On the Complexity Analysis of the Coriolis and Centripetal Effects of a 6 DOF Robot Manipulator

A. Y. ZOMAYA and A. S. MORRIS

Department of Control Engineering
University of Sheffield
Mappin Street
Sheffield S1 3JD

Research Report No. 344

October 1988

A.Y. Zomaya, and A.S. Morris
University of Sheffield, U.K.

ABSTRACT:-

The equations used in calculating the different forces and torques which control the movement of a robot manipulator involve a considerable amount of differential and non-linear terms that possess high computational complexity. Centripetal and Coriolis effects are of great importance when the manipulator is moving at high speeds. The previous effects, based on the Lagrangian formulation, have been simplified and a lower order form produced which has reduced computational complexity. Simulation results for a robot arm have been obtained to check for the validity of the derivation.

INTRODUCTION:

Robot arm dynamics and mechanisms deals mainly with the mathematical formulation of the equations of robot manipulator motion, that is, computing the actuators torques and forces to give a lower-pair kinematic chain certain desired trajectories (inverse dynamics), or, given the forces and torques to calculate the accelerations and velocities of the robot arm joints (forward dynamics). The dynamics consists of a set of differential, non-linear, and matrix oriented equations, which describes the behaviour of the robot arm and allows for great flexibility in computer dynamic modelling and simulation studies to evaluate and apply the different control and analysis schemes.

In recent years various techniques and approaches have been developed to formulate robot dynamics. The Lagrangian |1,2,3,4,14| has low computational efficiency with equations of order $O(n^4)$, but otherwise is a well organized and systematic method which give good insight into the application of different control techniques. The generalized D'Alembert formulation |5| has an order of $O(n^3)$ and consists of a fairly well structured set of equations. The Newton-Euler formulation |6,8,9,15| has a messy derivation but is the most efficient formulation with equations of order $O(n)$ that follow from its vector structure and recursive nature. Tabulation-dependent techniques, such as the configuration space method |12|, have very serious difficulties owing to their enormous computer memory requirements.

Other approaches include dynamic equations of Kane |13|, and the use of parallel processing and multi-tasking to reduce the order of computations |16,17,18,19|. Of the previous methods, the most commonly used are the Lagrangian and Newton-Euler. The interaction and equivalence between these has been shown by Silver|11|.

The Lagrangian formulation of the dynamic equations has a simple and algorithmic representation obtained from Lagrangian mechanics. The set of equations consists of second-order, highly-coupled, non-linear differential expressions which can be written in a compact form:

$$F_i = \sum_{k=i}^{n} \sum_{j=1}^{n} \mathrm{Tr}\left\{ \frac{\partial H_o^k}{\partial \nu_i} J_k \left| \frac{\partial H_o^k}{\partial \nu_i} \right|^T \right\} \ddot{v}_j + \sum_{r=i}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \mathrm{Tr}\left\{ \frac{\partial^2 H_o^r}{\partial \nu_j \partial \nu_k} J_r \left| \frac{\partial H_o^r}{\partial \nu_i} \right|^T \right\} \dot{v}_j \dot{v}_k - \sum_{j=1}^{n} m_j \, g \left\{ \frac{\partial H_o^j}{\partial \nu_i} \right\} r_j , \quad i=1,2,\ldots,n \qquad (1)$$

where

- H_o^i: link transformation matrices
- J_r: link inertia matrix
- g : gravitational effects vector
- Tr: trace operator of a matrix ($\mathrm{Tr}(A) = \sum_{i=1}^{m} a_{ii}$).
- F_i: force (prismatic joint) or torque (revolute joint) acting at joint (i).
- $v_i, \dot{v}_i, v_i$,: position, velocity and acceleration of joint (i)
- r_j: centre of mass of link j according to its own coordinates.
- m_j: mass of link j
- n : degree of freedom (DoF).

Eq.(1) was applied to a six DoF robot arm (stanford arm) and the different terms were analyzed and shown to be intertia loading, coupling coriolis and centripetal reactions, and the gravity effects. The inertial and gravity terms are of particular significance in controlling the servo stability and positioning accuracy of the robot arm. The coriolis and centripetal forces are important in high speed movements. All the attempts mentioned previously were made to solve for the dynamic equations with special interest in the coriolis effects, but some of the researchers neglected these second order effects under the assumption of low speed movements, and this assumption led to a suboptimal dynamic performance because of speed restrictions. In this paper, a simplification is described for the mathematical representation, and accordingly for the computational complexity of the coriolis and centripetal forces of a six DoF robot arm.

This is achieved by reducing the order of the lagrangian representation of those forces. Results obtained from simulation programs for the Stanford manipulator are given.

The Lagrangian:

The use of the Lagrangian formulation has the advantage of deriving the mechanics and dynamics of complex systems in a well structured and organised manner but it is very difficult to utilize this in real-time control without simplification. The symbolism and matrix notations used in |1,3| which depend on the Denavit-Hartenberg representation |10| will be used in our discussion. Eq.(1) that controls and governs the motion of the robot arm might be written in an alternate form:

$$F_i = \sum_{j=1}^{n} P_{ij}\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} P_{ijk}\dot{q}_j\dot{q}_k + P_i \qquad (2)$$

$$i = 1,2,\ldots n$$

where

P_{ii}, effective inertia at joint (i)

P_{ij}, coupling inertia between joint (i) and (j)

$$P_{ij} = \sum_{\ell=\max(i,j)}^{n} Tr\left(\frac{\partial H\ell}{\partial q_j} J_\ell \left|\frac{\partial H\ell}{\partial q_i}\right|^T\right) \qquad (3)$$

P_{ijj}, Centripetal forces at joint (i) due to velocity at joint (j).

P_{ijk}, Coriolis forces at joint (i) due to velocities at joint (j) and (k).

$$P_{ijk} = \sum_{\ell=\max(i,j,k)}^{n} Tr\left(\frac{\partial^2 H_\ell}{\partial q_j\, q_k} J_\ell \left|\frac{\partial H\ell}{\partial q_i}\right|^T\right) \qquad (4)$$

P_i, gravity loading vector

$$P_i = \sum_{\ell=1}^{n} -m_\ell\, g^T \left(\frac{\partial H_\ell}{\partial q_i}\right)^\ell r_\ell \qquad (5)$$

and

$$\frac{\partial H_\ell}{\partial q_i} = H_\ell\, {}^\ell\Delta_i \qquad (6)$$

where ${}^\ell\Delta_i$ is the differential translation and rotation transformation matrix if joint ℓ with respect to the i^{th} joint coordinate given by

$$
{}^{\ell}\Delta_i = \begin{vmatrix} 0 & -{}^{\ell}\delta_{iz} & {}^{\ell}\delta_{iy} & {}^{\ell}d_{ix} \\ {}^{\ell}\delta_{iz} & 0 & -{}^{\ell}\delta_{ix} & {}^{\ell}d_{iy} \\ -{}^{\ell}\delta_{iy} & {}^{\ell}\delta_{ix} & 0 & {}^{\ell}d_{iz} \\ 0 & 0 & 0 & 0 \end{vmatrix} \tag{7}
$$

and J_i is a pseudo inertia matrix, where the elements composing the matrix are the moments of inertia, cross product of inertia and first moments of each link, i.e.

$$
J_i = \begin{vmatrix} \frac{-I_{xxi}+I_{yyi}+I_{zzi}}{2} & I_{xyi} & I_{xzi} & m_{ixi} \\ I_{xyi} & \frac{I_{xxi}-I_{yyi}+I_{zzi}}{2} & I_{yzi} & m_{iyi} \\ I_{zyi} & I_{yzi} & \frac{I_{xxi}+I_{yyi}-I_{zzi}}{2} & m_{izi} \\ m_{ixi} & m_{iyi} & m_{izi} & m_i \end{vmatrix} \tag{8}
$$

Inertial and gravity terms:

The effective and coupling inertia terms at eq.(3) have been shown in |1| to be:

$$
P_{ij} = \sum_{\ell=\max i,j}^{n} Tr({}^{\ell}\Delta_j \; J_{\ell} \; {}^{\ell}\Delta_i^T) \tag{9}
$$

and the gravity loading vector are given as:

$$
P_i = {}^{i-1}g \sum_{\ell=i}^{n} m_{\ell} \; {}^{i-1}r_{\ell} \tag{10}
$$

where

$$
{}^{i-1}g = \begin{cases} \{-g.o \quad g.n \quad 0 \quad 0\} & \text{rotational joint} \\ \{\,0 \quad 0 \quad 0 \quad -g.a\,\} & \text{prismatic joint} \end{cases}
$$

The symmetry of the matrix P_{ij} was also shown in |1,3| which led to $p_{ij}=p_{ji}$. For more detailed discussion about the previous derivations, the reader is referred |1,3|.

Coriolis and Centripetal effects:

These terms of great importance in high speed operations which is the case in a lot of industrial applications. Eq. (4) can be simplified to give a reduced model of low computational complexity.

According to the mathematical identity.

$$\frac{\partial^2 A}{\partial_x \partial_y} = \frac{\partial}{\partial_x} \left(\frac{\partial A}{\partial_y}\right) \quad , \text{ A: matrix ; x,y scalar variables}$$

Eq.(4) could be manipulated as follows;

$$\frac{\partial H\ell}{\ell q_i} = H_\ell \, {}^\ell\Delta_i , \quad \left(\frac{\partial H\ell}{\ell q_i}\right)^T = {}^\ell\Delta_i^T \, H^T \tag{11}$$

So, $$\frac{\partial^2 H\ell}{\partial q_j \partial q_k} = \frac{\partial}{\partial q_j} \left(\frac{\partial H\ell}{\partial q_k} \right) = \frac{\partial}{\partial q_j} \left(H_\ell \, {}^\ell\Delta_k\right)$$

which if expanded more gives ;

$$\frac{\partial^2 H\ell}{\partial q_j \partial q_k} = \left(\frac{\partial H_\ell}{\partial q_j}\right) {}^\ell\Delta_k + H_\ell \left(\frac{\partial \, {}^\ell\Delta k}{\partial q_j}\right) \tag{12}$$

The second term of eq.(12) could be neglected because of its small significance in effecting the accuracy of the calculations, which yields,

$$\frac{\partial^2 H\ell}{\partial q_j \partial q_k} = \frac{\partial H\ell}{\partial q_j} \, {}^\ell\Delta_k \tag{13}$$

Substituting (6) into (13),

$$\frac{\partial^2 H\ell}{\partial q_j \partial q_k} = H_\ell \, {}^\ell\Delta_j \, {}^\ell\Delta_k \tag{14}$$

now substituting (11), (14) into eq.(4) gives a better form for simulation purposes;

$$P_{ijk} = \sum_{\ell=\max i,j,k}^{n} \mathrm{Tr} \left(H_\ell \, {}^\ell\Delta_j \, {}^\ell\Delta_k J_\ell \, {}^\ell\Delta_i^T H^T{}_\ell\right) \tag{15}$$

Eq.(15) could be simplified further; premultiplying and post multiplying by H_ℓ and $H^T{}_\ell$ respectively will effect the rotation part only, hence the trace will remain unchanged.

Eq.(15) will reduce to:

$$P_{ijk} = \sum_{\ell=\max i,j,k}^{n} \left({}^\ell\Delta_j \, {}^\ell\Delta_k J_\ell \, {}^\ell\Delta_i{}^T\right) \tag{16}$$

now expanding the expression; and assuming a matrix (M) such that:

$$M = {}^\ell\Delta_j \, {}^\ell\Delta_K J_\ell \, {}^\ell\Delta_i^T$$

the matrix (M) will have the following form:

$$M = \begin{vmatrix} m_{11} & m_{12} & m_{13} & \emptyset \\ m_{21} & m_{22} & m_{23} & \emptyset \\ m_{21} & m_{32} & m_{33} & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset \end{vmatrix}$$

the important elements of the previous matrix are the diagonal elements which constructs the trace operator. So eq. (16) will reduce to:

$$P_{ijk} = \sum_{\ell=\max i,j,k}^{n} (m_{11}+m_{22}+m_{33}) \qquad (17)$$

If in matrix $|J_i|$, $I_{xy_i} = I_{xz_i} = I_{yz_i} = \emptyset$ (which is the case in a lot of robot manipulators, the elements of eq.(17) will be as follows:

$$m_{11} = Q_1 + Q_2 + Q_3$$

where,

$$Q_1 = (-{}^{\ell}\delta_{iz})\{{}^{\ell}\delta_{jy}\ {}^{\ell}\delta_{kx} \frac{|I_{xx\ell} - I_{yy\ell} + I_{zz\ell}|}{2} + |{}^{\ell}d_{ky}(-{}^{\ell}\delta_{jz}) + {}^{\ell}\delta_{jy}\ {}^{\ell}d_{kz}|\, m_{\ell} y_{\ell}$$

$$Q_2 = ({}^{\ell}\delta_{iy})\{{}^{\ell}\delta_{jz}\ {}^{\ell}\delta_{kx} \left|\frac{I_{xx\ell} + I_{yy\ell} - I_{zz\ell}}{2}\right| + (-{}^{\ell}\delta_{jz})\ {}^{\ell}d_{ky} + {}^{\ell}\delta_{jy}\ {}^{\ell}d_{kz}\ m_{\ell} z_{\ell}$$

$$Q_3 = \{|(-{}^{\ell}\delta_{jz})\ {}^{\ell}\delta_{kz} + {}^{\ell}\delta_{jy}(-{}^{\ell}\delta_{ky})|X_{\ell} + {}^{\ell}\delta_{jy}\ {}^{\ell}\delta_{kx} y_{\ell} + {}^{\ell}\delta_{jz}\ {}^{\ell}\delta_{kx} z_{\ell} + |(-{}^{\ell}\delta_{jz})\ {}^{\ell}d_{ky} + {}^{\ell}\delta_{jy}\ {}^{\ell}d_{kz}|\}\ {}^{\ell}d_{ix} m_{\ell}$$

$$m_{22} = Q_4 + Q_5 + Q_6$$

where,

$$Q_4 = ({}^{\ell}\delta_{iz})\{{}^{\ell}\delta_{ky}\ {}^{\ell}\delta_{jx} \left|\frac{-I_{xx\ell} + I_{yy\ell} + I_{zz\ell}}{2}\right| + |{}^{\ell}\delta_{jz}\ {}^{\ell}d_{kx} - {}^{\ell}\delta_{jx}\ {}^{\ell}d_{kz}|\, m_{\ell} x_{\ell}\}$$

$$Q_5 = ({}^{\ell}\delta_{ix})\{{}^{\ell}\delta_{jz}\ {}^{\ell}\delta_{ky} \left|\frac{I_{xx\ell} + I_{yy\ell} - I_{zz\ell}}{2}\right| + |{}^{\ell}\delta_{jz}\ {}^{\ell}d_{kx} - {}^{\ell}\delta_{jx}\ {}^{\ell}d_{kz}|\, m_{\ell} z_{\ell}\}$$

$$Q_6 = \{{}^{\ell}\delta_{jx}\ {}^{\ell}\delta_{ky} x_{\ell} + |{}^{\ell}\delta_{jz}(-{}^{\ell}\delta_{kz}) - {}^{\ell}\delta_{jx}\ {}^{\ell}\delta_{kx}|y_{\ell} + {}^{\ell}\delta_{jz}\ {}^{\ell}\delta_{ky}\ z_{\ell} + |{}^{\ell}\delta_{jz}\ {}^{\ell}d_{kx} - {}^{\ell}\delta_{jx}\ {}^{\ell}d_{kz}|\}\ {}^{\ell}d_{iy} m_{\ell}$$

$$m_{33} = Q_7 + Q_8 + Q_9$$

where

$$Q_7 = ({}^{\ell}\delta_{iy})\{{}^{\ell}\delta_{kz}\,{}^{\ell}\delta_{jx}\left|\frac{-I_{xx\ell}+I_{yy\ell}+I_{zz\ell}}{2}\right| + (-{}^{\ell}\delta_{jy})\,{}^{\ell}d_{kx} + {}^{\ell}\delta_{jx}\,{}^{\ell}d_{ky}|\; m_{\ell}x_{\ell}\}$$

$$Q_8 = ({}^{\ell}\delta_{ix})\{{}^{\ell}\delta_{kz}\,{}^{\ell}\delta_{jy}\left|\frac{I_{xx\ell}-I_{yy\ell}+I_{zz\ell}}{2}\right| + (-{}^{\ell}\delta_{jy})\,{}^{\ell}d_{kx} + {}^{\ell}\delta_{jx}\,{}^{\ell}d_{ky}|\; m_{\ell}y_{\ell}\}$$

$$Q_9 = \{{}^{\ell}\delta_{jx}\,{}^{\ell}\delta_{kz}x_{\ell} + {}^{\ell}\delta_{jy}\,{}^{\ell}\delta_{kz}y_{\ell} + |(-{}^{\ell}\delta_{jy})\,{}^{\ell}\delta_{ky} + {}^{\ell}\delta_{jx}(-{}^{\ell}\delta_{kx})|\,z_{\ell} + |(-{}^{\ell}\delta_{jy})\,{}^{\ell}d_{kx} + {}^{\ell}\delta_{jx}\,{}^{\ell}d_{ky}|\}\,{}^{\ell}d_{iz}m_{\ell}$$

Simulation Results:

Computer programs were written to validate the simplified formulation by comparing the force and torque values calculated by

(a) the classical Lagrangian formulation and
(b) the simplified formulation developed in this paper.

For our example a stanford manipulator was simulated.

As an example, each joint position parameter is chosen as Θ_i = Ø.3 radians (i=1,2,...,6) for all the cases considered in our example. The resulting joint forces and torques were calculated for the different joint velocities and accelerations. The velocities and accelerations were chosen to give realistic simulation results, whilst maintaining consistency with the existing robot models.

Case (1):

joint velocities ($\dot{\Theta}_i$) = 0.5 rad/S.
joint accelerations ($\ddot{\Theta}_i$) = 0.5 rad/S^2.

Table 1.

Case \ Joint	1(Nm)	2(Nm)	3(N)	4(Nm)	5(Nm)	6(nm)
I	0.754	3.96	-57.26	0.0007	0.656	0.0105
II	0.739	4.041	-57.38	0.0074	0.7186	0.01048

I: Lagrangian
II: This paper model.

Case (2):

joint velocities ($\dot{\theta}_i$) = 0.8 rad/S.
joint accelerations ($\ddot{\theta}_i$) = 0.8 rad/S^2.

Table 2.

Joint / Case	1(Nm)	2(Nm)	3(N)	4(Nm)	5(Nm)	6(Nm)
I	1.1	4.91	-55.48	0.017	0.706	0.0168
II	0.946	4.95	-55.96	0.076	0.84	0.017

Case (3):

joint velocities ($\dot{\theta}_i$) = 1 rad/S.
joint accelerations ($\ddot{\theta}_i$) = 1 rad/S^2.

Table 3.

Joint / Case	1(Nm)	2(Nm)	3(N)	4(Nm)	5(Nm)	6(Nm)
I	1.178	5.432	-54.47	0.026	0.74	0.0211
II	1.184	5.604	-55.07	0.11	0.95	0.0209

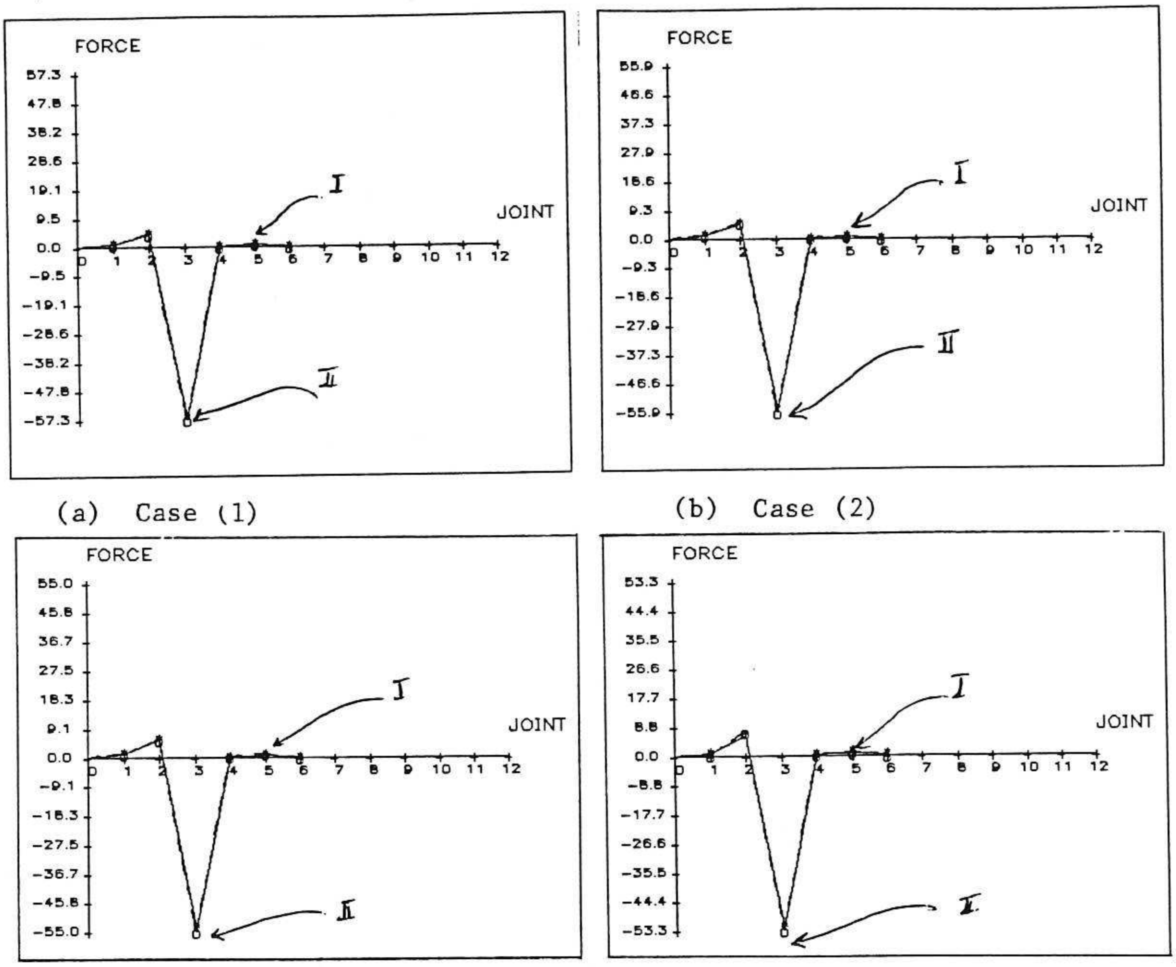
Case (4):

joint velocities ($\dot{\theta}_i$) = 1.5 rad/S.
joint accelerations ($\ddot{\theta}_i$) = 1.5 rad/S^2.

Table 4.

Joint / Case	1(Nm)	2(Nm)	3(N)	4(Nm)	5(Nm)	6(Nm)
I	0.63	6.18	-52.79	0.05	0.82	0.0318
II	0.392	7.17	-53.3	0.0686	0.788	0.0313

The previous four cases were plotted to show the difference between the two schemes, as shown in Fig.1.



(a) Case (1) (b) Case (2)

(c) Case (3) (d) Case (4)

Fig.1 Forces and torques effecting each joint in case (I) and (II).

In calculating the forces and torques effecting each joint, some simplifications were used to increase the speed and efficiency of the calculations |3,7| such as $P_{iii}=0$ always, because the joint which generates the centripetal force will not be effected by it, and $P_{ijk}=0$, if $i=K$, $K>j$.

Concluding Remarks:

The equations which describe the behaviour of a robot manipulator consist of a considerable number of differential and non-linear terms which complicate the calculations and make the real-time control a very hard task. Coriolis and Centripetal effects are of special importance in high speed movements. These effects were studied and a simplified representation was produced based on the Lagrangian formulation Simulation and numerical results were presented at different speeds (see table 1,2,3,4) for comparison between the classical Lagrangian formulation of coriolis and centripetal effects and the simplified model.

References:

(1) Paul, R,P, 1981, "Robot Manipulators: Mathematics, Programming and Control", MIT Press, Cambridge,Mass.

(2) Paul, R,P, 1972, "Modelling, Trajectory Calculation and Servoing of a Computer Controlled Arm", Stanford Artificial Intelligence Laboratory, Stanford University, AIM 177.

(3) Bejczy,A,K, 1974, "Robot Arm Dynamics and Control", NASA-JPL Technical Memorandum, 33-669.

(4) Hollerbach,J,M, 1980, "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation", IEEE Trans. on Systems, Man, and Cybernetics, SMC.10, pp 730-736.

(5) Lee,C.S. et al., 1983, "Development of the Generalized D'Alembert Equations of Motion for Mechanical Manipulators". Proc. of 22nd Conf. on Decision and Control, Dec.14-16.

(6) Orin D,E, et al., 1979, "Kinematic and Kinetic Analysis of Open Chain Linkages Utilizing Newton-Euler Methods", Math.Biosc., 43, 107-130.

(7) Liegeois,A, 1984, "Performance and Computer-Aided Design", Robot Technology, Volume 7.

(8) Luh, J,Y,S, et al., 1980, "On-line Computational Scheme for Mechanical Manipulators", Trans. ASME, J. Dynamic Sytems, Meas., and Control, 102, pp 69-76.

(9) Walker, M.W., Orin,D,E, 1982, "Efficient Dynamic Computer Simulation of Robotic Mechanisms", Trans. ASME, J Dynamic Systems, Meas., and Control, 104, pp 205-211

(10) Denavit,H., Hartenberg,R, 1955 "A Kinematic Notation for Lower Pair Mechanisms Based on Matrices", J. Applied Mechanics, 22, pp 215-221.

(11) Silver,W,M., 1982, "On the Equivalence of Lagrangian and Newton-Euler Dynamics for Manipulators", Int.J. of Robotics Res., Vol.1, No.2.

(12) Raibert,M.H., Horn, B.K, 1978, "Manipulator Control using the Configuration Space Method", The Industrial Robot, Vol.5, No.2, pp 69-73.

(13) Kane,T., Levinson,D, 1983, "The Use of Kane's Dynamical Equations in Robotics", Int.J.Robotics Res., Vol.2, No.3.

(14) Mahil S,S, 1982, "On the Application of Lagrange's Method to the Description of Dynamic Systems", IEEE Trans, Systems, Man and Cybernetics, Vol.SMC.12, pp 877-889.

(15) Armstrong, W,W, 1979, "Recursive Solution to the Equations of Motion of an N-link Manipulator", Proc. 5th World Congress on Theory of Machines and Mechanisms.

(16) Lee., C,S, Chang, P,R, 1986, "Efficient Parallel Algorithm for Robot Inverse Dynamics Computation", IEEE Trans. Systems, Man and Cybernetics, Vol.SMC.16, No.4.

(17) Binder., E,E, Herzog, J,H, 1986, "Distributed Computer Architecture and Fast Parallel Algorithms in Real-Time Robot Control", IEEE Trans. Systems, Man and Cybernetics, Vol.SMC.16, No.4.

(18) Luh, J,Y,S., Lin, C,S, 1982, "Scheduling of Parallel Computation for a Computer Controlled Mechanical Manipulator", IEEE Trans. Syst, Man and Cybernetics. Vol.SMC12, pp 214-234.

(19) Lathrop, L,H, 1983, "Parallelism in Manipulator Dynamics", MIT Artificial Intelligence Lab., Cambridge, MA, Tech.Rep.No.754.