COMPARISON OF JOINT SPACE VERSUS TASK FORCE LOAD DISTRIBUTION OPTIMIZATION FOR A MULTIARM MANIPULATOR SYSTEM

Donald I. Soloway
NASA Langley Research Center
Hampton, Virginia 23665-5225

Thomas E. Alberts
Old Dominion University
Norfolk, Virginia

Abstract

It is often proposed that the redundancy in choosing a force distribution for multiple arms grasping a single object should be handled by minimizing a quadratic performance index. The performance index may be formulated in terms of joint torques or in terms of the Cartesian space force/torque applied to the body by the grippers. The former seeks to minimize power consumption while the latter minimizes body stresses. Because the cost functions are related to each other by a joint angle dependent transformation on the weight matrix, it might be argued that either method tends to reduce power consumption, but clearly the joint space minimization is optimal. In this paper, a comparison of these two options is presented with consideration given to computational cost and power consumption. Simulation results using a two arm robot system are presented to show the savings realized by employing the joint space optimization. These savings are offset by additional complexity, computation time and in some cases processor power consumption.

1. Introduction

Some of the recent developments in multiple arm manipulation of a commonly grasped body include the work of Hayati [1], Alberts [2], and Carnignan [3]. The common thread between these papers is that each employs the minimization of some form of quadratic performance index to choose an appropriate load distribution. Hayati proposed an extension of Mason's [4] hybrid position/force control in which the inertia of each arm is artificially extended to include a portion of the payload inertia. From a practical point of view, the method may be difficult to implement effectively, because it requires precise knowledge of the inertial properties of the arms and of the jointly manipulated objects as well as the solution of inverse dynamics. Alberts closes a force feedback loop around a kinematic resolved rate controller for multiple arms, thus realizing the Damping Control Method. As in Hayati's work the problem of redundancy in determining the distribution of load among the manipulators is handled by minimizing a quadratic cost function in task-space force and torque. This tends to minimize internal forces in the body while maintaining control over a prescribed force and torque interaction with the external environment. Alberts formulation does not consider the closed chain dynamics of the manipulators and

payload, but rather each manipulator is viewed as an actuator with an independent control system. Carnignan used a quadratic cost function to minimize joint space torques including those due to manipulator kinetics, but due to the inclusion of the manipulator kinetic effects, undesirable internal forces may be produced in the body. This method is computationally more expensive than task space optimization.

It can be argued that minimizing a quadratic cost function in joint space has greater power efficiency than a minimization in task space, but at what computational cost? This paper compares the task space versus joint space cost functions with respect to power efficiency and computational cost. The development is to be based on Alberts' task space cost function and a joint space cost function developed along similar lines. The computational cost of using the joint space minimization scheme is similar to Carnignan's method. The torques required to compensate for manipulator kinetics are not included in the minimization so as to avoid imposing unnecessary internal forces and torques within the manipulated body.

2. Power Cost Calculation

In a multiarm robotic system, minimization of a joint torque based quadratic cost function would tend to minimize the dissipated power used by the motors under static conditions.[1]

A minimization scheme to establish an "optimal" force distribution could be formulated in either joint space or task space. From the standpoint of power used to drive the joints, it would be optimal to formulate the minimization in joint space. In this paper the question considered is that of how much power really can be saved by optimizing in joint space as opposed to task space.

Based upon a task space quadratic performance index

$$Q_t = \underline{\Gamma}_t^T W_t \underline{\Gamma}_t \tag{1}$$

the task space optimal load distribution force equation [2] is

$$\underline{\Gamma}_t = \Lambda_t \underline{F} \tag{2}$$

where

$$\Lambda_t = W_t^{-1} H^T [H W_t^{-1} H^T]^{-1} \tag{3}$$

and $\underline{\Gamma}_t$ is a vector of wrench vectors (as in Equation 15, Appendix I) such that each components $\underline{F}(i)$ of $\underline{\Gamma}_t$ is a wrench vector applied to the body at the grasp point of the i^{th} manipulator and subscript t denotes task space. H is a matrix of Jacobian transformations that depends upon the locations of the grasp points of each arm relative to the applied external force $\underline{F}$. The weight

[1]To truly minimize power consumption under dynamic conditions the cost function should include a term representing the product of joint torque and joint velocity. This is discussed further later in reference to Equation (9).

matrix is given by $W_t = \text{diag}\ (v_1, v_2, \ldots, v_N)$ where N is the number of arms and v_i is a diagonal weight matrix reflecting the relative cartesian end effector force and torque capabilities of the i^{th} arm.

Using a similar development (Appendix I) but, based upon a joint space quadratic performance index

$$Q_j = \tau^T W_j \tau \tag{4}$$

the joint space optimal load distribution force equation is

$$\underline{\Gamma}_j = \Lambda_j \underline{F} \tag{5}$$

where

$$\Lambda_j = J^{-T} W_j^{-1} J^{-1} H [H J^{-T} W_j J^{-1} H^{-T}]^{-1} \tag{6}$$

with $W_j = \text{diag}\ (w_1, w_2, \ldots, w_N)$ and w_i is a diagonal matrix whose elements are the squares of the reciprocals of the relative joint torque capabilities of the i^{th} arm. Subscript j denotes joint space.

The cost function is an indication of power used since torque (τ) is proportional to current and current squared is proportional to power dissipated. To compare power consumption on an equal basis the cost of task space optimized load distribution was evaluated in terms of the joint space cost function. Thus, for evaluative purposes, the torque-squared cost of executing the task space optimization scheme can be expressed as

$$Q_{tj} = \tau_t^T W_j \tau_t \tag{7}$$

where $\tau_t = J^T \Lambda_t \underline{F}$

and the cost associated with joint space optimization is given by Equation (4), that is $Q_{jj} = Q_j$. The cost resulting from the joint space formulation will always be less than that of the task space formulation according to the above criterion.

The power used by a motor is

$$\text{Power} = I_a^2 R + \frac{K_E}{K_T} \omega T_m + \frac{K_E}{K_T} \omega T_l \tag{8}$$

where I_a is the armature current
R is the armature resistance
K_E is the back emf constant
K_T is the torque constant
ω is the angular speed of the rotor
T_l is the torque of the load

$T_m = T_f + D_\omega$ where T_f is constant friction
and D is the viscous damping coefficient.

The current I_a is related to the dynamcs of the motor by

$$I_a = \frac{1}{K_T}\left[(J_m + J_l)\frac{d\omega}{dt} + T_m + T_l\right] \tag{9}$$

where J_m is the motor inertia and J_l is the load inertia. Observe that in the joint space minimization presented here (7) minimizes the power due to the term I_a^2R in (8) but does not account for the power associated with ωT_l in the last term of (8). From a practical point of view it appears that the contribution of this term will normally be small. In simulations conducted, in which the load velocity was 0.1 m/s, the ωT_l term resulted in power difference of less than 1%.

The percent difference in power is defined as follows

$$P_N = \frac{P_t - P_j}{P_j} * 100 \tag{10}$$

where P_t and P_j represent the power used by task and joint space respectively.

The power difference is

$$P_D = P_t - P_j \quad \text{where } P_D \text{ is in watts} \tag{11}$$

A simulation[2] was used with the system parameters of the NASA LTM[3] (given in Appendix III) to compute the power used by the two-arm LTM system in moving a 40 lb. payload on Earth. The center of gravity of the object was at the grasp point of one of the arms. The two arms moved horizontally without rotating the object. For this trajectory P_D and P_N were plotted in figures 1 and 2 respectively. The initial P_J = 1093 watts.

[2]The simulation is a combination of ROBSIM (a full dynamics robot simulator developed by NASA LaRC) and user coded. Robsim simulated the robotics dynamics and the user code calculated the load distributed joint torques and costs.

[3]LTM is an acronym for the Laboratory Telerobotic Manipulator developed for NASA LaRC by Oak Ridge National Laboratory. The LTM is a seven-degree-of-freedom arm employing differential friction drive joints. For simplicity the differential joint drive are treated as conventional gear driven joints.

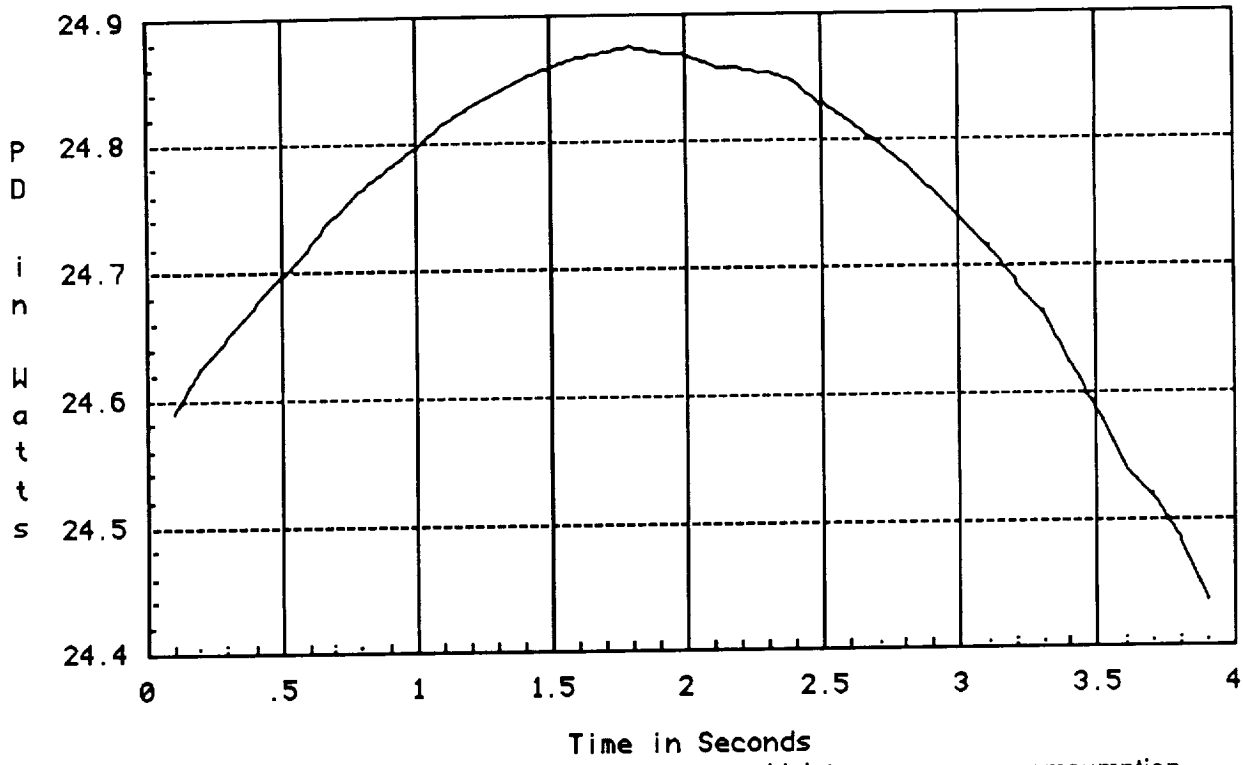

Figure 1. The difference between task space and joint space power comsumption.

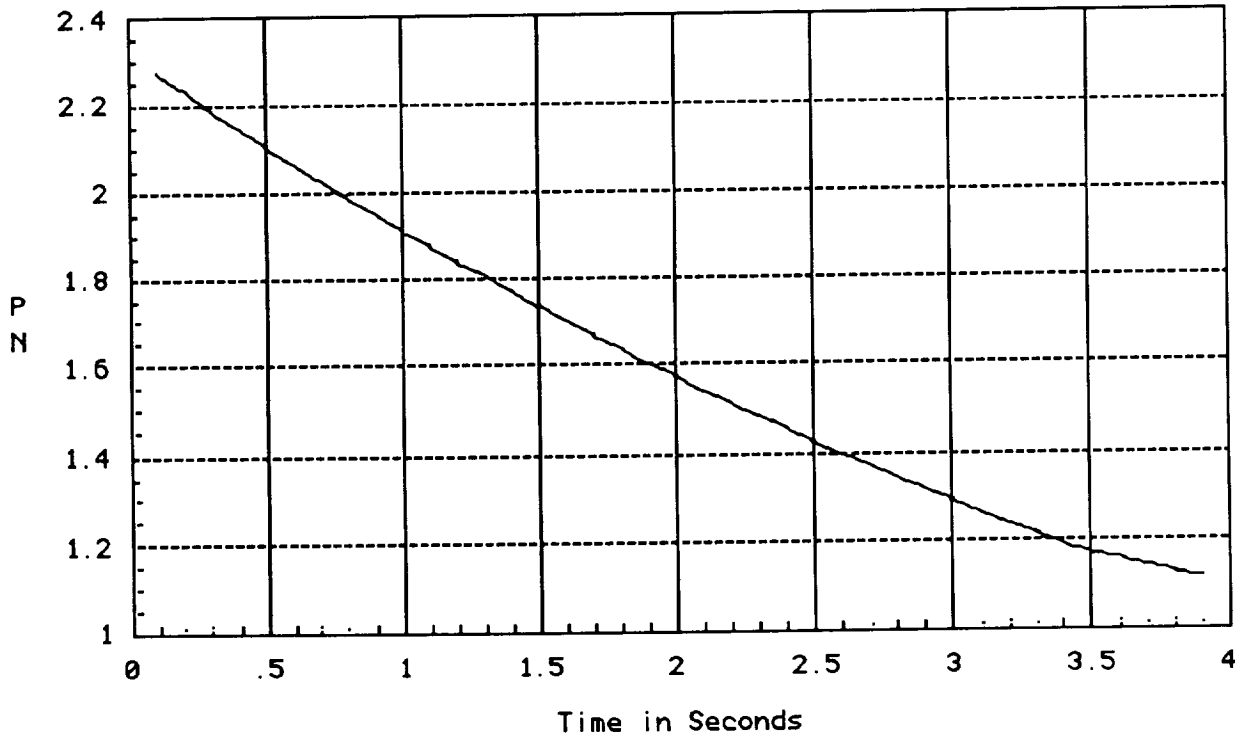

Figure 2. The percentage of power saved with respect to the total power used in joint space.

3. Computational Cost Calculation

In a real-time computer program it is desirable to keep computationally expensive operations to a minimum. The number of operations it takes to calculate task space and joint space load distribution are determined below.

For this analysis the following assumptions are made:

1. The manipulators grasp points are fixed while the body is in motion, thus H remains a constant matrix.
2. W does not change over the motion of the body.
3. All manipulators have the same number of degrees of freedom.

In task space Λ is constant thus requiring only a matrix vector multiply to calculate Γ_t. Λ is a 6N x 6 matrix where N is the number of manipulators grasping the body, and F is a 6 x 1 vector needing:

36N multiplies;
30N additions

to calculate Γ_t.

Joint space will have the same operations as above plus the operations to calculate Λ. The total number of operation needed to calculated τ_j as derived in Appendix II is:

N(48n + 72) + 195	multiplies
71	divides
N(48n + 30) + 206	additions

With the need to calculated the jacobian inverse the total number of operations to calculate Γ_j as derived in Appendix II is:

N(60n + 267) + 195	multiplies
71N + 71	divides
N(58n + 236) + 206	additions

Currently the most powerful space qualified procesor is the Harris 80C86. Based on using this processor with a 5 mhz clock and the number of arms and degrees of freedom (DOF), the results Table I, II, and III are obtained. The entries in the tables represent the number of times the computation can be executed in 1 second.

Degrees of Freedom	Number of Arms: 2	3	4
6	17.5	12.4	9.54
7	16.4	11.5	8.86
8	15.3	10.7	8.27

Joint Space with Jacobian Inverse Calculated
Table I

Degrees of Freedom	2	3	4
6	29.8	21.9	17.3
7	27.2	19.8	15.6
8	25.0	18.1	14.2

Joint Space without Jacobian Inverse Calculated
Table II

Degrees of Freedom	2	3	4
6	413	275	206
7	413	275	206
8	413	275	206

Task Space
Table III

The computational cost of task space optimization is small enough to execute the code on an existing processor whereas joint space optimization would likely require a separate processor. In this case the power required to operate additional processing equipment must be considered. The power requirement of an 64K 80C86 board based on Harris radiation hardened components is 30 watts.

4. Summary

It has been shown that an apparently significant amount of power can be saved by employing the joint space optimized load distribution. In the example presented the largest savings over task space optimization realized was about 25 watts. This can be viewed as an extreme case. It is important to note, however, that the joint space optimization represents a substantial increase in computational burden. If this results in a need for additional processors, the power required to operate them might offset the savings realized by the optimization scheme.

In the future, space qualified processors will be available that are much more powerful and possibly more power efficient than those now used. In the case where a surplus of efficient computing power is available, the joint space optimization may prove to be the method of choice.

5. Joint Space Optimal Load Distribution

Assume that a desired net cartesian space wrench vector $\underline{F}$ acting on the body with known point of application p is specified. The wrench vector is made up of cartesian space force $\underline{f}$ and moment $\underline{m}$ vectors such that

$$\underline{F} = \begin{bmatrix} \underline{f} \\ \underline{m} \end{bmatrix} \tag{12}$$

This wrench could be to counteract gravitational loading or inertial reactions due to body acceleration, or to apply a force to the external environment through the jointly manipulated object. Now consider a system of N manipulators, where $\underline{F}(i)$ denotes a cartesian space wrench applied to the jointly manipulated body by the end effector of arm i. The following conditions must be satisfied in order to establish equilibrium:

$$\underline{f} = \sum_{i=1}^{N} \quad (13) \qquad \text{and} \qquad \underline{m} = \sum_{i=1}^{N} \left(\underline{m}(i) + \underline{r}_p(i) \times \underline{f}(i) \right) \tag{14}$$

where $\underline{f}(i)$ and $\underline{m}(i)$ are the force and moment vectors, respectively, that make up $\underline{F}(i)$

$$\underline{F}(i) = \begin{bmatrix} \underline{f}(i) \\ \underline{m}(i) \end{bmatrix} \tag{15}$$

and $\underline{r}_p(i)$ is a vector drawn from the point of application p of the force $\underline{f}$ to the grasp point for manipulator i. The minimization procedure operates on a quadratic function Q in the vector τ, with positive definite symmetric weighting matrix W

$$Q = \tau^T W \tau \tag{16}$$

where

$\tau = J^T \Gamma$, J^T is the jacobian transpose of the manipulator

$$\underline{\Gamma} = \begin{bmatrix} \underline{F}(1) \\ \vdots \\ \underline{F}(N) \end{bmatrix} \quad \text{and} \quad W = \text{diag}\,(w_1, w_2, \ldots, w_N) \tag{17}$$

Conditions (13) and (14) are expressed in matrix form as

$$\left[\begin{array}{c|c|c|c|c|c} I_3 & 0 & I_3 & 0 & \cdots & I_3 \quad 0 \\ \hline [r_p(1)X] & I_3 & [r_p(2)X] & I_3 & \cdots & [r_p(N)X] \quad I_3 \end{array}\right] \left[\begin{array}{c} F(1) \\ \vdots \\ F(N) \end{array}\right] = F \tag{18}$$

where, I_3 is a 3 by 3 identity matrix and $[\underline{r}_p(i)X]$ is a matrix that operates on $\underline{f}(i)$ to form the cross product $\underline{r}_p(i)X\ \underline{f}(i)$. A more compact expression for (18) is

$$HJ^T \tau = \underline{F} \tag{19}$$

Using a Lagrange multiplier λ to append equation (19) to (16) the augmented cost function is obtained.

$$\tilde{Q} = \tau^T W \tau - \lambda^T (HJ^{-T}\tau - \underline{F}) \tag{20}$$

The optimal solution must satisfy

$$\frac{\partial \tilde{Q}}{\partial \tau} = 0, \text{ that is } \quad 2\tau^T W - \lambda^T H J^{-1} = 0 \tag{21}$$

Rearranging and applying equation (19) to equation (21)

$$\tau = \frac{1}{2} HW^{-1}J^{-1}H^T\lambda \text{ which yields } \quad \lambda = 2(HJ^{-T}W^{-1}J^{-1}H^T)^{-1}\underline{F} \tag{22}$$

Now upon eliminating λ between equations (21) and (22) the final expression for joint torques is obtained.

$$\tau = W^{-1}J^{-1}H^T(HJ^{-T}W^{-1}J^{-1}H^T)^{-1}\underline{F} = \Lambda\underline{F} \tag{23}$$

Appendix II

To count the operations in Λ it is necessary to first simplify Λ into two matrices A and B, such that

where $\quad \Lambda = AB^{-1}, \quad A = J^{-T}W_j^{-1}J^{-1}H^T \quad$ and $\quad B = HA$

Calculating A:

$$\left[\begin{array}{cccc} J_1^{-1} & & & 0 \\ & J_2^{-1} & & \\ & & \ddots & \\ 0 & & & J_N^{-1} \end{array}\right], \quad W_j^{-1} = \left[\begin{array}{cccc} W_1^{-1} & & & 0 \\ & W_2^{-1} & & \\ & & \ddots & \\ 0 & & & W_N^{-1} \end{array}\right] \text{ and } \quad H^T = \left[\begin{array}{c} H_1^T \\ \vdots \\ H_N^T \end{array}\right]$$

where J_i^{-1} is the jacobian inverse for manipulator i,

W_i^{-1} is the weight matrix inverse for manipulator i,

and H_i^T is the transform grasp point to controlled force F for manipulator i.

so

$$A = J^{-T}W_j^{-1}J^{-1}H^T = \begin{bmatrix} J_1^{-1}W_1^{-1}J_1^{-T}H_1^T & & 0 \\ & \ddots & \\ 0 & & J_N^{-1}W_N^{-1}J_N^{-T}H_N^T \end{bmatrix}$$

and

$$B = HA = \begin{bmatrix} H_1J_1^{-1}W_1^{-1}J_1^{-T}H_1^T & & 0 \\ & \ddots & \\ 0 & & H_1J_N^{-1}W_N^{-1}J_N^{-T}H_N^T \end{bmatrix}$$

The cost to calculate $J_i^{-T}W_i^{-1}J_i^{-1}H_i^T$ is:

$$J_i^{-1}H_i^T = \begin{bmatrix} n \times 6 \\ \text{full} \\ \text{matrix} \end{bmatrix} \begin{bmatrix} 1 & 0 & o & z & -y \\ & 1 & -z & o & x \\ 0 & & y & -x & o \\ & & 1 & & 0 \\ & 0 & & 1 & \\ & & 0 & & 1 \end{bmatrix}$$

at a cost of n*(6 multiplies + 6 additions)

$$W_i^{-1}[J_i^{-1}H_i^T] = \begin{bmatrix} n \times 6 \\ \text{full} \\ \text{matrix} \end{bmatrix}$$

at a cost of n*6 multiplies

$$J_i^{-T}[W_i^{-1}J_i^{-1}H_i^T] = \begin{bmatrix} 6 \times 6 \\ \text{full} \\ \text{matrix} \end{bmatrix}$$

at a cost of n36 multiplies + (n-1)36 additions. The total cost of A is N[48n multiplies + (48n-36) additions].

Calculating B:

$$B_i = H_i A_i = \begin{bmatrix} 6 \times 6 \\ \text{full} \\ \text{matrix} \end{bmatrix} \begin{bmatrix} 6 \times 6 \\ \text{full} \\ \text{matrix} \end{bmatrix}$$

at a cost of 36 multiplies 30 additions. Using triangular factorization to count the operation for B^{-1}, the cost is 195 multiplies 71 divides 206 additions. Total cost for Λ_j is

$N(48n + 36) + 195$	multiplies
71	divides
$N(48n) + 206$	additions

To calculate the cost for a jacobian inverse a generalized inverse formulation is used. $J_i^{-1} = J_i^T[J_i J_i^T]^{-1}$ where J_i is the 6 x n jacobian for the ith manipulator and n is the degree of freedom of the manipulator. The cost is $12n + 6$ multiplies 71 divides and $10n + 206$ additions giving a total cost for Λ_j with calculation of the jacobian inverse as

$N(60n + 231) + 195$	multiplies
$71N + 71$	divides
$N(58n + 206) + 206$	additions

Appendix III

6. LTM System Parameters

Motor constants (for all 7 joints)

Torque Constant K_T	8.5	oz-in/A
Back emf constant K_E	6.3	V/KRPM
Armature resistance R	2.5	OHM
Armature inertia J_m	0.0015	oz-in-sec^2
Viscous Damping D	0.3	oz-in-KRPM
Static Friction T_f	0.8	oz-in

Gear ratio from motor shaft to joint (assuming conventional gear driven joints)

Joint 1	522
Joint 2	522
Joint 3	522
Joint 4	522
Joint 5	121
Joint 6	121
Joint 7	25

Denavit and Hartenberg parameters

Denavit-Hartenberg Parameters				
joint	d	a	a	θ
1	0	-90°	0	θ_1 + 90°
2	0	90°	23"	θ_2
3	0	-90°	0	θ_3 - 90°
4	0	90°	20"	θ_4
5	0	-90°	0	θ_5
6	0	90°	0	θ_6 + 90°
7	5.9"	0	0	θ_7 - 90°

Joint Inertia Matrix

Link Number	Inertia Matrix KG-M			Orientation Matrix		
1	0.029	0.0	0.0	1.0	0.0	0.0
	0.0	0.0145	0.0	0.0	0.0	-1.0
	0.0	0.0	0.0145	0.0	1.0	0.0
2	0.029	0.0	0.0	0.0	1.0	0.0
	0.0	0.2989	0.0	0.0	0.0	1.0
	0.0	0.0	0.2989	1.0	0.0	0.0
3	0.029	0.0	0.0	1.0	0.0	0.0
	0.0	0.0145	0.0	0.0	0.0	-1.0
	0.0	0.0	0.0145	0.0	1.0	0.0
4	0.029	0.0	0.0	0.0	-1.0	0.0
	0.0	0.2296	0.0	0.0	0.0	1.0
	0.0	0.0	0.2296	-1.0	0.0	0.0
5	0.0163	0.0	0.0	1.0	0.0	0.0
	0.0	0.0082	0.0	0.0	0.0	-1.0
	0.0	0.0	0.0082	0.0	1.0	0.0
6	0.0163	0.0	0.0	1.0	0.0	0.0
	0.0	0.0269	0.0	0.0	0.0	1.0
	0.0	0.0	0.0269	0.0	-1.0	0.0
7	0.0182	0.0	0.0	0.0	-1.0	0.0
	0.0	0.0099	0.0	0.0	0.0	1.0
	0.0	0.0	0.0099	-1.0	0.0	0.0

where the orientation matrix is referenced to the base of LTM.

7. Bibliography

1. Hayati, S.: "Hybrid Position/Force Control of Multiarm Cooperating Robots." IEEE Conference on Robotics and Automation, April 1986.

2. Alberts, T., and Soloway, D.: "Force Control of Multiarm Robot System." IEEE Conference on Robotics and Automation, August 1988.

3. Carnignan, Craig R., and Akin, David L.: "Cooperative Control of two arms in The Transport of an Inertial Load in Zero G." IEEE Conference on Robotics and Automation, August 1988.

4. Mason, M. T.: "Compliance and Force Control for Computer Controlled Manipulators." IEEE Trans. on Systems, Man Cybernetics, v. 11, June 1981, pp. 418-432.

5. Denavit, J., and Hartenberg, R. S.: "A Kinematic Notation for Lower Pair Mechanisms Based on Matrices." ASME Journal of Applied Mechanics, June 1955, pp. 215-221.