PROPERTIES AND DATA STRUCTURES FOR EFFICIENT ADAPTIVE CONTROL ALGORITHMS IN ROBOT MANIPULATORS

J. C. Fernández * L. Peñalver ** V. Hernández * J. Tornero ******

* *Departamento de Informática*
Universidad Jaume I, 12071-Castellón (Spain)
Phone: +34-964-728265; Fax: +34-964-728435
e-mail: jfernand@inf.uji.es
** *Dept. de Informática de Sistemas y Computadores*
Universidad Politécnica de Valencia, 46071-Valencia (Spain)
Phone: +34-96-3877572; Fax: +34-96-3877579
e-mail: lourdes@disca.upv.es
*** *Dept. de Sistemas Informáticos y Computación*
Universidad Politécnica de Valencia, 46071-Valencia (Spain)
Phone: +34-96-3877356; Fax: +34-96-3877359
e-mail: vhernand@dsic.upv.es
**** *Dept. de Ingeniería de Sistemas y Automática*
Universidad Politécnica de Valencia, 46071-Valencia (Spain)
Phone: +34-96-3879578; Fax: +34-96-3879579
e-mail: jtornero@isa.upv.es

Abstract: The dynamics equation of robot manipulators is non linear and coupled. One way to solve the control movement is to use an inverse dynamic control algorithm that requires a full knowledge of the dynamics of the system. This knowledge is unusual in industrial robots, so adaptive control is used to identify these unknown parameters. In general their linear regressor is based on the linear relationship between the unknown inertial parameters and their coefficients. It is difficult to obtain a general algorithm for this relationship when more than two links are considered. In this paper a formulation to generalize this relationship is presented. This formulation is applied to Slotine & Li adaptive algorithm. The results have been obtained by simulation for a robot Puma with six links. *Copyright ©2000 IFAC*

Keywords: Robot manipulator, Lagrange-Euler formulation, algebraic operators, inertial parameters, adaptive control algorithms.

1. INTRODUCTION

The dynamic equation of robot manipulator torque in open chain is determined by highly coupled and non linear differential equation systems. Thus, it is necessary to use approximate or cancelling techniques for applying some control algorithms over the full system (multivariable control). One example of these techniques is known as inverse dynamic control. The idea is to perform a change of variable for cancelling the non linear terms and the coupled variables.

To apply these control techniques it is necessary to know the dynamics of the system, that is, the differential equations that define this system. This knowledge allows the existing relations among the different links to be established. The links to establish the kinematic relations of the system are defined from the Denavit-Hartenberg parameters. The inertial parameters, the mass and the inertial moments for each arm that are usually unknown, provide full knowledge of the dynamics. These parameters should be estimated using last square or adaptive control techniques.

Using adaptive control it is possible to solve both problems, movement control and parameter identification. Some of these algorithms can be found in (Craig, 1988), (Spong and Vidyasagar, 1988), (Ortega and Spong, 1989), (Johansson, 1990). All these algorithms use the following property.

Property All constant parameters of interest, such as the mass and inertial momentum, etc. appear as coefficients of known functions of generalized coordinates. Taking each coefficient as a parameter a linear relationship can be obtained and thus the dynamic equation can be written:

$$\tau = Y_\tau(q, \dot{q}, \ddot{q})\theta, \tag{1}$$

where $Y_\tau(q, \dot{q}, \ddot{q})$, known as regressor, is a $n \times r$ matrix, being n the number of links and r the number of parameters; and θ is the $r \times 1$ parameter vector. Considering all different parameters, r will be $10n$.

One of the problems in applying this kind of algorithm is that of obtaining the relationship given in equation (1). This problem has several solutions developing it symbolically by hand, or using a program such as Mathematica or Maple. (Gautier *et al.*, 1990) obtains the lineal and minimal expression of (1) from the modified Denavit-Hartenberg parameters and applies some properties of inertial parameters. These solutions are very complex for two or more links. (Khalil and Creusot, 1997) presents a program to solve this, but the solution is for a particular robot and is difficult to generalize and extend to computable algorithms. Other solutions are those given by (Ling, 1995) based on the Newton-Euler formulation but due to the complexity of the solution it is difficult to generalize to different adaptive control algorithms. One of the most recent solutions is given by (Kawasaki and Shimizn, 1998), and builds minimal new dynamic equations using the Grobner basis. This solution is more general but has the same drawback as the solution given by (Khalil and Creusot, 1997).

This paper formulates in detail the linear relationship for the torque equation and extends it to the known algorithm given by Slotine & Li (Slotine and Li, 1987). To obtain the relationship expressed in (1), several algebraic properties and definitions can be used to obtain this relationship symbolically and algorithmically for any robot manipulator in open chain. The presented formulation is a computable and general solution for any robot manipulator in open chain using the Denavit-Hartenberg parameters and is an extension of the Lagrange-Euler formulation.

The structure of the paper is the following: In section two the algebraic definitions and properties used to obtain the linear relationship of the dynamic equations with the inertial parameters are presented. In section three the reformulation for a torque equation is given. Section four describes the Slotine & Li algorithm and its reformulation. The results for a robot Puma are described in section five. And finally the conclusions of this paper are presented in last section.

2. ALGEBRAIC DEFINITIONS AND PROPERTIES

Firstly, the definitions and properties of the well-known Kronecker product operator and the column vector operator are presented.

The Kronecker product of the two matrices $A = [a_{ij}] \in R^{m\times n}$ and $B \in R^{p\times q}$ is a matrix $A \otimes B \in R^{mp\times nq}$.

The column vector operator on the space of matrices $R^{m\times n}$ is defined by

$$\nu : R^{m\times n} \to R^{mn\times 1},$$

$$\nu(A) = \begin{bmatrix} a_{11} \dots a_{m1} \dots a_{1n} \dots a_{mn} \end{bmatrix}^T . \tag{2}$$

This operator is linear and bijective, and it defines an isomorphism between the spaces $R^{m\times n}$ and $R^{mn\times 1}$. The inverse operator is defined by

$$\nu^{-1} : R^{mn\times 1} \longrightarrow R^{m\times n}$$

$$\nu^{-1}(b) = \begin{bmatrix} b_1 & b_{m+1} & \cdots & b_{(n-1)m+1} \\ b_2 & b_{m+2} & \cdots & b_{(n-1)m+2} \\ \vdots & \vdots & \ddots & \vdots \\ b_m & b_{2m} & \cdots & b_{mn} \end{bmatrix}, \tag{3}$$

where $b = [b_i] \in R^{mn\times 1}$. If $b = \nu(A)$ then $b_k = a_{ij}$ where $k = i + (j-1)m$.

These two algebraic operators satisfy the following property, if $A \in R^{m\times n}$, $B \in R^{n\times p}$ and $C \in R^{p\times q}$ then

$$\nu(ABC) = (C^T \otimes A)\nu(B).$$

2.1 *New Definitions and propositions*

Two new concepts are introduced: the trace of a vector and the row-trace of a matrix.

Definition 1 Consider the vector $b \in R^{m^2\times 1}$. The trace of this vector is defined by $tr(b) =$

$tr\left(\nu^{-1}(b)\right) = b_1 + b_{m+2} + b_{2m+3} + \cdots + b_{m^2}$. Note that the trace of b is equal to the trace of the matrix $\nu^{-1}(b)$.

Note that if $A \in R^{m \times n}$ then $tr(A) = tr(\nu(A))$.

Definition 2 Given a matrix $A \in R^{m^2 \times nq}$, and the partition by columns of A, $A = [a_1, a_2, \cdots, a_{nq}], a_i \in R^{m^2 \times 1}, i = 1, \cdots, nq$. The row-trace of A denoted by $rtr(\cdot)$, is the row vector defined by the following expression

$$rtr(A) = [tr(a_1), tr(a_2), \cdots, tr(a_{nq})]_{1 \times nq}.$$

Note that the row-trace of A is the row vector whose components are the traces of the columns of A.

Proposition 1 If $A \in R^{m^2 \times nq}$ and $B \in R^{n \times q}$ then $tr\left(A\nu(B)\right) = rtr(A)\nu(B)$.

The demostration is given in (Peñalver, 1998).

3. THE DYNAMIC MODEL

The dynamic equation of rigid manipulators with n arms in matrix form is

$$\tau = D(q)\ddot{q} + h(q, \dot{q}) + c(q), \tag{4}$$

where τ is the $n \times 1$ vector of nominal driving torques. q is the $n \times 1$ vector of nominal generalized coordinates, $\dot{q}$ and $\ddot{q}$ are the $n \times 1$ vectors of the first and second derivatives of the vector q, respectively, D is the inertia matrix, $h(q, \dot{q})$ is the vector of centrifugal and Coriolis forces and $c(q)$ is the vector of gravitational forces.

The elements of the inertial symmetric matrix D are given by

$$d_{ij} = \sum_{k=\max(i,j)}^{n} tr(U_{jk} J_k U_{ik}^T) \quad i, j = 1, \cdots, n, \tag{5}$$

where J_k is the inertia tensor related to the link k, and U_{jk} is the effect of the movement of link k over all the points of link j.

$h(q, \dot{q})$ is a $n \times 1$ vector whose elements are defined by

$$h_i = \sum_{j=1}^{n} \sum_{k=1}^{n} h_{ijk} \dot{q}_j \dot{q}_k \quad i = 1, \cdots, n, \tag{6}$$

where

$$h_{ijk} = \sum_{l=\max(i,j,k)}^{n} tr(U_{kjl} J_l U_{il}^T) \; i, j, k = 1, \cdots, n. \tag{7}$$

where U_{kjl} is the effect of the movement of links j and l over all the points of link k. These matrices define a block structure given by

$$H = \left[H_1 \; H_2 \; \cdots \; H_n \right]^T, \tag{8}$$

where $H_i = [h_{ijk}]_{j,k=1,\cdots,n}$.

Note that H_i is a symmetric matrix. Moreover, the vector of centrifugal and Coriolis forces can be expressed as

$$h(q, \dot{q}) = \left[q^T H_1 q \; q^T H_2 q \; \cdots \; q^T H_n q \right]^T,$$

because $U_{kji} = U_{jki}$.

The elements of the vector of gravitational forces, $c(q) \in R^{n \times 1}$, are given by

$$c_i = \sum_{j=i}^{n} (-m_j g^T U_{ij} \, {}^j\bar{r}_j) \quad i = 1, \cdots, n. \tag{9}$$

In this expression ${}^j\bar{r}_j$ is the position of the centre of mass of link j (m_j) with respect to the origin of the coordinates of link j.

The idea is to obtain a linear relationship formulation of the torque equation

$$\tau = Y_\tau \; (q, \dot{q}, \ddot{q})\theta, \tag{10}$$

where Y_τ is a time variable $n \times 10n$ matrix function and θ is a $10n \times 1$ matrix composed of all the constant and inertial parameters. To obtain this relationship the known algebraic definitions and properties and the new ones, which are described in the previous section are used.

3.1 *Linear Relationship for the dynamic equation*

To obtain a computable expression for Y_τ the previous definitions are used. The same procedure is applied to each element of the torque equation, that is, the inertial matrix, centrifugal and Coriolis forces, and gravitational forces. Using these independent expressions for each dynamic term the computable expression for Y_τ can be obtained.

From equation (5) and using proposition 1 the following expression for the inertial matrix can be obtained

$$D(q)\ddot{q} = Y_D(q, \ddot{q})\theta_D,$$

where

$$Y_D(q, \ddot{q}) = \begin{bmatrix} rtr\,(B_{D11}) & \cdots & rtr\,(B_{D1n}) \\ 0 & \ddots & rtr\,(B_{D2n}) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & rtr\,(B_{Dnn}) \end{bmatrix}, \tag{11}$$

$$\theta_D = \begin{bmatrix} \nu(J_1) \\ \nu(J_2) \\ \vdots \\ \nu(J_n) \end{bmatrix}, \tag{12}$$

with B_{Dik} defined by

$$B_{Dik} = \sum_{j=1}^{k} (U_{ik} \otimes U_{jk}) \ddot{q}_j. \qquad (13)$$

Note that Y_D is an $n \times 16n$ upper triangular matrix and θ_D is a $16n \times 1$.

Given expression (6) for link i, applying proposition 1 and reordering the adders the following expression for the centrifugal and Coriolis forces can be obtained

$$h(q, \dot{q}) = Y_H(q, \dot{q}) \theta_h,$$

where

$$Y_H(q, \dot{q}) = \begin{bmatrix} rtr\,(B_{h11}) & \cdots & rtr\,(B_{h1n}) \\ 0 & \ddots & rtr\,(B_{h2n}) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & rtr\,(B_{hnn}) \end{bmatrix}, \qquad (14)$$

$$\theta_h = \theta_D, \qquad (15)$$

with

$$B_{hij} = \sum_{l=1}^{j} \sum_{k=1}^{j} (U_{ij} \otimes U_{klj}) \dot{q}_l \dot{q}_k. \qquad (16)$$

Note that Y_H is an $n \times 16n$ upper triangular matrix and θ_h is a $16n \times 1$ vector.

Given expression (9), the vector of gravitational forces can be expressed as a linear relationship with the inertial parameters

$$c(q) = Y_c(q) \theta_c, \qquad (17)$$

where

$$Y_c(q) = - \begin{bmatrix} g^T U_{11} & g^T U_{12} & \cdots & g^T U_{1n} \\ 0 & g^T U_{22} & \cdots & g^T U_{2n} \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & g^T U_{nn} \end{bmatrix}, \qquad (18)$$

$$\theta_c = \begin{bmatrix} m_1 \bar{r}_1 \\ m_2 \bar{r}_2 \\ \vdots \\ m_n \bar{r}_n \end{bmatrix}. \qquad (19)$$

3.2 *Torque equation*

The dynamic equation (4) can be written as

$$\tau(t) = [Y_D(q, \ddot{q}) + Y_H(q, \dot{q})] \theta_D + Y_c(q) \theta_c. \qquad (20)$$

Defining

$$\tilde{Y}_c = [\tilde{Y}_{c1}, \tilde{Y}_{c2}, \cdots, \tilde{Y}_{cn}], \qquad (21)$$

where

$$\tilde{Y}_{ci} = [0_{n \times 4}, 0_{n \times 4}, 0_{n \times 4}, Y_{c_i}], i = 1, \cdots, n. \qquad (22)$$

and

$$Y_{c_i} = \left[g^T U_{1i} \cdots g^T U_{ii} \; 0 \cdots 0 \right]^T, \qquad (23)$$

and including θ_c in θ_D, then

$$\tau = Y_\tau \theta_\tau, \qquad (24)$$

where $Y_\tau = [Y_D + Y_H + \tilde{Y}_c]$ and $\theta_\tau = \theta_D$. In expression (24), Y_τ is the $n \times 16n$ matrix with the variable dynamic terms and θ_τ is the $16n \times 1$ vector with the inertial parameters.

As the inertia tensor J_i, $i = 1 : n$, is symmetrical then θ_τ has $10n$ different parameters. It is possible to define Y_τ as a matrix $n \times 10n$ in spite of $n \times 16n$ as

$$\check{Y}_\tau(i, z) = \begin{cases} Y_\tau(i, l) + Y_\tau(i, w) & l = 16(i-1) + (r + 4(s-1)) \\ & w = 16(i-1) + (s + 4(r-1)) \\ Y_\tau(i, l) & l = w \rightarrow r = s \end{cases} \qquad (25)$$

for $r = 1 : 4$, $s = 1 : 4$ and $z = 10(i-1) + 4(r-1) - r\frac{r-1}{2} + s$.

4. SLOTINE & LI ALGORITHM

To improve this formulation the adaptive algorithm given by Slotine & Li (Slotine and Li, 1987) is used. Firstly the algorithm is presented, then its adaptive version is shown and finally the formulation presented is applied.

4.1 *Slotine & Li Control Algorithm*

The main characteristic of this algorithm is that it does not need the aceleration measure or estimation, or the inertial matrix inverse. This approach is not exactly based on dynamic inverse, because the control objective it not feedback linearization but only preservation of the passivity properties of the rigid robot in the closed loop. But this method has been chosen because it is a well known adaptive algorithm.

Considering

$$\tau = D(q) \dot{v} + h(q, \dot{q}) v + c(q) - K_D (\dot{q} - v), \qquad (26)$$

defining $v = \dot{q}^d - \Lambda(q - q^d) = \dot{q}^d - \Lambda e$, where Λ is a positive force diagonal matrix, $r = \dot{q} - v = \dot{e} + \Lambda e$ and replacing (26) in (4) then

$$D(q) \dot{r} + h(q, \dot{q}) r + K_D r = 0. \qquad (27)$$

Note that this control law does not lead to a close loop system. This control law is useful when considering the parameter estimation problem.

4.2 *Adaptive version*

From the dynamic equation (4) the control law chosen is

$$u = \widehat{D}\,(q)\,\dot{v} + \widehat{h}\,(q,\dot{q})\,v + \widehat{c}(q) - K_D r. \quad (28)$$

Replacing this expression in (4) then

$$D\ddot{q} + h\dot{q} + c = \widehat{D}\dot{v} + \widehat{h}v + \hat{c} - K_D r. \quad (29)$$

Given that $\ddot{q} = \dot{r} + \dot{v}$ and $\dot{q} = r + v$ the previous equation can be written as

$$D\dot{r} + hr + K_D r = \widetilde{D}\dot{v} + \widetilde{h}v + \widetilde{c} = Y_S(q,\dot{q},v,\dot{v})\widetilde{\theta}, (30)$$

where $\widetilde{D} = \widehat{D} - D$, $\widetilde{h} = \widehat{h} - h$, $\widetilde{c} = \widehat{c} - c$ and $\widetilde{\theta} = \widehat{\theta} - \theta$. Note that Y_S is not a function of manipulator aceleration. Only vector $\dot{v}$ depends on the reference aceleration and the velocity error.

The adaptive control law resulting is

$$\dot{\hat{\theta}} = -\Gamma^{-1} Y_S^T(q,\dot{q},v,\dot{v})r, \quad (31)$$

$$\tau = Y_S(q,\dot{q},v,\dot{v})\hat{\theta} - K_D r. \quad (32)$$

Moreover, the matrices Λ, K_D and Γ are diagonal matrices.

4.3 *Reformulation*

To obtain a computable version of $Y_S(q,\dot{q},v,\dot{v})$ the results obtained previously for the dynamic equation are used.

- Relative terms to inertial matrix $D(q)\dot{v}$:

$$D(q)\dot{v} = Y_{D\dot{v}}(q)\theta_D, \quad (33)$$

where

$$Y_{D\dot{v}}(q) = \begin{bmatrix} rtr\,(B_{D11}) & \cdots & rtr\,(B_{D1n}) \\ 0 & \ddots & rtr\,(B_{D2n}) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & rtr\,(B_{Dnn}) \end{bmatrix}, (34)$$

$$\theta_D = \begin{bmatrix} \nu(J_1) \\ \nu(J_2) \\ \vdots \\ \nu(J_n) \end{bmatrix}, \quad (35)$$

with

$$B_{Dik} = \sum_{j=1}^{k} (U_{ik} \otimes U_{jk})\dot{v}_j. \quad (36)$$

- Coriolis and Centrifugal forces $h(q,\dot{q})$:

$$h(q,\dot{q})v = Y_{Hv}(q,\dot{q})\theta_h, \quad (37)$$

where

$$Y_{Hv}(q,\dot{q}) = \begin{bmatrix} rtr\,(B_{h11}) & \cdots & rtr\,(B_{h1n}) \\ 0 & \ddots & rtr\,(B_{h2n}) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & rtr\,(B_{hnn}) \end{bmatrix}, (38)$$

$$\theta_h = \theta_D, \quad (39)$$

with

$$B_{hil} = \sum_{j=1}^{l}\sum_{k=1}^{l} (U_{il} \otimes U_{kjl})\dot{q}_j v_k. \quad (40)$$

- The terms related to the gravitational forces are the same as (18) and (19).

From the previuos results, the computable version of $Y_S(q,\dot{q},v,\dot{v})$ is given by

$$Y_S(q,\dot{q},v,\dot{v}) = Y_{D\dot{v}} + Y_{Hv} + \tilde{Y}_c.$$

Using (25), the dimension of Y_S is $10n \times n$.

5. RESULTS

With the results presented in previous sections a general algorithm to compute the known variable terms of adaptive control is obtained. This is easily applicable to any robot manipulator in open chain if the Denavit-Hartenberg parameters are known. The figures (1), (2) and (3) show the results obtained using the data corresponding to the Puma 600 robot, applying the previous control law, considering the following values of the constant matrices for the Slotine & Li algorithm, $\Lambda = 39$, $\Gamma = 0.075$, $K_D = 35$, and using a set of excited points to build polynomial trajectories for the reference trajectories used

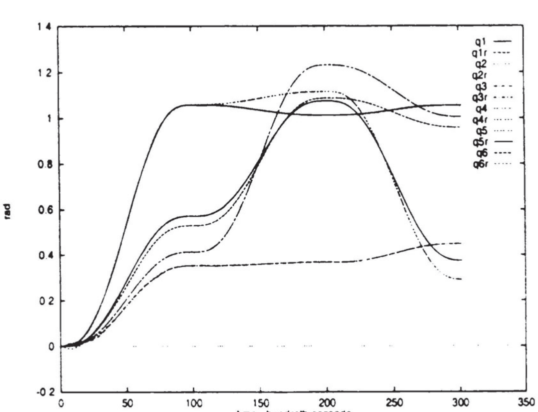

Fig. 1. Positions and references for joints 1 to 6.

Applying the considerations described in (Peñalver, 1998), the figure (4) shows the inertial parameters convergence for the Puma 600 robot.

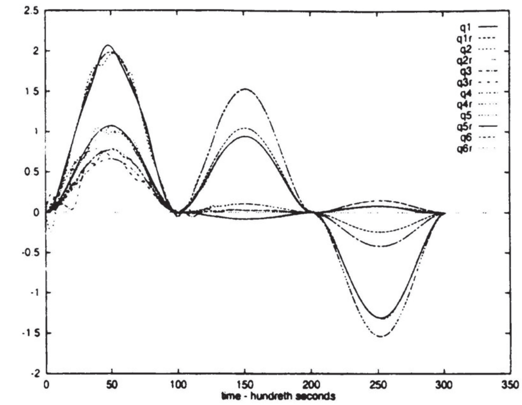

Fig. 2. Velocities and references for joints 1 to 6.

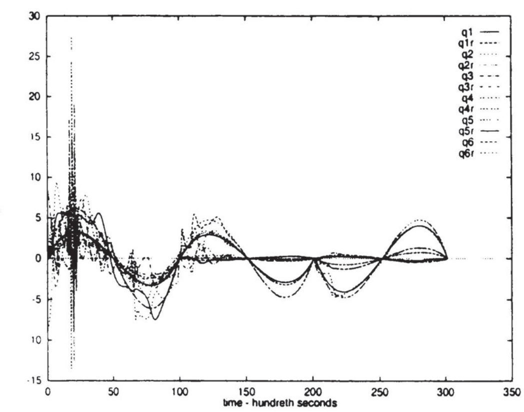

Fig. 3. Accelerations and references for joints 1 to 6.

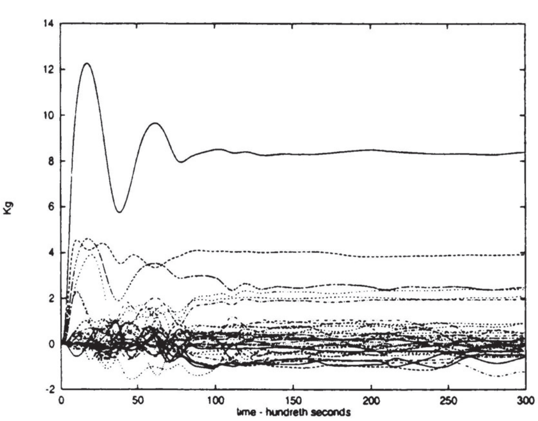

Fig. 4. Inertial parameters convergence.

6. CONCLUSIONS

The goal was to design a generalized formulation for the linear relationship between the variable dynamic terms and the inertial ones. The presented formulation provides this linear relationship knowing only the Denavit-Hartenberg parameters, common information in industrial robot manuals, for any robot manipulator in open chain. Moreover, this formulation is also applicable to any other way of expressions of the dynamic equations, such as Hamiltonian, etc.

Although the use of Lagrange-Euler formulation seems to produce a high computational cost, it is possible to reduce it exploiting the matrix structure by eliminating the high quantity of null terms and exploiting the symmetry properties of the matrices. A previous study of this method can be seen in (Fernández *et al.*, 1997). Moreover computer time response can be reduced using parallel computing.

Finally, to show that the presented formulation can be used for a robot manipulator with more than two links, this formulation has been applied to a six links Puma manipulator.

7. REFERENCES

Craig, J. (1988). *Adaptative Control of Mechanical Manipulators.*

Fernández, J. C., L. Pe nalver, M. Sobejano, J. Martínez, J. Tornero and V. Hernández (1997). An interface based on transputers to simulate the dynamic equation of robot manipulators. In: *Lectures Notes in Computers Science, Vecpar96* (Springer-Verlag, Ed.). pp. 365–379.

Gautier, M., F. Bennis and W. Khalil. (1990). The use of generalized links to determine the minimun inertial parameters of robots. *Journal of Robotic Systems* **7**(2), 225–242.

Johansson, R. (1990). Quadratic optimization of motion coordination and control. *IEEE Trans. Automat. Control* **35**,(11), 1197–1208.

Kawasaki, H. and T. Shimizn (1998). Symbolic analysis of robot base parameter set using grobner-basis. *Journal of Robotics and Mechatronics* **10**(6), 475–481.

Khalil, W. and D. Creusot (1997). Symoro+: A system for the symbolic modelling of robots. *Robotica* **15**, 153–161.

Ling, S.K. (1995). Minimal lineal combinations of inertial parameters of a manipulator. *IEEE-TransRA* **11**(3), 360–373.

Ortega, J.M. and M. Spong (1989). Adaptive motion control of rigid robots: A tutorial. *Automatica* **25**(6), 877–888.

Peñalver, L. (1998). Modelado dinámico e identificación paramétrica para el control de robots manipuladores. PhD thesis. Univ. Politécnica de Valencia. Dept. de Ingeniería de Sistemas y Computadores, Valencia (Spain).

Slotine, J. J. and W. Li (1987). On adaptative control of robot manipulators. *Internat. J. Robotics Research.* **6**(3), 49–59.

Spong, M.W. and M. Vidyasagar (1988). *Robot dynamics and control.* John Whiley and Sons.