Inertial-Space Disturbance Rejection for Space-Based Manipulators

Kevin Holt and Alan A. Desrochers
Rensselaer Polytechnic Institute
Troy, New York

Disturbance Rejection for Space–Based Manipulators

K. Holt
Lockheed Engineering & Sciences Company
2400 NASA Road 1
M/S C33
Houston, TX 77058

A.A. Desrochers
Electrical, Computer, & Systems
Engineering Department
Rensselaer Polytechnic Institute
Troy, NY 12180-3590

ABSTRACT

This paper describes the implementation of a disturbance rejection controller for a 6–DOF PUMA manipulator mounted on a 3–DOF platform. A control algorithm is designed to track the desired position and attitude of the end–effector in inertial space, subject to unknown disturbances in the platform axes. Experimental results are presented for step, sinusoidal, and random disturbances in the platform rotational axis and in the neighborhood of kinematic singularities.

Robotic manipulators have been proposed as a means of reducing the amount of extra vehicular activity time required for space station assembly and maintenance. The proposed scenario involves a robotic manipulator attached to some mobile platform, such as a spacecraft, satellite, or the space station itself.

Disturbances in the platform position and attitude may prevent the manipulator from successfully completing the task. This work explores the possibility of using the *manipulator* to compensate for platform disturbances.

The problem of controlling a robotic manipulator on a mobile platform has received considerable attention in the past few years. Joshi and Desrochers [1] designed a nonlinear feedback control law to carry out tasks (with respect to the robot base frame) in the presence of roll, pitch and yaw disturbances in the platform axes. Dubowsky, Vance, and Torres [2] proposed a time-optimal planning algorithm for a robotic manipulator mounted on a spacecraft, subject to saturation limits in the attitude control reaction jets. Papadopoulos and Dubowsky [3] developed a general framework for analyzing the control of free-floating space manipulator systems. Most recently, Torres and Dubowsky [4] have presented a technique called the enhanced disturbance map to find manipulator trajectories that reduce the effect of disturbances in the spacecraft position and attitude.

One common assumption in the literature is that the disturbance signal is exactly known. If this is the case, then the end-effector location can be calculated without relying on direct end-point sensing. However, this assumption is invalid if there is a significant delay in the platform position and attitude measurements, or if the kinematics of the platform are not well known, or if the platform is a non-rigid structure. In the more likely case that only the *nominal* platform location and *upper bound* on the disturbance signal are known, direct end-point sensing is needed to measure the end-effector location.

1 The Jacobian and Singularities

The inverse Jacobian is ill-defined for certain manipulator configurations. This section presents an alternative mapping, called the approximate pseudoinverse Jacobian, which is defined for all manipulator configurations.

The Jacobian maps differential changes in joint position to differential changes in Cartesian position and orientation according to the following relationship:

$$du = J(q)dq \tag{1}$$

where $du \in \Re^6$ is the differential Cartesian displacement vector (linear and angular), $q \in \Re^n$ is the vector of joint positions, $dq \in \Re^n$ is the vector of differential joint displacements, and $J \in \Re^{6\times n}$ is the Jacobian matrix.

For the PUMA, the Jacobian matrix is simplest when expressed in frame 6:

$$ {}^6J_{3,9} = \begin{bmatrix} --(d_5+d_6)C_{56} & d_7+a_5S_6 & d_7 \\ (d_5+d_6)S_{56} & a_6+a_5C_6 & a_6 \\ a_5C_5+a_6C_{56}+d_7S_{56} & 0 & 0 \\ --S_{56} & 0 & 0 \\ --C_{56} & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \tag{2} $$

$$ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -S_7 & C_7S_8 \\ --1 & 0 & -C_8 \\ 0 & C_7 & S_7S_8 \end{bmatrix} $$

The following compact notation will be used to denote the matrix ${}^6J_{3,9}$:

$$ {}^6J_{3,9} = \begin{bmatrix} B & 0 \\ D & E \end{bmatrix} \tag{3} $$

where B, D, and E are 3×3 submatrices of the Jacobian.

The inverse Jacobian, when it exists, can also be written in block matrix form

$$ {}^{6}J_{3,9}^{-1} = \begin{bmatrix} B & 0 \\ D & E \end{bmatrix}^{-1} = \begin{bmatrix} B^{-1} & 0 \\ --E^{-1}DB^{-1} & E^{-1} \end{bmatrix} \quad (4) $$

The PUMA has three singularities. The first is referred to as the *Arm Fully Stretched singularity*. This singularity occurs whenever the arm switches between the **flex** and the **noflex** configurations.

The second singularity corresponds to the *Hand Over Head singularity*. The Hand Over Head configuration corresponds to changing between the **right** and **left** configurations.

The third singularity is the *Wrist singularity*, and occurs when the arm switches between the **flip** and **noflip** configurations.

2 Approximate Pseudoinverse Jacobian

The usual method of dealing with singularities of the Jacobian is to avoid them. This approach is not applicable to the disturbance rejection problem since a sufficiently large disturbance could force the manipulator into a singular configuration. Also, the manipulator must avoid not just singular *points*, but singular *regions*, since the norm of J^{-1} becomes very large in the neighborhood of a singularity.

The pseudoinverse Jacobian is often used to overcome the difficulty of J being a nonsquare matrix, and is defined as

$$ J^{\dagger} \triangleq \begin{cases} J^T(JJ^T)^{-1} & m \leq n \\ J^{-1} & m = n \\ (J^TJ)^{-1}J^T & m \geq n \end{cases} \quad (5) $$

Clearly, this method of computing $J^{\dagger}$ does not address the issue of singularities since it still relies on matrix inversion. A more general approach to computing the pseudoinverse uses *singular value decomposition*. This has one serious drawback, which is the high cost of computing the singular value decomposition (SVD). The SVD algorithm uses a series of Householder transformations to reduce the input matrix to diagonal form. Since this is an $\mathcal{O}(N^3)$ operation, finding the SVD for the 6×6 Jacobian matrix can be too costly to implement in real-time. The alternative presented in this section is called the *approximate pseudoinverse Jacobian*, and is denoted by $J^{\ddagger}$.

The basic idea behind the approximate pseudoinverse is to use the partitioned form of J and perform the SVD on the submatrices B and E. This reduces the number of computations by a factor of four, since two 3×3 singular value decompositions is an $\mathcal{O}(2(N/2)^3)$ operation.

The definition of the approximate pseudoinverse Jacobian is

$$ J^{\ddagger} \triangleq \begin{bmatrix} B^{\dagger} & 0 \\ --E^{\dagger}DB^{\dagger} & E^{\dagger} \end{bmatrix} \quad (6) $$

where B, D, and E are defined as in (3).

Several properties of the approximate pseudoinverse are stated below.

1. $J^{\ddagger} = J^{-1}$ when J is nonsingular.
2. $J^{\ddagger}$ does *not* satisfy the Moore-Penrose conditions when J is singular.
3. Let $dp, d\phi \in \Re^3$ be the linear and angular components of du, respectively, and let $dq_1, dq_2 \in \Re^3$ be the components of dq. Then, the approximate pseudoinverse solution is

$$ \begin{bmatrix} dq_1 \\ dq_2 \end{bmatrix} = \begin{bmatrix} B^{\dagger} & 0 \\ --E^{\dagger}DB^{\dagger} & E^{\dagger} \end{bmatrix} \begin{bmatrix} dp \\ d\phi \end{bmatrix} \quad (7) $$

If J is singular, the approximate pseudoinverse finds the minimum norm solution *as if dp and $d\phi$ were decoupled*; that is, $dq = J^{\ddagger}du$ minimizes $\|Bdq_1 - dp\|_2$ and $\|Edq_2 - d\phi\|_2$.

3 Behavior Near Singularities

Figure 1 compares the 2-norm, or the maximum singular value, of $J^{\dagger}$ (solid curve), $J^{\ddagger}$ (dashed curve), and J^{-1} (dotted curve) in the vicinity of the Hand Over Head singularity.

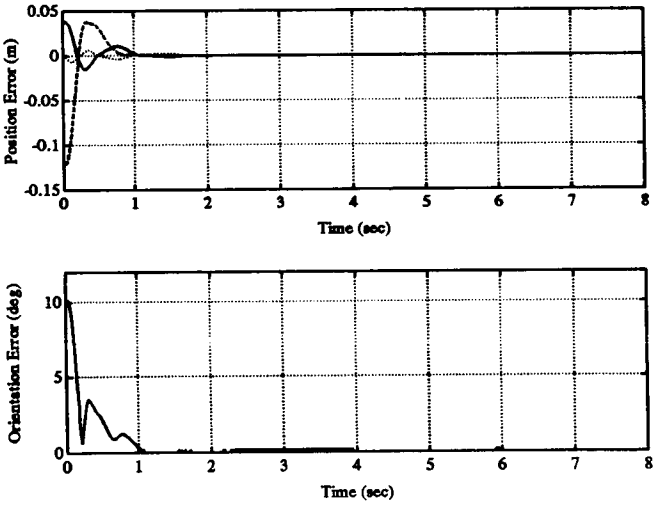

Figure 1: 2-Norms of $J^{\dagger}$ (solid curve), $J^{\ddagger}$ (dashed curve), and J^{-1} (dotted curve) Near Hand Over Head Singularity

The discontinuities in $\|J^{\dagger}\|_2$ and $\|J^{\ddagger}\|_2$ occur when the smallest nonzero singular value, σ_r, falls below the threshold value, σ_{min}. Setting σ_{min} to a relatively small value will shrink the width of the "well" about the singular point, thus extending the range over which $J^{\dagger} = J^{-1}$ and $J^{\ddagger} = J^{-1}$. The side-effect is that the norm will be very large and highly discontinuous near the singularity. By the same token, setting σ_{min} to a relatively large value will reduce the discontinuity in the norm by increasing the width of the singular region. A threshold value of $\sigma_{min} = 0.1$ was used to generate Figure 1.

4 Bound on Approximation Error

The pseudoinverse and approximate pseudoinverse Jacobians are identical only when J is nonsingular. In order to characterize the difference in behavior at a singularity, some measure of the *approximation error* is needed. A reasonable way to measure the approximation error is to see "how close" $J^{\ddagger}$ is to being a true generalized inverse using the following norm:

$$\|JJ^{\ddagger}J - J\|_2$$

Consider the matrix

$$JJ^{\ddagger}J = \begin{bmatrix} BB^{\dagger}B & 0 \\ D-(I-EE^{\dagger})D(I-B^{\dagger}B) & EE^{\dagger}E \end{bmatrix} \quad (8)$$

Subtracting J yields

$$JJ^{\ddagger}J - J = \begin{bmatrix} I & 0 \\ 0 & I-EE^{\dagger} \end{bmatrix} \begin{bmatrix} B & 0 \\ D & E \end{bmatrix} \begin{bmatrix} I-B^{\dagger}B & 0 \\ 0 & I \end{bmatrix} \quad (9)$$

When both B and E are singular, the approximation error is bounded as follows:

$$\begin{aligned} &\|JJ^{\ddagger}J - J\|_2 = \\ &\left\| \begin{bmatrix} I & 0 \\ 0 & I-EE^{\dagger} \end{bmatrix} \begin{bmatrix} B & 0 \\ D & E \end{bmatrix} \begin{bmatrix} I-B^{\dagger}B & 0 \\ 0 & I \end{bmatrix} \right\|_2 \quad (10) \\ &\leq \|J\|_2 \end{aligned}$$

If B is nonsingular, a less conservative upper bound can be found:

$$\begin{aligned} \|JJ^{\ddagger}J - J\|_2 &= \left\| \begin{bmatrix} I & 0 \\ 0 & I-EE^{\dagger} \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & E \end{bmatrix} \right\|_2 \\ &\leq \|E\|_2 \quad (11) \end{aligned}$$

Likewise, when E is nonsingular the upper bound reduces to

$$\begin{aligned} \|JJ^{\ddagger}J - J\|_2 &= \left\| \begin{bmatrix} B & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} I-B^{\dagger}B & 0 \\ 0 & I \end{bmatrix} \right\|_2 \\ &\leq \|B\|_2 \quad (12) \end{aligned}$$

Finally, if both B and E are nonsingular, the approximate pseudoinverse is identical to the pseudoinverse:

$$\|JJ^{\ddagger}J - J\|_2 = 0 \quad (13)$$

Coordinate Frame k	Computation Time		
	${}^{k}J^{-1}_{3,E}$	${}^{k}J^{\dagger}_{3,E}$	${}^{k}J^{\ddagger}_{3,E}$
0	1.31 ms	25.31 ms	6.38 ms
1	1.31 ms	25.31 ms	6.38 ms
2	1.19 ms	25.31 ms	6.25 ms
3	1.09 ms	24.98 ms	6.11 ms
4	0.97 ms	24.65 ms	5.98 ms
5	0.97 ms	24.65 ms	5.98 ms
6	0.88 ms	24.98 ms	5.98 ms
7	0.82 ms	24.98 ms	5.84 ms
8	0.81 ms	24.98 ms	5.85 ms
9	0.81 ms	24.65 ms	5.85 ms
E	0.95 ms	25.31 ms	6.11 ms

Table 1: Computation Times for ${}^{k}J^{-1}_{3,E}$, ${}^{k}J^{\dagger}_{3,E}$, and ${}^{k}J^{\ddagger}_{3,E}$

5 Computation Time

Table 1 compares the computation times of the the inverse, pseudoinverse, and approximate pseudoinverse Jacobians for each coordinate frame. As predicted, the approximate pseudoinverse is about four times faster to compute than the pseudoinverse.

The inverse, pseudoinverse, and approximate pseudoinverse Jacobian solutions were implemented in the C programming language using the GNU[1] gcc Version 2.2.2 compiler. The data in Table 1 was collected by timing the software on a Motorola MVME 147SA-2 Single Board Computer.

6 A Kinematic Control Law for Disturbance Rejection

Consider a 6-DOF PUMA manipulator mounted on a 3-DOF platform. The goal is to maintain the desired position and attitude of the end-effector with respect to the inertial reference frame (frame 0), subject to arbitrary disturbances in the platform axes. The following information is assumed to be available:

1. $\theta \in \Re^6$, the PUMA joint positions
2. $\eta_o \in \Re^3$, the nominal platform joint positions
3. $\bar{\delta} \in \Re^3$, the maximum deviations from the nominal platform joint positions
4. ${}^{0}u_{0,E} \in \Re^6$, the inertial end-effector location

Two factors contribute to the motion of the end-effector: the differential displacement of the PUMA joints, which can be measured, and the differential displacement of the platform joints, which is unknown. Let δ denote the disturbance signal and let dv be the component of the end-effector motion caused by the

[1]Copyright (C) 1989, 1991 Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA.

differential displacement of the platform joints. Then, the differential end-effector displacement can be written as

$$\begin{aligned} {}^0du_{0,E} &= {}^0J_{3,E}(\eta_o+\delta,\theta)d\theta + dv \\ &= {}^0_3\mathbf{R}(\eta_o+\delta)\,{}^3J_{3,E}(\theta)d\theta + dv \end{aligned} \quad (14)$$

Note that coordinate frame transformations have been applied to isolate the dependence of the PUMA Jacobian on the platform joint positions.

A discrete-time model of the system will now be derived by approximating the differential quantities in (14) with displacements. The underlying assumption here is that the sampling period, ΔT, is sufficiently small (i.e., the sampling rate is much higher than the bandwidth of the system).

Define Δu_k as $\Delta u_k \triangleq u_k - u_{k-1}$, where the subscript k denotes the kth sample step. In the limit as ΔT goes to zero, the displacement Δu_k equals the differential du:

$$\lim_{\Delta T \to 0} \Delta u_k = du \quad (15)$$

Similarly, $\Delta\theta_k \to d\theta$ and $\Delta v_k \to dv$ as $\Delta T \to 0$. Therefore, the discrete-time approximation is

$$\begin{aligned} du &\approx \Delta u_k = u_k - u_{k-1} \\ d\theta &\approx \Delta\theta_k = \theta_k - \theta_{k-1} \\ dv &\approx \Delta v_k = v_k - v_{k-1} \end{aligned} \quad (16)$$

and the discrete version of (14) is

$${}^0u_k - {}^0u_{k-1} = {}^0_3\mathbf{R}(\eta_o+\delta_k)\,{}^3J_{3,E}(\theta_k)\Delta\theta_k + \Delta v_k \quad (17)$$

where the subscripts denoting the reference and velocity frames of du have been dropped to avoid confusion with the time index.

Let ${}^0u_d$ be the desired position and orientation of the end-effector along some specified trajectory. The control objective is to drive the end-effector to this position and orientation:

$${}^0u_k \to {}^0u_d \quad \text{as} \quad k \to \infty \quad (18)$$

Ideally, the control objective could be achieved in minimum time by computing the PUMA joint displacements $\Delta\theta_d$ needed to cancel out the inertial-space error. However, exact cancellation would require complete knowledge of the disturbance signal. The next best solution then is to compute a $\Delta\theta_d$ which *approximately* cancels out the inertial-space error. With this goal in mind, the proposed control law is

$$\Delta\theta_d = {}^3J^{\dagger}_{3,E}(\theta_k)\,{}^3_0\mathbf{R}(\eta_o)K_c({}^0u_d - {}^0u_k) \quad (19)$$

where $K_c \in \Re^{6\times 6}$ is a matrix of control gains. Equation (19) will be referred to as the $J^{\dagger}$ *control law* in the sequel.

A simple expression for the closed-loop system can be derived by assuming that there is a one period delay in the control actuation:

$$\Delta\theta_{k+1} = \Delta\theta_d \quad (20)$$

$$\begin{aligned} {}^0u_k - {}^0u_{k-1} &= {}^0_3\mathbf{R}(\eta_o+\delta_k)\,{}^3J_{3,E}(\theta_k) \\ &\quad {}^3J^{\dagger}_{3,E}(\theta_{k-1})\,{}^3_0\mathbf{R}(\eta_o)K_c({}^0u_d - {}^0u_{k-1}) + \Delta v_k \end{aligned} \quad (21)$$

In order to simplify this expression, define the quantity

$$M_{k,k-1} \triangleq {}^0_3\mathbf{R}(\eta_o+\delta_k)\,{}^3J_{3,E}(\theta_k)\,{}^3J^{\dagger}_{3,E}(\theta_{k-1})\,{}^3_0\mathbf{R}(\eta_o)K_c \quad (22)$$

Rewriting (21) in terms of $M_{k,k-1}$, it is easy to see that the closed-loop system is linear with time-varying coefficients:

$${}^0u_k = (I - M_{k,k-1})\,{}^0u_{k-1} + M_{k,k-1}\,{}^0u_d + \Delta v_k \quad (23)$$

A block diagram of the closed-loop system is shown in Figure 2.

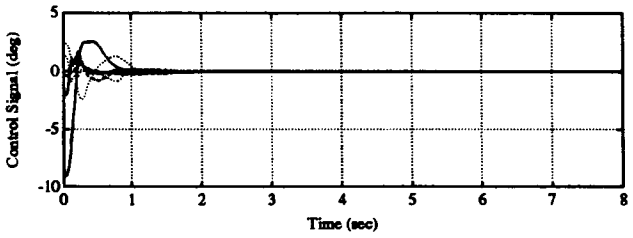

Figure 2: Block Diagram of Closed-Loop System

6.1 Design Parameters

The selection of the control gain is greatly simplified by restricting K_c to be a scalar times the identity matrix:

$$K_c = k_c I, \quad 0 \le k_c \le 2 \quad (24)$$

The parameter k_c controls the spectral radius of M_k. For example, if $k_c = 0.5$, then the eigenvalues of M_k will lie on a circle of radius 0.5 in the Λ-plane (or at zero, if J is singular).

It is straightforward to choose a stable k_c if $\bar{\delta}$ is known *a priori*. (Recall that $\bar{\delta}$ is the vector of maximum deviations in the platform joint positions.) Let $\bar{\lambda}$

denote the spectrum of the matrix ${}^0_3\mathbf{R}(\eta_o + \bar{\delta}){}^3_0\mathbf{R}(\eta_o)$. By invoking the slowly time-varying condition, α can be approximated as follows:

$$\alpha \approx \sup_i \arg(\bar{\lambda}) \tag{25}$$

and k_c is calculated as

$$k_c = \frac{2}{\sqrt{\tan^2 \alpha + 1}} \tag{26}$$

The selection of σ_{min} is essentially a trade-off between tracking accuracy and the norm of the control signal. The selection of σ_{min} should be based on the desired upper bound on the norm of $\Delta\theta_d$, which in turn is dictated by the saturation limits of the joint-level controller.

7 Experimental Results

Three sets of experiments focused on the time response of the closed-loop system for step disturbances in the platform joints, sinusoidal disturbances in the platform joints, and random disturbances in the platform joints. Here we present only the results for the step disturbance.

This section analyzes the time response of the closed-loop system for 10° and 30° step disturbances in the platform rotation. For each case, the control gain K_c was set to identity.

7.1 10° Step Disturbance

Figure 3 shows the inertial-space errors errors when a 10° step disturbance is applied to the platform rotational joint. The linear (X, Y, and Z) components of the error are shown in the upper plot and the orientation error in the lower plot. The components of $\Delta\theta_d$, the control vector, are plotted in Figure 4.

	Maximum Overshoot	4% Settling Time
X	$1.527\times10^{+0}$ cm	1.54 s
Y	$3.825\times10^{+0}$ cm	0.84 s
Z	6.366×10^{-1} cm	1.70 s
ϕ_e	$3.503\times10^{+0}$ deg	1.00 s

Table 2: Maximum Overshoot and 4% Settling Time for 10° Step Disturbance in Platform Rotation

Table 2 lists the maximum overshoot and 4% settling time for the X, Y, Z, and orientation errors. The 4% settling time refers to the time required for the error to enter and remain within $\pm\epsilon$ of zero, where ϵ is 4% of the peak absolute error.

7.2 30° Step Disturbance

The inertial-space errors and control signals for the 30° case are shown in Figures 5 and 6. The maximum overshoot and settling time for each coordinate are displayed in Table 3.

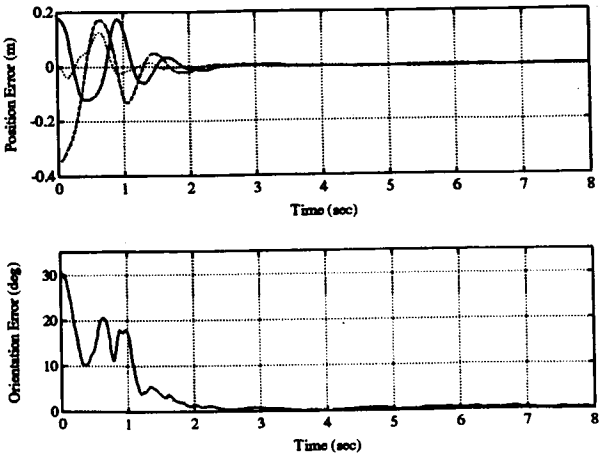

Figure 3: Position Error (X – solid curve; Y – dashed curve; Z – dotted curve) and Orientation Error for 10° Step Disturbance in Platform Rotation

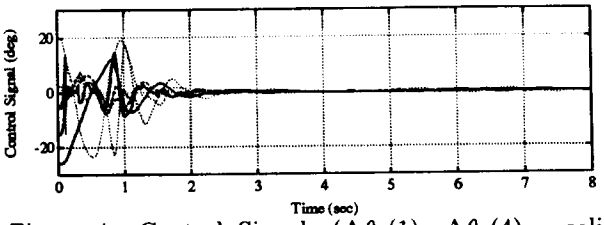

Figure 4: Control Signals ($\Delta\theta_d(1)$, $\Delta\theta_d(4)$ – solid curves; $\Delta\theta_d(2)$, $\Delta\theta_d(5)$ – dashed curves; $\Delta\theta_d(3)$, $\Delta\theta_d(6)$ – dotted curves) for 10° Step Disturbance in Platform Rotation

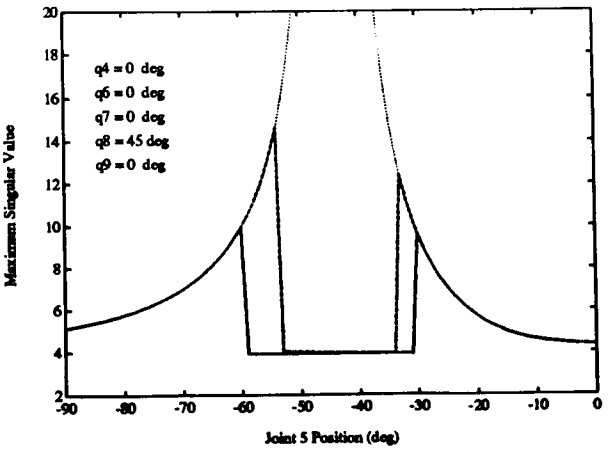

Figure 5: Position Error (X – solid curve; Y – dashed curve; Z – dotted curve) and Orientation Error for 30° Step Disturbance in Platform Rotation

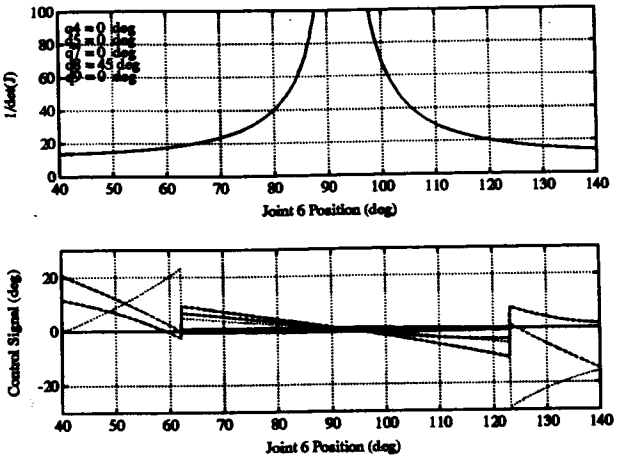

Figure 6: Control Signals ($\Delta\theta_d(1)$, $\Delta\theta_d(4)$ – solid curves; $\Delta\theta_d(2)$, $\Delta\theta_d(5)$ – dashed curves; $\Delta\theta_d(3)$, $\Delta\theta_d(6)$ – dotted curves) for 30° Step Disturbance in Platform Rotation

	Maximum Overshoot	4% Settling Time
X	$1.737\times10^{+1}$ cm	1.97 s
Y	$1.706\times10^{+1}$ cm	2.43 s
Z	$1.253\times10^{+1}$ cm	1.66 s
ϕ_e	$2.055\times10^{+1}$ deg	2.08 s

Table 3: Maximum Overshoot and 4% Settling Time for 30° Step Disturbance in Platform Rotation

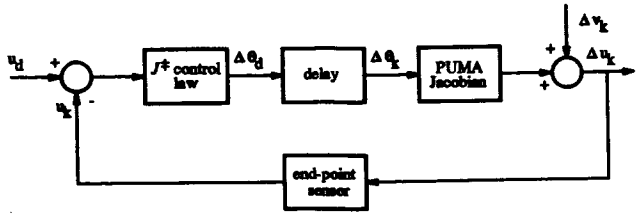

Figure 7: Behavior of $1/\det(J)$ and Open-Loop Control Signals ($\Delta\theta_d(1)$, $\Delta\theta_d(4)$ – solid curves; $\Delta\theta_d(2)$, $\Delta\theta_d(5)$ – dashed curves; $\Delta\theta_d(3)$, $\Delta\theta_d(6)$ – dotted curves) Near Arm Fully Stretched Singularity

8 Behavior Near Singularities

Figure 7 shows the vector of open-loop control signals near the Arm Fully Stretched singularity. The minimum singular value parameter, σ_{min}, was set to 0.1. At this value of σ_{min}, the control in the direction of the workspace boundary becomes very weak approximately 30° from the singular point. This prevents the end-effector from getting too close to the workspace boundary. Consequently, the manipulator will not switch between the `flex` and `noflex` configurations while the $J^{\ddagger}$ controller is running.

If the parameter σ_{min} is sufficiently small, however, the width of the singular region will be reduced to the point where the control signal for joint 6 ($\Delta\theta_d(3)$) could drive the arm through the singularity. This may lead to an undesirable "chattering" behavior, in which the arm rapidly oscillates between the `flex` and `noflex` configurations.

9 Summary

Several important conclusions can be drawn from the experimental results.

1. The relative stability of the closed-loop system is a function of the amplitude of the disturbance signal.

2. The relative performance of the controller is a function of the frequency of the disturbance signal.

In other words, the $J^{\ddagger}$ controller is like a high-pass filter; the lowest frequency components of the disturbance signal are attenuated the most.

3. The control in certain directions becomes very weak near singularities.

This implies that there may be an unavoidable tracking error in the "forbidden" directions when the arm is at or near a singularity.

References

[1] J. Joshi and A. A. Desrochers, "Modeling and Control of a Mobile Robot Subject to Disturbances," in *Proc. 1986 IEEE Robotics and Automation Conference*, (San Francisco, CA), 1986.

[2] S. Dubowsky, E. E. Vance, and M. A. Torres, "The Control of Space Manipulators Subject to Spacecraft Attitude Control Saturation Limits," in *Proc. NASA Conf. Space Telerobotics*, (Pasadena, CA), January – February 1989.

[3] E. Papadopoulos and S. Dubowsky, "On the Nature of Control Algorithms for Space Manipulators," in *Proc. 1990 IEEE Robotics and Automation Conference*, (Cincinnati, OH), 1990.

[4] M. A. Torres and S. Dubowsky, "Minimizing Spacecraft Attitude Disturbances in Space Manipulator Systems," *Journal of Guidance, Control, and Dynamics*, vol. 15, pp. 1010 – 1016, July – August 1992.