**Monograph:**

Zomaya, A.Y. and Morris, A.S. (1988) The Dynamic Performance of Robot Manipulators Under Different Operating Conditions. Research Report. Acse Report 345 . Dept of Automatic Control and System Engineering. University of Sheffield

# The Dynamic Performance of Robot Manipulators Under Different Operating Conditions

A. Y. ZOMAYA and A. S. MORRIS

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

Research Report No. 345

October 1988

Abstract:-

The dynamical performance of robot manipulators is greatly affected by the different payloads handled by the end effector (hand). Hence, it is very important, especially for industrial applications, to study the different interconnected relationships between the manipulator joints, speeds, loads, and actuation forces. In this paper, a simplified symbolic Lagrangian representation of the different terms is presented, with emphasis on the coriolis and centripetal effects. The accuracy and computational efficiency of this new formulation is demonstrated by simulation of a Stanford and PUMA 560 manipulator. Useful quantitative measurements and error analysis are also included on the significance of coriolis and centripetal terms under different load and speed conditions.

## 1. Introduction:-

The mathematical formulation of the equation of motion of a robot manipulator divides into the two separate areas of :

(a) the inverse dynamics, which are concerned with finding the forces required to drive the arm through some specified trajectory.

(b) the forward dynamics, which are concerned with calculating the position, velocity and accleration of each link for a given set of applied forces.

The dynamics consists of a set of differential, coupled, non linear, and matrix oriented representation which describes the behaviour of the robot arm.

Various robot dynamic formulations have been proposed during the past few years. The Lagrange-Euler (LE) [1, 2, 3] has low computational efficiency but a very well structured and systematic representation that allows for different control applications. The Recursive Lagrangian [4 ] has better computation time but destroys the structure of the equations. The Generalized D' Alambert formulation [5] has a fair representation with some computation improvements. The Newton-Euler (NE) [6, 7, 8, 9] has a very efficient computational representation with very untidy recursive equations. Tabulation dependent schemes [12] have very serious difficulties owing to the enormous computer memory storage requirements. Other approaches include the dynamic equations of Kane [13] and the use of parallel processing and advanced computer architectures to reduce the computation-time [14, 15, 16, 17,]. Of the previous methods, the most commonly used are the (LE) and (NE). The interaction and equivalence between these schemes has been shown by Silver [11 ] and Turney et. al. [18].

In this paper, a simplified symbolic mathematical description of the dynamics based on the (LE) will be presented accompanied by execution-time results. Also for the first time, exact quantification is given of what effect the inertial parameters of coriolis and centripetal forces actually have as the load and speed are varied. This allows for a much clearer understanding in any particular circumstance of the limits of operating speed at which the value of the dynamic forces are fairly valid. The analysis is based on a set of compiled data of two robot arms (stanford, PUMA 560) to facilitate the robot dynamic performance problem.

## 2. The Lagrangian:-

The importance and usefulness of the (LE) rise from its simple, algorithmic and highly structured equation based on lagrangian mechanics which is derived from energetic principles.

The set of equations can be written in a compact form which is the final outcome from solving (LE):

$$F = D(v)\ \ddot{v} + C(v)\ \dot{v} + G(v) \quad \ldots \quad (1)$$

Where,

- D : n x n matrix which represent the coupling and effective inertia terms, position and acceleration dependent.
- C : n x n x n matrix which represent the centripetal and coriolis terms, position and velocity dependent.
- G : n - dimensional vector representing the gravity loading effects, position dependent.
- $v, \dot{v}, \ddot{v}$ : position, velovity and acceleration.
- n : degree of freedom.
- F : force (prismatic joint) or torque (revolute joint).

The very general form of eq.(1) is very important in state space and modern control applications [19, 20, 21]. Because of the changing geometrical configuration of the robot as it moves, the inertial parameters of the robot are time-varying. The inertial and gravity terms affect the servo stability and positioning accuracy of the arm. The coriolis and centripetal terms contributes little to the dynamic forces at low operational speeds but become highly significant at high speeds. Normal practice in industrial robots is therefore to limit their operational speed such that this problem is not encountered.

The use of (LE) is very useful but cannot be utilised in real-time control without further simplifications. Because of that, many attempts had been made to reduce the order of computations |3, 22, 23, 24, 25, 26, 27|. The nomenclature used in |1, 3| which is based on the Denavit-Hartenberg conventions |10| will be used in our discussion, Eq. (1) might be written in an alternate form:

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

where

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

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

$$P_{ij}, = \sum_{\ell=\max(i,j)}^{n} tr\left(\frac{\partial H_\ell}{\partial q_j}\ {}^{\ell}J\left(\frac{\partial H_\ell}{\partial q_i}\right)^T\right) \qquad (3)$$

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

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

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

$P_i$, gravity loading vector

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

$m_\ell$, mass of $\ell$ink $\ell$

${}^{\ell}r_\ell$, centre of mass of $\ell$ink $\ell$ according to its own coordinate .

$H_\ell$, 4 x 4 $\ell$ink transformation or denavit-hartenberg matrices.

g, gravitational effects vector.

and ${}^{\ell}J$ is a pseudo inertia matrix, which is luckly for most industrial manipulator, and in our case the stanford and PUMA 560, is of the form,

$$ {}^{\ell}J_{ij} = \begin{bmatrix} \frac{-I_{xx\ell}+I_{yy\ell}+I_{zz\ell}}{2} & 0 & 0 & m_\ell x_\ell \\ 0 & \frac{I_{xx\ell}-I_{yy\ell}+I_{zz\ell}}{2} & 0 & m_\ell y_\ell \\ 0 & 0 & \frac{I_{xx\ell}+I_{yy\ell}-I_{zz\ell}}{2} & m_\ell z_\ell \\ m_\ell x_\ell & m_\ell y_\ell & m_\ell z_\ell & m_\ell \end{bmatrix} $$

## 3. Simplification of the Lagrangian Formulation:-

### 3.1 Inertial and Gravity Terms:-

The effective and coupling inertia terms of eq. (3) have been shown in [1] to be:

$$P_{ij} = \sum_{\ell=\max(i,j)}^{n} tr\left({}^{\ell}\Delta_j {}^{\ell}J {}^{\ell}\Delta_i^T\right) \qquad (7)$$

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

$${}^{\ell}\Delta_i = \begin{bmatrix} 0 & -{}^{\ell}\delta_{iz} & {}^{\ell}\delta_{iy} & {}^{\ell}d_{ix} \\ {}^{\ell}\delta_{iz} & 0 & -{}^{\ell}\delta_{ix} & {}^{\ell}d_{iy} \\ -{}^{\ell}\delta_{iy} & {}^{\ell}\delta_{ix} & 0 & {}^{\ell}d_{iz} \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (8)$$

$$\text{Where } {}^{\ell}d_i = \begin{cases} \left(-{}^{i-1}n_{\ell x}\,{}^{i-1}P_{\ell y} + {}^{i-1}n_{\ell y}\,{}^{i-1}P_{\ell x}\right) i \\ +\left(-{}^{i-1}o_{\ell x}\,{}^{i-1}P_{\ell y} + {}^{i-1}o_{\ell y}\,{}^{i-1}P_{\ell x}\right) j \\ +\left(-{}^{i-1}a_{\ell x}\,{}^{i-1}P_{\ell y} + {}^{i-1}a_{\ell y}\,{}^{i-1}P_{\ell x}\right) k & \text{revolute joint} \\ \left({}^{i-1}n_{\ell z}\, i + {}^{i-1}o_{\ell z}\, j + {}^{i-1}a_{\ell z}\, k\right) & \text{prismatic joint} \end{cases}$$

$${}^{\ell}\delta_i = \begin{cases} \left({}^{i-1}n_{\ell z}\, i + {}^{i-1}o_{\ell z}\, j + {}^{i-1}a_{\ell z}\, k\right) & \text{revolute joint} \\ 0 & \text{prismatic joint} \end{cases}$$

By expanding eq. (7) to reduce the multiplication by zero operations and to give insight into more customization of the dynamics [27] which depends mainly on the arm architecture to lead to more simplifications. Assume a matrix (E) such that:

$$E = \left[\begin{array}{c|c} e_{ij} & 0 \\ \hline 0 & 0 \end{array}\right], e: 3 \times 3 \text{ matrix}$$

using the trace operator will give:

$$P_{ij} = \sum_{\ell=\max(i,j)}^{n} \sum_{m=1}^{3} e_{mm} \qquad (9)$$

Where $\sum_{m=1}^{n} e_{mm}$ is given as,

$$= {}^{\ell}J_{11} \begin{bmatrix} \delta_{iy} \\ \delta_{iz} \end{bmatrix}_{\ell} \cdot \begin{bmatrix} \delta_{jy} \\ \delta_{jz} \end{bmatrix}_{\ell} + {}^{\ell}J_{22} \begin{bmatrix} \delta_{ix} \\ \delta_{iz} \end{bmatrix}_{\ell} \cdot \begin{bmatrix} \delta_{jx} \\ \delta_{jz} \end{bmatrix}_{\ell} + {}^{\ell}J_{33} \begin{bmatrix} \delta_{ix} \\ \delta_{iy} \end{bmatrix}_{\ell} \cdot \begin{bmatrix} \delta_{jx} \\ \delta_{jy} \end{bmatrix}_{\ell}$$

$$+ {}^{\ell}J_{44} \begin{bmatrix} d_{ix} \\ d_{iy} \\ d_{iz} \end{bmatrix}_{\ell} \cdot \begin{bmatrix} d_{jx} \\ d_{jy} \\ d_{jz} \end{bmatrix}_{\ell} + {}^{\ell}J_{14} \left\{ \begin{vmatrix} \delta_{iz} & d_{iz} \\ \delta_{jy} & d_{jy} \end{vmatrix} + \begin{vmatrix} \delta_{jz} & d_{jz} \\ \delta_{iy} & d_{iy} \end{vmatrix} \right\}_{\ell}$$

$$+ {}^{\ell}J_{24} \left\{ \begin{vmatrix} \delta_{jx} & d_{jx} \\ \delta_{iz} & d_{iz} \end{vmatrix} + \begin{vmatrix} \delta_{ix} & \delta_{jz} \\ \delta_{jz} & d_{jz} \end{vmatrix} \right\}_{\ell}$$

$$+ {}^{\ell}J_{34} \left\{ \begin{vmatrix} \delta_{iy} & d_{iy} \\ \delta_{jx} & d_{jx} \end{vmatrix} + \begin{vmatrix} \delta_{jy} & d_{jy} \\ \delta_{ix} & d_{ix} \end{vmatrix} \right\}_{\ell}$$

where $|A|$: determinant of A, {.}: scalar multiplication.
The gravity loading vector is given in [1] to be:

$$P_i = {}^{i-1}g \sum_{\ell=i}^{n} m_{\ell} \, {}^{i-1}\bar{r}_{\ell} \qquad (10)$$

where

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

For a 6-dof arm at a given set of position variables, 36 elements of the matrix $[P_{ij}]$ should be evaluated, but due to symmetry the nember reduces to 21 elements. For the gravity loading vector 6 elements should be calculated [1,3,26].

## 3.2 Coriolis and Centripetal Effects:

These terms have great importance in high speed operations which is the case in many industrial applications. Eq. (4) can be simplified to give a reduced order of computation [28].
According to the mathematical identity,

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

Eq. (4) can be manipulated as follows;

$$\frac{\partial H_{\ell}}{\partial q_i} = H_{\ell} \, {}^{\ell}\Delta_i \qquad (11.1)$$

$$\left(\frac{\partial H_\ell}{\partial q_i}\right)^T = {}^\ell\Delta_i^T H_\ell^T \qquad (11.2)$$

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

expanding will result in;

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

neglect the second term of eq. (12) will yield,

$$\frac{\partial^2 H_\ell}{\partial q_j \partial q_k} = \left(\frac{\partial H_\ell}{\partial q_j}\right){}^\ell\Delta_k \qquad (13)$$

substituting (11.1) into (13),

$$\frac{\partial^2 H_\ell}{\partial q_j \partial q_k} = H_\ell\, {}^\ell\Delta_j\, {}^\ell\Delta_k \qquad (14)$$

now substituting (11.2) and (14) into eq. (4) gives a better form for simulation purposes;

$$P_{ijk} = \sum_{\ell=\max(i,j,k)}^{n} tr(H_\ell\, {}^\ell\Delta_j\, {}^\ell\Delta_k\, {}^\ell J\, {}^\ell\Delta_i^T H_\ell^T) \qquad (15)$$

Eq. (15) could be simplified further; premultiplying and postmultiplying by $H_\ell$ and $H_\ell^T$ respectively will effect the rotation part only, hence the trace operator will remain unchanged, eq. (15) will reduce to,

$$P_{ijk} = \sum_{\ell=\max(i,j,k)}^{n} ({}^\ell\Delta_j\, {}^\ell\Delta_k\, {}^\ell J\, {}^\ell\Delta_i^T) \qquad (16)$$

now expanding eq. (16) and assuming a matrix (u) such that:

$$u = {}^\ell\Delta_j\, {}^\ell\Delta_k\, {}^\ell J\, {}^\ell\Delta_i^T$$

the matrix (u) will have the same form of matrix (E), i.e.

$$u = \left[\begin{array}{c|c} u_{ij} & 0 \\ \hline 0 & 0 \end{array}\right]$$

the trace operator will give:

$$P_{ijk} = \sum_{\ell=\max(i,j,k)}^{n} \sum_{m=1}^{3} u_{mm} \qquad (17)$$

where $\sum_{m=1}^{3} u_{mm}$ is given as,

$$= {}^{\ell}J_{11}\; {}^{\ell}\delta_{jx} \begin{vmatrix} \delta_{ky} & \delta_{kz} \\ \delta_{iy} & \delta_{iz} \end{vmatrix}_{\ell} + {}^{\ell}J_{22}\; {}^{\ell}\delta_{ix}\; {}^{\ell}\delta_{jy} \begin{vmatrix} \delta_{kz} & \delta_{kx} \\ 1 & 1 \end{vmatrix}_{\ell}$$

$$+ {}^{\ell}J_{33}\; {}^{\ell}\delta_{jz} \begin{vmatrix} \delta_{kx} & \delta_{ky} \\ \delta_{ix} & \delta_{iy} \end{vmatrix}_{\ell} + {}^{\ell}J_{44} \begin{bmatrix} d_{ix} \\ d_{iy} \\ d_{iz} \end{bmatrix}_{\ell} \cdot \begin{bmatrix} \begin{vmatrix} \delta_{jy} & d_{ky} \\ \delta_{jz} & d_{kz} \end{vmatrix} \\ \begin{vmatrix} \delta_{jz} & d_{kz} \\ \delta_{jx} & d_{kx} \end{vmatrix} \\ \begin{vmatrix} \delta_{jx} & d_{kx} \\ \delta_{jy} & d_{ky} \end{vmatrix} \end{bmatrix}_{\ell}$$

$$+ {}^{\ell}J_{14} \; {}^{\ell}\delta_{iz} \begin{vmatrix} \delta_{jz} & d_{kz} \\ \delta_{jx} & d_{kx} \end{vmatrix}_{\ell} + {}^{\ell}\delta_{jx} \begin{vmatrix} \delta_{ky} & d_{iz} \\ -\delta_{kz} & d_{iy} \end{vmatrix}_{\ell} + {}^{\ell}\delta_{iy} \begin{vmatrix} \delta_{jy} & d_{ky} \\ \delta_{jx} & d_{kx} \end{vmatrix}_{\ell}$$

$$- {}^{\ell}d_{ix} \begin{vmatrix} \delta_{jy} & -\delta_{kz} \\ \delta_{jz} & \delta_{ky} \end{vmatrix}_{\ell}$$

$$+ {}^{\ell}J_{24} \left( {}^{\ell}\delta_{ix} \left\{ \begin{vmatrix} \delta_{jz} & d_{kz} \\ -\delta_{jy} & d_{ky} \end{vmatrix}_{\ell} \begin{vmatrix} \delta_{jx} & d_{kx} \\ \delta_{jy} & d_{ky} \end{vmatrix}_{\ell} \right\} + {}^{\ell}\delta_{jy} \begin{vmatrix} \delta_{kx} & d_{iz} \\ -\delta_{kz} & d_{ix} \end{vmatrix}_{\ell} \right.$$

$$\left. - {}^{\ell}d_{iy} \begin{vmatrix} \delta_{jx} & -\delta_{kz} \\ \delta_{jz} & \delta_{kx} \end{vmatrix}_{\ell} \right)$$

$$+ {}^{\ell}J_{34} \left( {}^{\ell}\delta_{iy} \begin{vmatrix} \delta_{jy} & d_{ky} \\ \delta_{jz} & d_{kz} \end{vmatrix}_{\ell} + {}^{\ell}\delta_{jz} \begin{vmatrix} \delta_{kx} & d_{iy} \\ -\delta_{ky} & d_{ix} \end{vmatrix}_{\ell} + \delta_{ix} \begin{vmatrix} \delta_{jx} & d_{kx} \\ \delta_{jz} & d_{kz} \end{vmatrix}_{\ell} \right.$$

$$\left. - {}^{\ell}d_{iz} \begin{vmatrix} \delta_{jy} & \delta_{kx} \\ -\delta_{jx} & \delta_{ky} \end{vmatrix}_{\ell} \right)$$

For a 6-dof robot arm at a given set of position variables, 216 elements of the matrix $[P_{ijk}]$ should be evaluated, but due to symmetry and other simplifying elements [3,26] such as,

1. $P_{ijk} = -P_{kji}$, $i,k \geq j$ (reflexive coupling)

2. $P_{ijk} = P_{ikj}$

were used to reduce the number to 56 elements.
Computational results are given in section 5.

## 4. The Load and No-Load Conditions:

### 4.1 Problem:

The previously derived symbolics will be used in the following discussion to re-emphasis the importance of the coriolis and centripetal forces in effecting the dynamic performance of robot manipulators under different conditions.

In the case of no-load the previous equations can be used directly. In case of load conditions the problem becomes much more complicated because of the dynamic interaction between the load and the arm [33,34,35,36,39]

It's well known that the last link of a robot arm contributes the most complex configurational dynamics, and with a load in the hand the problem will get complicated. In our study and for simplicity reasons the load will be assumed to be a cube which fits exactly in the hand with its centre of mass at the origin of the hand. This assumption will alter only the pseudo-inertia matrix of the last $\ell$ink ($\ell$ink n). In case of eq. (3) and eq. (4) the pseudo-inertia matrix will change to,

$$
{}^{n}J_{ij} = (m_n + m_{load}) \begin{vmatrix} \frac{-K^2_{xx\ell}+K^2_{yy\ell}+K^2_{zz\ell}}{2} & 0 & 0 & x_\ell \\ 0 & \frac{K^2_{xx\ell}-K^2_{yy\ell}+K^2_{zz\ell}}{2} & 0 & y_\ell \\ 0 & 0 & \frac{K^2_{xx\ell}+K^2_{yy\ell}-K^2_{zz\ell}}{2} & z_\ell \\ x_\ell & y_\ell & z_\ell & 1 \end{vmatrix} \quad (18)
$$

and for eq. (10), the mass of $\ell$ink (n) will be $(m_n + m_{load})$.

### 4.2 Quantification of Neglecting Coriolis/Centripetal Forces:

The error that results from neglecting the coriolis and centripetal effects was computed under different speeds and pay-loads.

The formula used to compute the error is given by,

$$
\text{Error} = \left| \frac{f_2 - f_1}{f_2} \right| \times 100\%
$$

Where

$f_2$ : joint forces including coriolis and centripetal effects.

$f_1$ : joint forces excluding coriolis and centripetal effects.

$|\cdot|$ : absolute value.

Two robot models were selected, the stanford arm [1,2,3,26] and the PUMA 560 [37,38] to give a broader range of data.

As an example, each joint position parameter is chosen as $\Theta_i = 0.4$ radians ($i = 1,2,\ldots\ldots,6$) for all the cases considered in our examples. The resulting joint forces and torques were calculated for the different joint velocities, accelerations and loads. The velocities, accelerations and loads were chosen to give realistic simulation results, whilst maintaining consistency with the existing robot models.

In the case of load, the pseudo-inertia matrix of the two robot models will have the following form:

$$^6J_{ij}(\text{stanford}) = (0.51 + m_{load}) \begin{bmatrix} 0.00059 & 0 & 0 & 0 \\ 0 & 0.00059 & 0 & 0 \\ 0 & 0 & 0.0516 & 0.1554 \\ 0 & 0 & 0.1554 & 1 \end{bmatrix}$$

$$^6J_{ij}(\text{PUMA 560}) = (0.09 + m_{load}) \begin{bmatrix} 0.000444 & 0 & 0 & 0 \\ 0 & 0.000444 & 0 & 0 \\ 0 & 0 & 0.0029 & 0.032 \\ 0 & 0 & 0.032 & 1 \end{bmatrix}$$

All the required data for the dynamics simulation can be found in published literature. For the stanford arm the data is in [1] and the PUMA 560 in [37] .

The result of the simulation are given in the next section.

## 5. Computational Results:

### 5.1 Execution time of the Simplified Symbolics:

Symbolic representation can lead to a better understanding and simplification of robot dynamic equations. This will enhance the development of efficient computational algorithms and programs [29,30,31].

A very efficient FORTRAN program was written to test the computing time of the symbolics derived in this paper. The model choosen was a 6-dof stanford arm and the computation time was calculated in two different ways, i.e.

(I) Including the cost of computing all the pre-required terms such as $H_i, \partial H_{i/q_i}$ etc.

(II) Using the argument of Hollerbach [4] to exclude the computing time of the terms mentioned in (I) because of their dependency on the arm configuration.

The program was executed on a SUN workstation (32), and computation time are recorded in table 1.

| | average CPU time (sec) |
|---|---|
| I | 0.208 |
| II | 0.147 |

table 1. CPU execution time

## 5.2 Quantification results and error analysis:

A set of compiled data of two robot arms (stanford, PUMA 560) was produced to study the dynamic performance of a robot manipulator when subject to different pay loads and speeds. The error committed at each joint when coriolis and centripetal effects are neglected was calculated. The results are recorded in table 2,3,4,

case (1) :

Load = 0.0 Kg (no-load)

| joint velocity & acceleration $\dot{\theta}_i$ (rad/s) $\ddot{\theta}_i$ (rad/s$^2$) | Error in calculating joint forces (%) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Stanford | | | | | | PUMA 560 | | | | | |
| | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ | $E_6$ | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ | $E_6$ |
| 0.5 | 15 | 2.6 | 0.5 | 60.8 | 7 | 0.72 | 11.06 | 8 | 1.07 | 2.07 | 0.713 | 0.7 |
| 1.0 | 35.4 | 7.07 | 2.1 | 52 | 22 | 1.45 | 25 | 17.3 | 4 | 18.6 | 1.4 | 1.56 |
| 1.5 | 65 | 11.8 | 5 | 57 | 37 | 2.2 | 43 | 28.4 | 8.3 | 280 | 2.12 | 2.5 |
| 2.0 | 109.7 | 16.3 | 9 | 62 | 50 | 3 | 66.2 | 42 | 13.4 | 61.7 | 2.84 | 3.4 |
| 2.5 | 190 | 20.5 | 14.3 | 66 | 59 | 3.7 | 99.2 | 58.4 | 18.9 | 54 | 3.6 | 4.3 |
| 3.0 | 364 | 24.4 | 20.7 | 70 | 66 | 4.5 | 150 | 80 | 24.5 | 55.1 | 4.3 | 5.24 |

Table 2.

Case (2) :

Load = 1.0 Kg

| joint velocity & acceleration $\dot{\theta}_i$ (rad/s) $\ddot{\theta}_i$ (rad/s$^2$) | Error in calculating joint forces (%) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Stanford | | | | | | PUMA 560 | | | | | |
| | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ | $E_6$ | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ | $E_6$ |
| 0.5 | 12.8 | 4 | 0.9 | 138 | 7.15 | 2 | 10.7 | 8 | 1.14 | 56 | 9 | 3 |
| 1.0 | 23 | 11.6 | 3.7 | 106.4 | 22.7 | 4.08 | 23.9 | 17.24 | 4.3 | 292.4 | 13.9 | 8.64 |
| 1.5 | 30.6 | 19.7 | 8.52 | 88 | 39 | 6.24 | 40.63 | 28.3 | 9 | 357 | 20.23 | 17.4 |
| 2.0 | 37 | 27.2 | 15.1 | 85 | 52 | 8.5 | 63 | 42 | 14.2 | 954 | 27.5 | 30 |
| 2.5 | 42.3 | 34 | 23 | 86 | 62 | 10.9 | 93 | 58.13 | 20 | 2140 | 36 | 44.4 |
| 3.0 | 47 | 39.5 | 32 | 86.3 | 69 | 13.3 | 137 | 80 | 26 | 599 | 45.3 | 64.6 |

Table 3.

Case (3) :

Load = 2.0 Kg

| joint velocity & acceleration $\dot{\theta}_i$ (rad/s) $\ddot{\theta}_i$ (rad/s$^2$) | Error in calculating joint forces (%) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Stanford | | | | | | PUMA 560 | | | | | |
| | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ | $E_6$ | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ | $E_6$ |
| 0.5 | 30.3 | 4.5 | 1.2 | 71 | 7.2 | 3.13 | 10.3 | 8 | 1.2 | 27.2 | 67.3 | 3 |
| 1.0 | 46.5 | 13.7 | 5 | 153 | 23 | 6.5 | 23.06 | 17.2 | 4.5 | 99 | 41.3 | 10.8 |
| 1.5 | 56.6 | 23.4 | 11.08 | 104 | 39.2 | 10 | 39 | 28.2 | 9.2 | 370 | 57 | 23.7 |
| 2.0 | 64 | 32.3 | 19.2 | 96 | 52.5 | 13.8 | 60 | 42 | 15 | 1358 | 80 | 43.4 |
| 2.5 | 69 | 40 | 29 | 94 | 62 | 18 | 88.2 | 58 | 20.8 | 368 | 112 | 74.2 |
| 3.0 | 72.3 | 46.2 | 38.3 | 93 | 69.8 | 22.3 | 128 | 78.6 | 27 | 250 | 160 | 126.2 |

Table 4.

The previous three cases were plotted to show the unpredictable non-linear changes in the dynamic performance, as shown in Fig 1,2,3.

Concluding Remarks:

The dynamic equations used to model robot manipulators consist of three types of effects of equal importance, the inertial, the coriolis and the centripetal, and the gravity terms. A simplified symbolic representation based on the lagrangian for the dynamics has been presented. FORTRAN programs were written to verify the derivation and computational time for a 6-DOF manipulator which are recorded in table 1. The derived symbolics were used to perform an error analysis study, and for the first time quantified results have been produced to measure the effect of neglecting the coriolis and centripetal terms on the dynamic performance of robot manipulators under different payloads and speeds. The analysis has been performed on both the Stanford and PUMA 560 arms and a set of compiled numerical data is presented in tables (2,3,4). Graphical representation of the data is given in Fig (1,2,3) to help in further illustration of the results.

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, no.11, pp730-736.

5. Lee C S, Lee B H, Nigam R, 1983, "Development of the Generalized D'Alambert Equations of Motion for Mechanical Manipulators," Proc. 22nd Conf Decision and Control, San Antonio, Tex, Dec. 14-16, pp. 1205-1210.

6. Orin D E, McGihee R B, Vukobratovic M, Hartoch G, 1979, "Kinematic and Kinetic Analysis of Open-Chain Linkages Utilizing Newton-Euler Methods," Math. Biosci., vol 43, pp 107-103.

7. Luh J Y S, et al., "On-Line Computational Scheme for Mechanical Manipulators," Trans. ASME J, Dynamic Systems, Meas., and Control, 102, pp 69-76.

8. 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.

9. 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, vol 2 pp 1343-1346.

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 pp. 60-70.

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, pp. 3-21 .

14. Lee C S, Chang P R, 1986, "Efficient Parallel Algorithm for Robot Inverse Dynamics Computation," IEEE Trans, Syst., Man and Cybernetics, vol. smc. 16, no. 4.

15. Binder E E, Herzog J H, 1986, "Distributed Computer Architecture and Fast Parallel Algorithms in Real-Time Robot control," IEEE Trans. Syst., Man, and Cybernetics, vol. smc. 16, no. 4.

16. 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. smc 12, pp. 214-234.

17. Lathrop L H, 1983, "Parallelism in Manipulator Dynamics," MIT Artificial Intelligance Lab., Cambridge, MA, Tech. Rep. No. 754.

18. Turney J L, et. al., 1982, "Connection Between Formulations of Robot Arm Dynamics with Applications to Simulation and Control," CRIM Technical Report no. RSD-TR-4-82, the University of Michigan, Ann Arbor, Michigan.

19. Lee C S G, 1983, "On the Control of Robot Manipulators," Proc. 27th of the Society of Photo-Optical Instrumentation Engineers, vol. 442, San Diego, pp. 58-83.

20. Neuman C P, Tourassis V D, 1983, "Robot Control Issues and Insight," Proc. of the 3rd Yale workshop on Applications of Adaptive Systems Theory, pp. 179-189.

21. Tourassis V D , Neuman C P, 1985, "Properties and structure of Dynamic Robot Models for Control Engineering Applications," Mechanism and Machine Theory, vol. 20, no. 1, pp. 27-40.

22. Bejczy A K, Paul R P, 1981, "Simplified Robot Arm Dynamics for Control," proc. 20th IEEE conf. Decision and Control, San Diego, pp. 261-262.

23. Luh J Y S, Lin C S, 1981, "Automatic Generation of Dynamic Equations for Mechanical Manipulators," proc. of Joint Automatic Control conf. Charlottesville, pp. TA-2D.

24. Lin C S , Chang P R, 1984, " Automatic Dynamics Simplification for Robot Manipulators," proc. ofthe 23rd conf. on Decision and Control Las Vegas, pp. 752-759.

25. Megahed S, Renaud M, 1982, "Minimization of the Computation Time Necessary for the Dynamic Control of Robot Manipulators," proc. 12th conf. on industrial robot technology 6th int. Symp. Industrial Robots Paris, pp. 469-478.

26. Lewis R A, 1974, "Autonomous Manipulation on a Robot: Summary of Manipulator Software Functions," Technical Memo. 33-679, Jet Propulsion Laboratory, Pasadena, California.

27. Neuman C P, Murray J J, 1987, "Customized Computational Robot Dynamics," J Robotic Systems, 4(4), pp. 503-526.

28. Zomaya A Y, Morris A S, 1988, "On the Complexity Reduction of the Coriolis and Centripetal Effects of a 6-DOF Robot Manipulator," Robot Control: Theory and Applications, edited by K Warwick and A Pugh, published by Peter Peregrinus Ltd. pp. 71-81.

29. Murray J J, Neuman C P, 1984, "ARM: An Algebric Robot Dynamic Modeling Program," Proc. of 1st International conf. on Robotics, Atlanta, pp. 103-113.

30. Koplik J, Leu M C, 1986, "Computer Generation of Robot Dynamics. Equations and Related issues, "J Robotic Systems, 3(3), pp. 301-319.

31. Faessler H, 1986, "Computer Assisted Generation of Dynamic Equations for Mupltibody Systems," Int. J Robotics Res., vol. 5, no. 3, pp. 129-141.

32. FORTRAN Programmer's Guide for the SUN Workstation and the Floating-Point Programmer's Guide for the SUN Workstation, SUN Microsystem, Inc., 2550 Garcia Avenue, Mountain View, CA 94043.

33. Izaguirre A, Paul R P, 1985,"Computation of the Inertial and Gravitational Co-efficients of the Dynamics Equations for a Robot Manipulator with a Load," Proc. of IEEE int. conf. on Robotics and Automation, ST. Lousis, Missouri, pp. 1024-1032.

34. Shin K G, McKay N D, 1987, "Robust Trajectory Planning for Robotic Manipulators Under Payload Uncertiiainties," IEEE Trans. on Automatic Control, vol. 32, no. 12, Dec., pp. 1044-1054.

35. Wang L T, Ravani B, 1988, "Dynamic Load Carrying Capacity of Mechanical Manipulators - Part I: Problem Formulation," Trans. ASME J. Dynamic Sys., Meas., and Control, vol. 110, March, pp. 46-52.

36. Wang L T, Ravani B, 1988, "Dynamic Load Carrying Capacity of Mechanical Manipulators - Part II : Computational Procedure and Applications," Trans, ASME J Dynamic Sys., Meas., and Control, vol. 110, March, pp. 53-61.

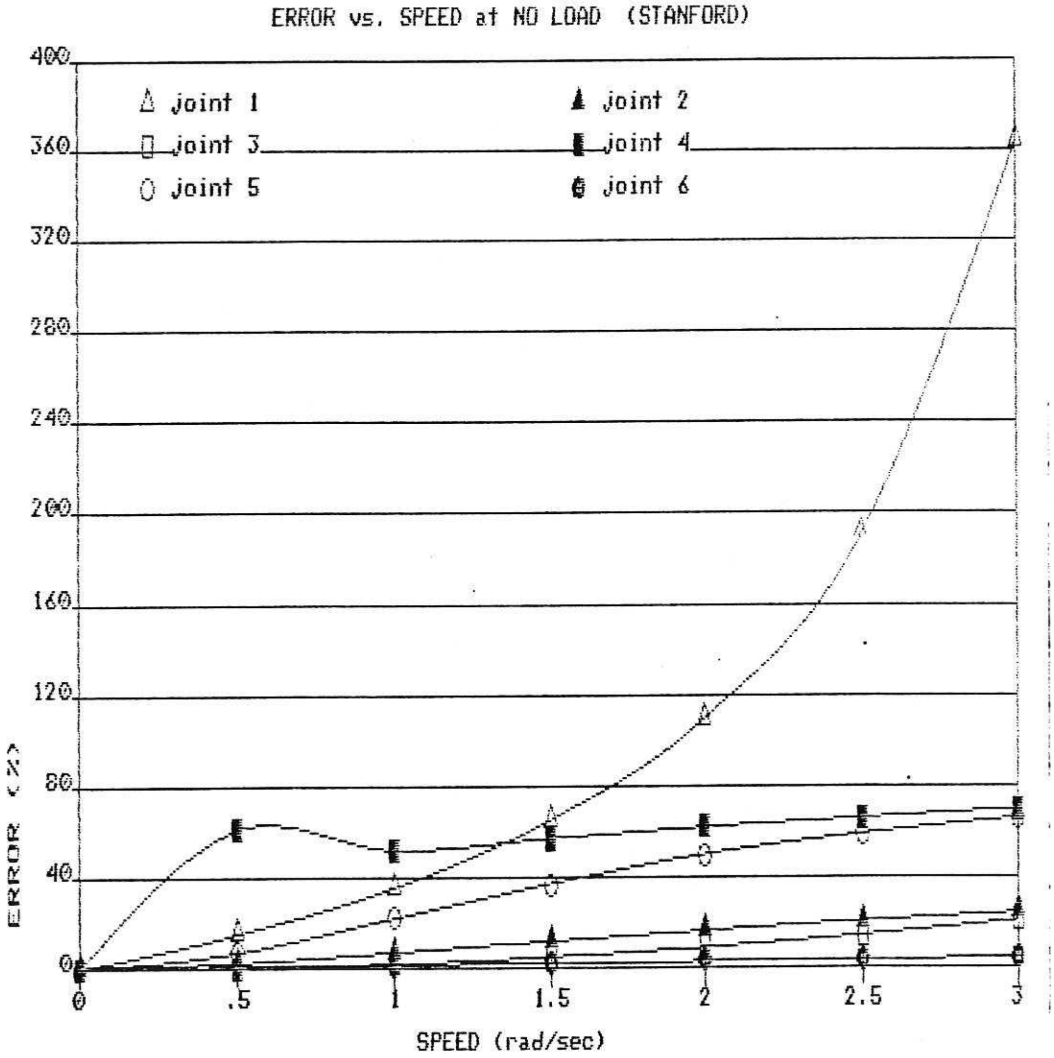
37. Armstrong B, Khatib O, Burdick J, 1986, "The Explicit Dynamic Model and Inertial Parameters of the PUMA 560 Arm," proc. of IEEE Robotics and Automation Conf., pp. 510-518.

38. Paul R P, Rong M, Zhang H, 1983, "The Dynamics of the PUMA Manipulator," Proc. of the American Control Conference, San Francisco, CA., pp. 491-496.

39. Coiffet P, 1983, "Robot Technology Volume 2: Interaction with the Environment," Kogan Page Ltd.

Fig. 1 Case 1

(a)


ERROR vs. SPEED at NO LOAD (STANFORD)
joint 1
joint 2
joint 3
joint 4
joint 5
joint 6
ERROR (%)
SPEED (rad/sec)
0
40
80
120
160
200
240
280
320
360
400
.5
1
1.5
2
2.5
3

(b)

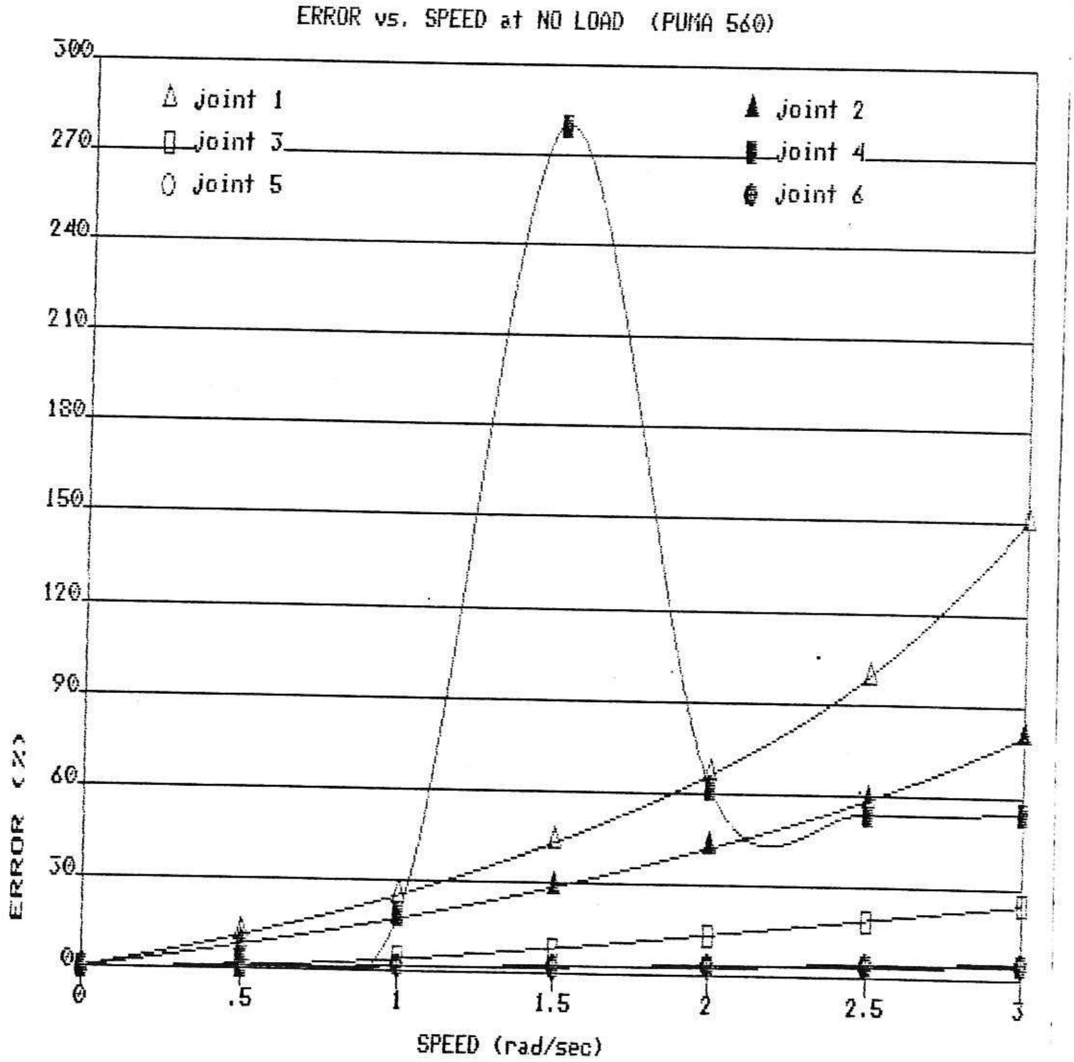
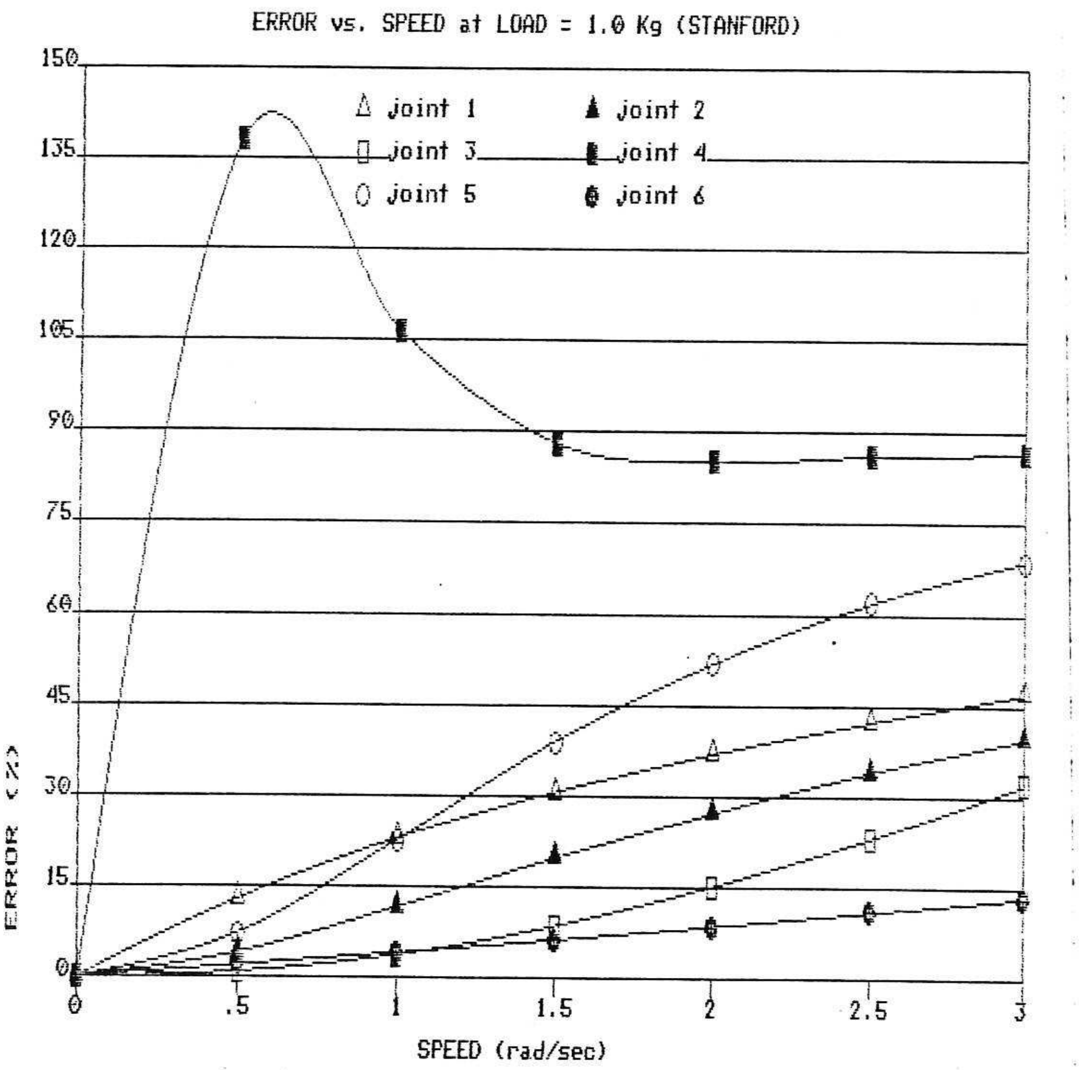

ERROR vs. SPEED at NO LOAD (PUMA 560)
joint 1
joint 3
joint 5
joint 2
joint 4
joint 6
300
270
240
210
180
150
120
90
60
30
0
ERROR (%)
0
.5
1
1.5
2
2.5
3
SPEED (rad/sec)

Fig. 2 Case 2

(a)


ERROR vs. SPEED at LOAD = 1.0 Kg (STANFORD)
joint 1
joint 2
joint 3
joint 4
joint 5
joint 6
150
135
120
105
90
75
60
45
30
15
0
ERROR (%)
0
.5
1
1.5
2
2.5
3
SPEED (rad/sec)

(b)

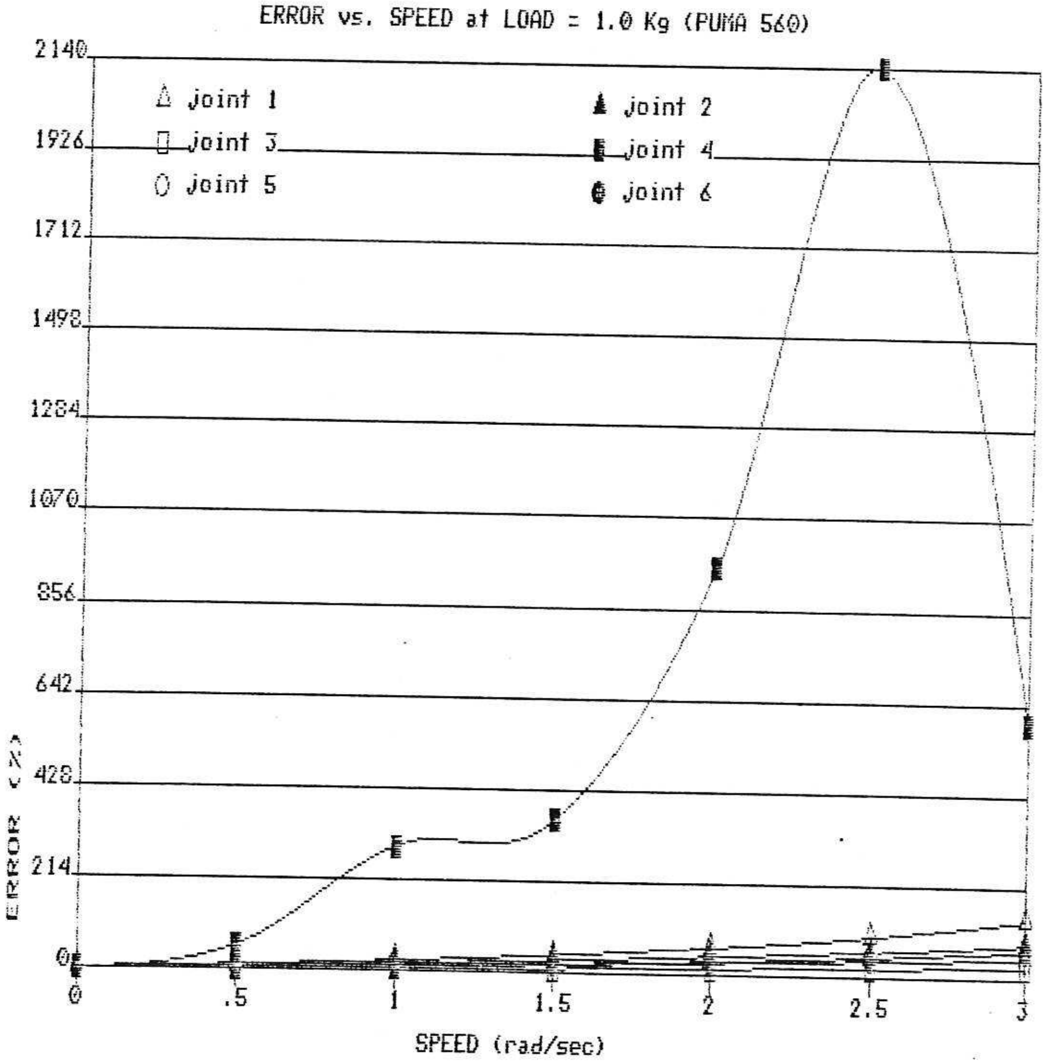
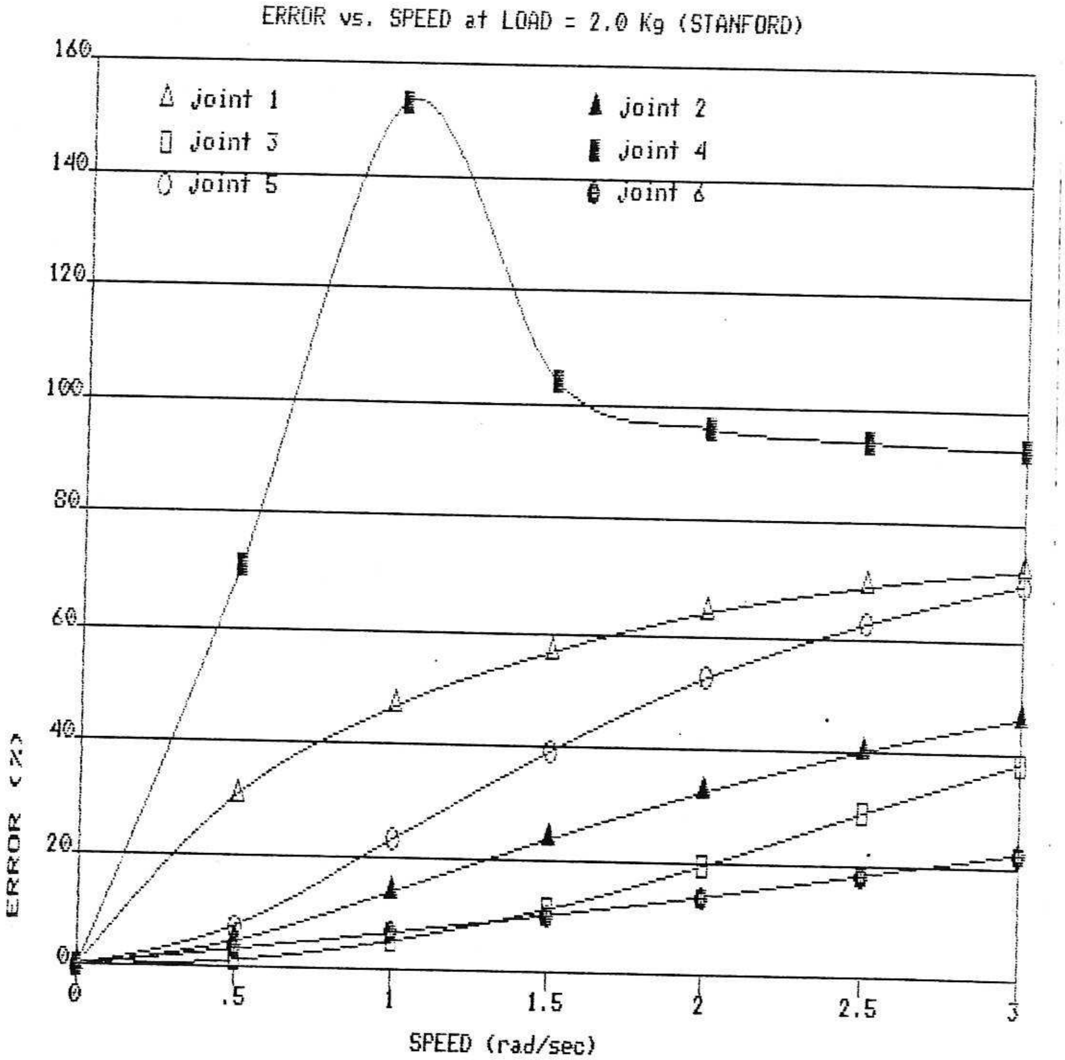
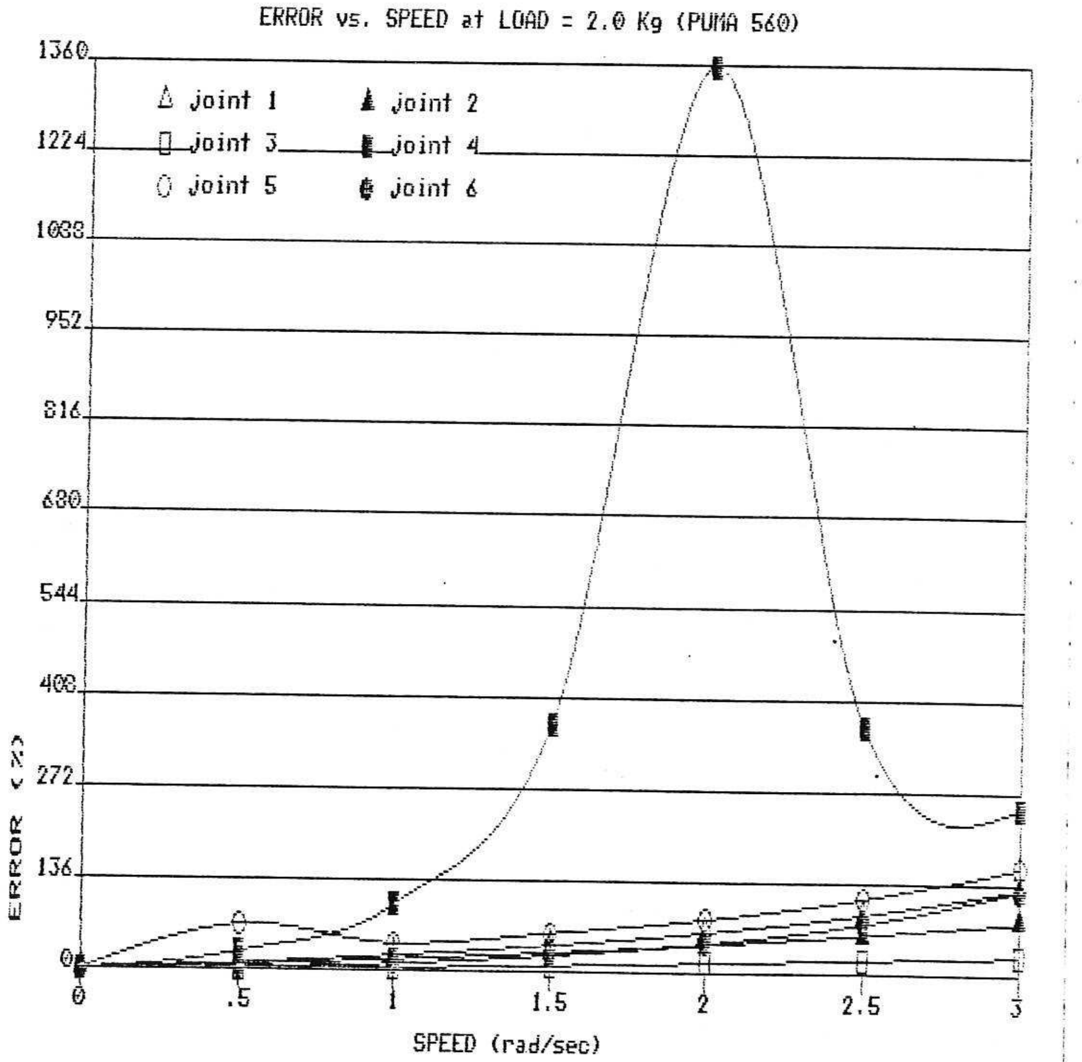

ERROR vs. SPEED at LOAD = 1.0 Kg (PUMA 560)
2140
1926
1712
1498
1284
1070
856
642
428
214
0
ERROR (%)
joint 1
joint 2
joint 3
joint 4
joint 5
joint 6
0
.5
1
1.5
2
2.5
3
SPEED (rad/sec)

Fig. 3. Case 3.

(a)


ERROR vs. SPEED at LOAD = 2.0 Kg (STANFORD)
joint 1
joint 2
joint 3
joint 4
joint 5
joint 6
160
140
120
100
80
60
40
20
0
ERROR (%)
0
.5
1
1.5
2
2.5
3
SPEED (rad/sec)

(b)



ERROR vs. SPEED at LOAD = 2.0 Kg (PUMA 560)
joint 1
joint 2
joint 3
joint 4
joint 5
joint 6
1360
1224
1088
952
816
680
544
408
272
136
0
ERROR (%)
0
.5
1
1.5
2
2.5
3
SPEED (rad/sec)