint Type	k^{th} Joint Type	Conditions	Value
R	R	$j < k \le n$	$\underline{S}^j \times \underline{S}^k$
R	R	j>k or k>n	$\underline{0}$
P	R	All j, k	$\underline{0}$
R	P	All j, k	$\underline{0}$

where $\underline{S}^j$ and $\underline{O}^j$ is the rotation vector of j^{th} joint and the vector from the origin of base coodinate frame to the origin of the j^{th} coordinate frame, respectively. P is the point of interest in the manipulator.

Reference

1. Luh, J.Y.S., Walker, M.W., and Paul, R.P., "On Line Computational Scheme for Mechanical Manipulators", J. Dyn. Syst., Meas. Cont., Vol. 102, No. 4, pp. 69-76, June, 1990.
2. Walker, M.W. and Orin, D.E., "Efficient Dynamic Computer Simulation of Robotic Mechanisms", ASME J. of Dyn. Syst. Meas. Cont., Vol. 104, No. 3., pp. 205-211, Sept. 1982.
3. Renaud, M., "An Efficient Iterative Analytical Procedure For Obtaining A Robot Manipulator Dynamic Model", Robotics Research: The First International Symposium, MIT Press, pp. 749-764, 1984.
4. Li, Chang-Jin, "A New Lagrangian Formulation of Dynamics for Robot Manipulators", ASME J. of Dynamic Systems, Measurement, and Control, Vol. 111, pp.559-567, Dec., 1989.
5. Cho, W. and Tesar, D., and Freeman, R.A., "The Dynamics and Stiffness Modeling of General Robotic Manipulator Systems with Antagonistic Actuation.", Proc. IEEE Conf. Robotics and Automation, Scottsdale, AZ. Vol. 3, pp. 1380-1387, 1988.
6. Freeman, R.A., "Kinematic and Dynamic Modeling, Analysis and Control of Robotic Systems via Generalized Coordinate Transformations", Dissertation, University of Florida, 1985.
7. Wittenburg, J., "Dynamics of Systems of Rigid Bodies", Stuttgart, FRG: B. G. Teubner, 1977.
8. Thomas, M., and Tesar, D., "Dynamic Modeling of Serial Manipulator Arms", ASME J. of Dyn. Syst., Meas. & Contr., Vol. 104, No. 3, pp. 218-228, 1982.
9. Wander, J.P. and Tesar, D., "Software for Pipelined Computation of Manipulator Modeling Matrices", IEEE J. of Robotics and Auto. Vol.RA-3, No.6, Dec. 1987.
10. Hernandez, E., "Efficient Algorithms and Real-Time Software for Quasi-State Manipulator Deflection Modeling", Master Thesis, The University of Texas at Austin, May, 1989.
11. Kang, H. J., "Dynamic Modeling and Optimal Joint Torque Coordination of Advanced Robotic Systems", Ph.D Dissertation, University of Texas at Austin, 1991.
12. Freeman, R.A., "Feedforward Stiffness Control of Overconstrained Mechanisms/Robotic Linkage System", ASME winter annual meeting, 1988.
13. Yi, B. J., Freeman, R.A. and Tesar, D., "Open-Loop Stiffness Control of Overconstrained Mechanisms/Robotic Linkage Systems", IEEE Conf. Vol. 3, pp.1340-1345, 1989.
14. Bevill, P. J., "Criteria Normalization To Support Decision Making In Intelligent

Machines", Master Thesis, The Unversity of Texas at Austin, May, 1990.

15. Mussa-Ivaldi,F.A. and Hogan, N.,"Solving Kinematic Redundancy with Impedance Control: A Class of Integrable Pseudoinverse", IEEE Conf. Vol. 1, pp. 283-289, 1989.

16. Yuan-Chou, B. C., "Coputer Aided Optimization in the Dynamic Analysis and Parametric Design of Robotic Manipulators", Ph.D Dissertation, University of Texas at Austin, August, 1985.