KINEMATICALLY REDUNDANT ARM FORMULATIONS FOR COORDINATED MULTIPLE ARM IMPLEMENTATIONS

Robert W. Bailey and Leslie J. Quiocho
LinCom Corporation
1020 Bay Area Blvd, Suite 200 Houston, Tx 77058

Dr. Timothy F. Cleghorn
Mission Planning and Analysis Division - FM7
NASA/Lyndon B. Johnson Space Center Houston, Tx 77058

ABSTRACT

Although control laws for kinematically redundant robotic arms were presented as early as 1969 [1], redundant arms have only recently become recognized as viable solutions to limitations inherent to kinematically sufficient arms. The advantages of run-time control optimization and arm reconfiguration are becoming increasingly attractive as the complexity and criticality of robotic systems continues to progress. This paper presents a generalized control law for a spatial arm with 7 or more degrees of freedom (DOF) based on Whitney's resolved rate formulation [1]. Results from a simulation implementation utilizing this control law are presented. Furthermore, results from a two arm simulation are presented to demonstrate the coordinated control of multiple arms using this formulation.

INTRODUCTION

Within a resolved rate motion control scheme, the joint motors are run simultaneously to provide varying joint velocities consistent with constant commanded point of resolution (POR) velocities in Cartesian space. The fundamental relationship between joint or configuration space and task space velocities is the Jacobian matrix, which maps a linear transformation between the two spaces. For kinematically redundant arms, transforming task space commands (6 DOF) to joint space consisting of 7 or more DOFs requires resolving the redundancy, i.e., solving an underdetermined set of equations.

Two primary techniques have been proposed to resolve the redundancy as applied to robotic systems: the method of Lagrange multipliers and the generalized or pseudoinverse technique. Whitney [1,2] uses Lagrange multipliers with an optimality criterion to be satisfied during the motion of the manipulator. Liegeois [3] utilizes generalized inverse matrices (also referred to as pseudoinverse matrices) and adds to the solution a minimization vector representing the deviation from the mean positions of each of the joints. Klein and Huang [4] similarly incorporate a function minimizing excursion from the joint center positions. Bourgeois [5] implements Whitney's algorithm with a weighting matrix to keep joints from approaching motion limits during a trajectory. In addition to these efforts, other optimality algorithms have been proposed to provide avoidance with obstacles [6] and to minimize torque loading on the joints in a least squares sense [7].

Despite these efforts, few results have been presented demonstrating the physical performance characteristics of the control laws [3,5]. Furthermore, literature on spatial formulations and simulation implementations of kinematically redundant arms is scarce (although numerous commercial vendors are marketing hardware with spatial 7-DOF control laws [8]). As a result, the focus of this effort is on 1) the spatial implementation of a generalized resolved rate law for a 7 or more DOF arm based on Whitney's original formulation, and 2) the physical performance characteristics of the control law.

FORMULATION

Equation (1) defines the robotic arm POR velocities as functions of the joint velocities. This relation may be derived using methods of classical mechanics [9] based on the parameter definitions depicted in Figure 1†.

$$\begin{bmatrix} \Omega \\ V \end{bmatrix} = [x] = [J]\left[\dot{\theta}\right] \qquad (1)$$

where J, the Jacobian matrix, is defined as

$$J = \begin{bmatrix} U_1 & U_2 & \cdots & U_n \\ U_1 x R_1 & U_2 x R_2 & \cdots & U_n x R_n \end{bmatrix}.$$

The Jacobian J is an mxn matrix, where m represents the dimension of the task space and n the dimension of the joint space. In the case of a kinematically redundant system, $n>m$ and the system of equations is underdetermined, i.e., J is nonsquare and the inverse of J is undefined. Hence, additional criteria need to be introduced to produce a unique solution.

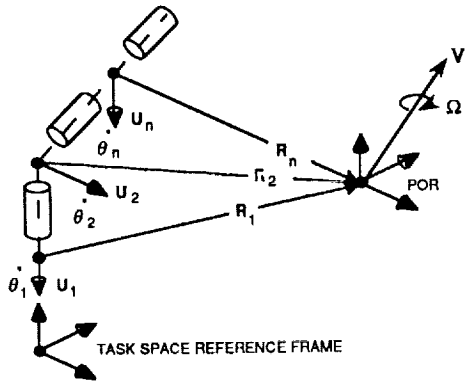


Figure 1 - N DOF Configuration Space Parameters

†Manipulator joint rates $\dot{\theta}$ previously defined as $\dot{\gamma}$ in [9].

The method of Lagrange multipliers [10] can be used to optimize an objective or cost function subject to a given constraint. For the problem at hand, the method can be used to "square" the system of equations, thereby producing an invertible solution to Equation (1). The problem can be formulated in the following manner: suppose f and g are vector functions of some joint angle vector $\dot{\theta}$. If $f(\dot{\theta})$ has a local extrema (minimum or maximum) at $\dot{\theta}_o$ subject to the constraint $g(\dot{\theta})$, there must exist constants (Lagrange multipliers), Λ, such that the following expression is satisfied,

$$\nabla f(\dot{\theta}_o) = \Lambda \nabla g(\dot{\theta}_o). \qquad (2)$$

Whitney [1,2] introduces the following objective function in matrix form,

$$f(\dot{\theta}) = 1/2\ [\dot{\theta}]^T A [\dot{\theta}], \qquad (3)$$

where A is an nxn diagonal matrix of weighting terms. The constraint function, from Equation (1), is given by

$$g(\dot{\theta}) = [\dot{x}] = [J][\dot{\theta}]. \qquad (4)$$

Applying Equation (2) to Equations (3) and (4) yields the following expression,

$$[A][\dot{\theta}] = [J]^T[\Lambda] \qquad (5)$$

which can be rewritten as

$$[\dot{\theta}] = [A]^{-1}[J]^T[\Lambda]. \qquad (6)$$

After applying the method of Lagrange multipliers, matrix algebra can now be used to form the inverse solution to Equation (1). Substituting Equation (6) into Equation (4), and solving for Λ results in

$$[\Lambda] = \{\ [J][A]^{-1}[J]^T\ \}^{-1}\ [\dot{x}] \qquad (7)$$

Finally, the expression for Λ in Equation (7) can be substituted into Equation (6) resulting in the following resolved rate control law,

$$[\dot{\theta}] = [A]^{-1}[J]^T\ \{\ [J][A]^{-1}[J]^T\ \}^{-1}\ [\dot{x}] \qquad (8)$$

Alternatively, this solution could have been produced using a pseudoinverse method [1].

There seems to be some confusion in the literature as to the physical nature of the optimization function introduced in Equation (3). Whitney refers to this criterion as the "approximate instantaneous weighted system kinetic energy." Since the time of Whitney's publication, this statement has received several different interpretations. For example, Klein and Huang [4] state that "... since energy consumption can be related to the norm of joint velocities, and since the pseudoinverse finds the minimum norm solution, instantaneous power is minimized." Baker and Wampler [11] propose finding "... the joint speeds $\dot{\theta}_o$ which minimize some norm of $\dot{\theta}$, such as $\dot{\theta}^T\dot{\theta}$ or kinetic energy." The actual physical interpretation of this minimization criterion is illustrated in Figure 2. If the A matrix contains the moments of inertia of each of the arm segments about their axis of rotation, then Equation (3) represents the sum of the rotational kinetic energies of each of the segments, as if each segment was a 1 DOF system. In essence, Equation (3) seems to provide a convenient means to invert the equation set of Equation (1), rather than providing an actual physical criterion to minimize or maximize.

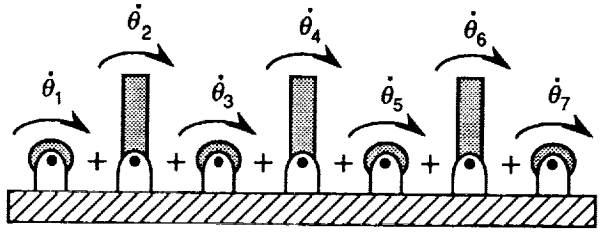


Figure 2 - Physical Interpretation of Objective Function

The vectors Ω_c and V_c, Equations (9) and (10) respectively, correspond to the commanded POR rotational and translational velocities. These parameters are graphically defined in Figure 3.

$$\Omega_c = u^T[T_o] \cdot \Omega_{lim} \qquad (9)$$

$$V_c = \text{unit}\ ([R_f - R_o]) \cdot V_{lim} \qquad (10)$$

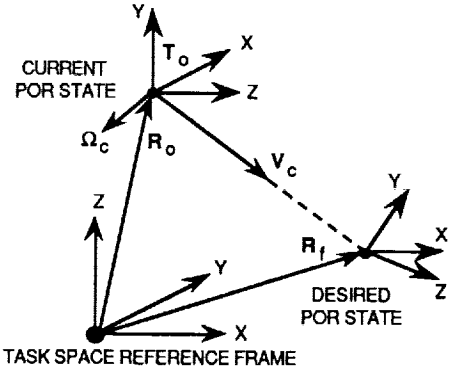


Figure 3 - POR Rotational & Translational Velocity Commands

Equations (9) and (10) produce linear, goal oriented command inputs to the control law [9]. In other words, for an ideal system, Equations (9) and (10) will produce linear paths to the desired POR orientation and position, both in a rotational and translational sense. For an imperfect system (sensor corruption, motor lag, etc.), actual POR paths will have a tendency to "spiral in" to the desired orientation and position, always moving towards the goal point regardless of the actual path taken.

SIMULATION IMPLEMENTATION

The control law of Equation (8) is implemented within the Robotics Software Testbed (RST) architecture developed by the Mission Planning and Analysis Division at the Lyndon B. Johnson Space Center. The RST provides a discrete time cyclic executive with the capability of integrating both hardware and software components into a simulation for both ground based and orbital applications. The application discussed here is generalized for seven or more DOF and data driven so that different arm configurations may be tested without code recompilation. A high level schematic flow diagram of the single arm simulation is presented in Figure 4.

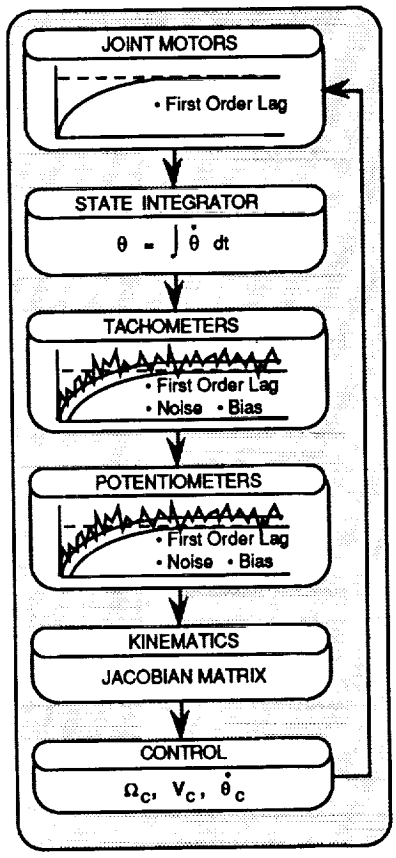


Figure 4 - Kinematic Arm Simulation Schematic Flow Diagram

The Robotics Research Corporation's (RRC) Model K-1607™ seven DOF arm was selected as the test configuration for initial analysis [8]. The arm has a reach of approximately five feet and a R-P-R-P-R-P-R joint configuration, as depicted in Figure 5. Numerous simulation runs were performed to help analyze the response of the control law. Specifically, five runs were chosen which best demonstrate the effects of the *A* matrix weighting on control performance.

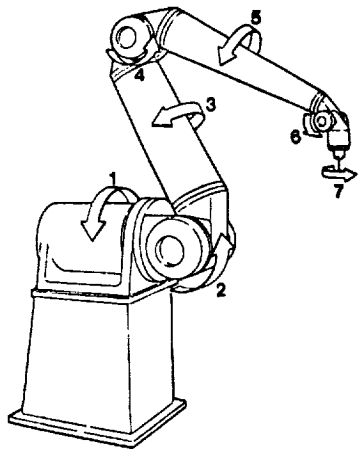


Figure 5 - Robotics Research Corporation Model K-1607™

The test case scenario used for each run consists of a single POR move from an initial to a final position and orientation in the task space. Each of the five simulations are identical upon initialization except for the weighting of the *A* matrix. All runs are executed with a 12.5 Hz control loop cycle. The specific differences between the *A* weighting for each of the runs are as follows:

Run 1 – constant unity weighting over duration of the run.
Run 2 – constant weighting over duration of run with joint 6 at two orders of magnitude above all other joints.
Run 3 – constant weighting over duration of run with joints 2 and 6 at two orders of magnitude above all other joints.
Run 4 – varied weighting over duration of run based upon actual joint distance from the joint median angle.
Run 5 – a dual arm simulation with the characteristics of Run 1 for both arms.

The weighting algorithm used for Run 4 performs the following steps:

1) find the minimum and maximum percentage deviations from the median angles among the joints (percentage deviation is the actual angular deviation over the actual angular displacement to the joint limit for the joint).

2) scale the *A* component corresponding to the joint with the minimum percentage of deviation to 1.0. Scale the *A* component corresponding to the joint with the maximum percentage of deviation to 100.0. Scale all other *A* components proportionally based on their respective joint percentage deviations.

Run 5 is presented to show the control law's ability to support multiple arm coordination. Cleghorn and Bailey [9] discuss the coordination of 6 DOF robotic arms utilizing the same linear, goal oriented command generation scheme of Equations (9) and (10). Run 5 consists of a concurrent implementation of two identical RRC arm models as used in Runs 1 through 4. The arms are configured side by side with the POR of arm 1 placed at the tip of the last joint segment (as in Runs 1 through 4), and the POR of arm 2 placed 0.2 feet off the end of the last joint segment and 1.0 feet to the side; this places the POR between the two arms. The arms are initialized with their PORs at the same location in the task space, and then given identical POR maneuver commands (the same maneuver executed for Runs 1 through 4).

RESULTS

Several interesting results are found common to all test runs. First, there is no appreciable change in the POR trajectory between the runs. This stability demonstrates the robustness of the control algorithm while performing a maneuver, given a wide variety of *A* weighting values and joint solutions. Although the POR trajectories were uniform over the five runs, the joint angle and joint rate histories differed considerably.

Second, the *A* matrix values are important in a relative rather than an absolute sense. For example, Run 1 was executed with 1) all *A* values equal to 1.0, and 2) all A values equal to 10000.0. No POR or joint differences were observed between these two cases. Also, Run 2 was executed with 1) the *A* values for joint 6 set at 100.0 and the remaining values set to 1.0; and 2) the *A* values for joint 6 set at 100000.0 and the remaining values set to 1000.0. For both runs, the ratio of the largest *A* value to the smallest *A* value was 100 (two orders of magnitude). Once again there was no noticeable POR or joint

differences were observed between the two runs. This shows the importance of the relative magnitudes of the A values.

Third, as shown by Bourgeois [5], increasing a single value of the A matrix relative to the other values has the effect of limiting the motion of that respective joint. In general, as the ratio of the largest to smallest A values falls below 100, the joint histories approach that of the all unity case (Run 1). As the ratio increases above 100, the weighting tends to increase the motion of all the low A value joints rather than decreasing the motion of all the high A value joints.

While all of the five runs exhibit common characteristics, there are some specific results of the individual runs worth mentioning. In Figure 6 notice that the acceleration, maneuver velocity, and deceleration regions of the POR velocity profiles are nearly linear. These velocity profiles result in linear POR position histories. Although the rotational velocity vector components are approximately linear, the orientation histories exhibit a slight curvature throughout the duration of the run. This can be attributed to the fact the velocity histories are presented as a rotational velocity vector in the task space frame, whereas the orientation histories are presented as pitch-yaw-roll Euler angle sequences. As discussed by Cleghorn and Bailey [9], this linear response can be used to an advantage in multiple arms systems. If each manipulator of a multiple arm system can be expected to follow a linear trajectory at a given speed, then the identical POR velocity commands given to each of the independently controlled arms should trace the same linear POR path. Further dual arm results are discussed for Run 5.

Figures 7 and 8 present joint angle and joint rate histories for each of Runs 1 through 4, where Run 1 represents the nominal case (unity weighting). Run 2 demonstrates the powerful ability of the A matrix weighting to limit the motion of a single joint; in this case joint 6. However, notice that the restricted motion of joint 6 induces more motion in the most of the remaining joints.

Run 3 is similar to Run 2, but with two joints weighted to limit their motion (joints 2 and 6). In comparison to the unity case (Run 1), motion for joints 2 and 6 is inhibited while motion for the remaining joints increases. In general, as more joints are weighted to limit their motion, their respective joint histories approach that of Run 1. As was previously mentioned, the relative magnitudes of the weighting elements are important and not their absolute magnitudes.

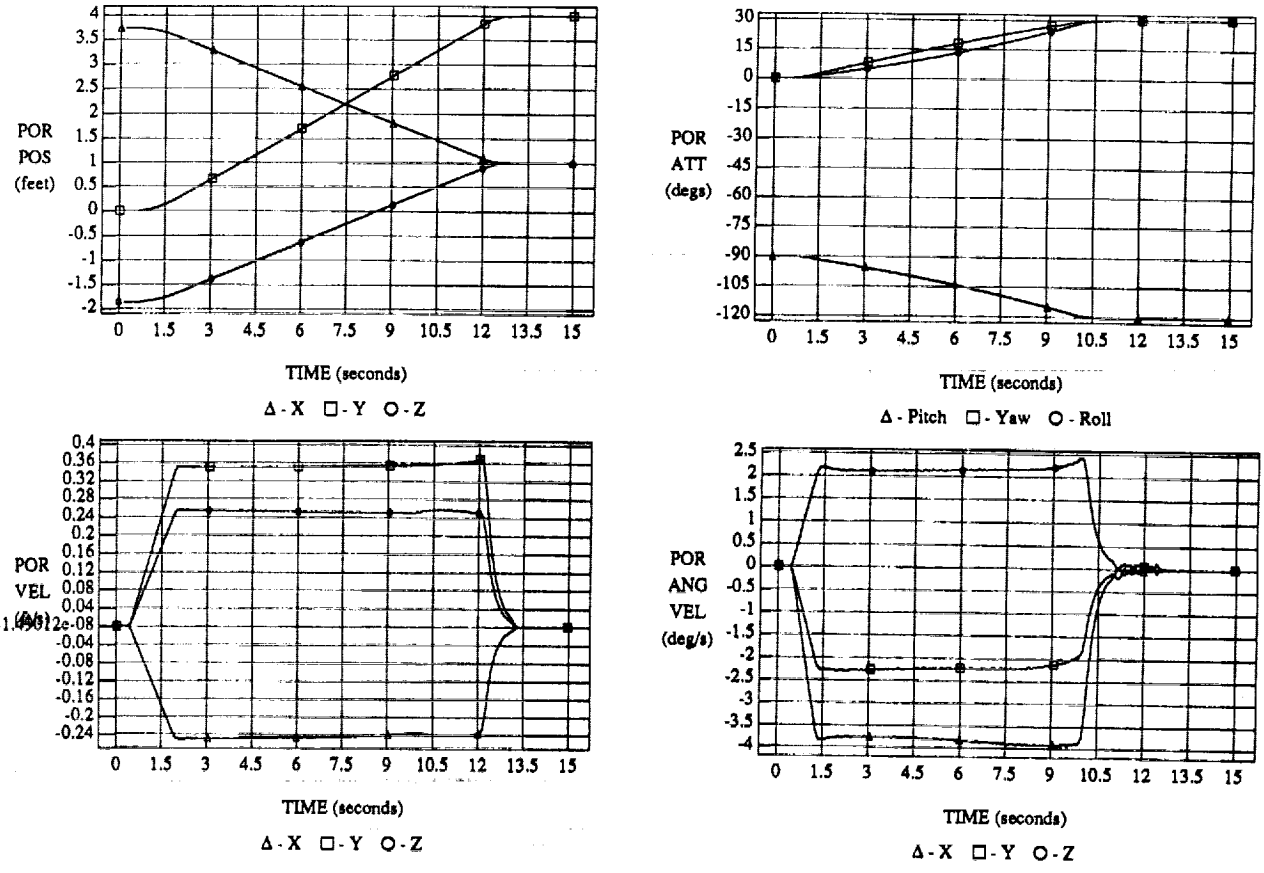
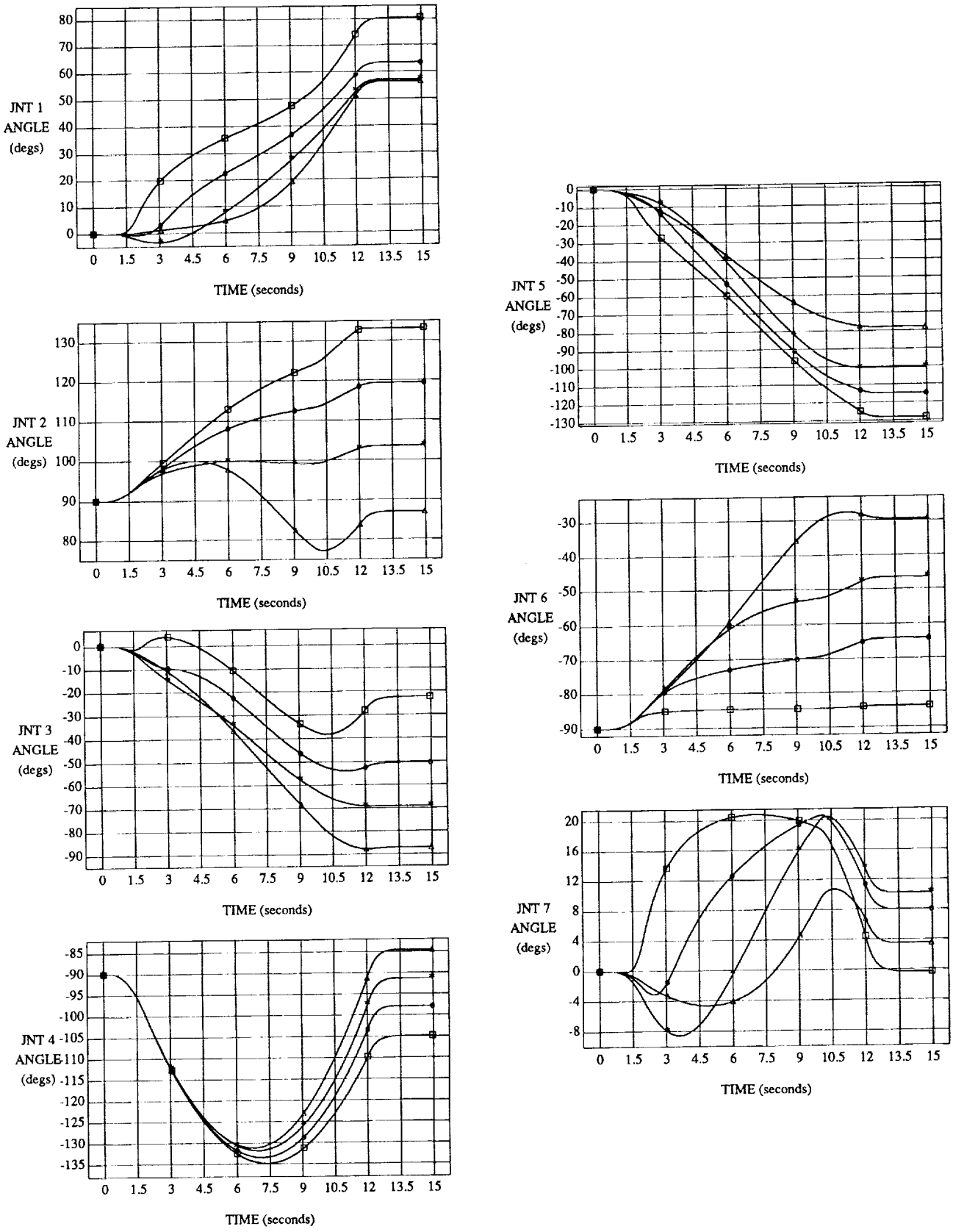


Figure 6 - POR Position and Velocity Histories for Run 1



Δ - Run 1 □ - Run 2 ○ - Run 3 * - Run 4

Figure 7 - Joint Angle Histories For Runs 1 Through 4

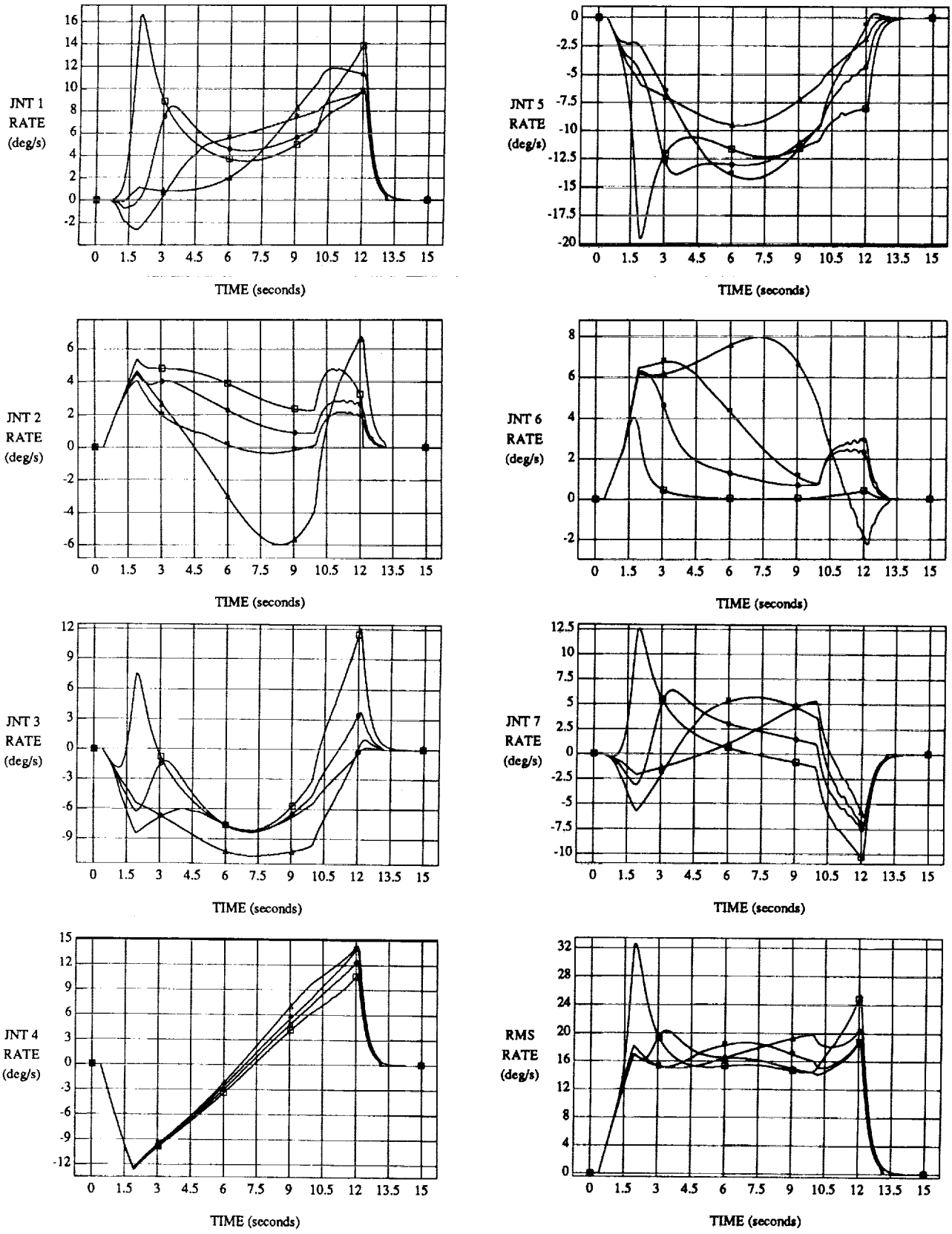


Figure 8 - Joint Rate Histories For Runs 1 Through 4

Run 4 differs from the previous three runs in that the weighting is performed dynamically rather than remaining constant. As the joint travels farther from the median position, the weighting is scaled to impede further motion away from the center position, which is not to be confused with attracting the joint to the center position. The effects are reflected in Figures 7 and 8. Since all joints are being weighted simultaneously, not all joints can be expected to fulfill the criteria of remaining close to the center position. One would expect the three joints farthest away from their center angles to exhibit the slowest motion. Likewise, the three joints closest to their center positions should exhibit increased motion and the remaining joint should stay relatively unchanged from the nominal (Run 1). In general, Run 4 demonstrates these results; as the data presented in Table 1 confirms. A plus in the PERFORM row of Table 1 indicates motion was slowed compared to Run 1; a zero indicates relatively no change in motion; and a minus indicates an increase in motion.

Table 1 - Configuration Data Pertinent to Run 4

JOINT	1	2	3	4	5	6	7
CENTER ANGLE	0.0	45.0	0.0	-90.0	0.0	-90.0	0.0
LOW HARDSTOP	-360.0	-45.0	-360.0	-180.0	-360.0	-180.0	-1080.0
HIGH HARDSTOP	360.0	135.0	360.0	0.0	360.0	0.0	1080.0
% DEVIATION @ 15s	16	66	19	2	28	49	1
+/0/- PERFORM	0	+	+	+	-	+	-

Run 5 is included to show the control law's ability to enable multiple arm control based on the methodologies proposed by Cleghorn and Bailey [9]. Figure 9 shows the root mean square (RMS) POR position and attitude error between arms 1 and 2 over the duration of the maneuver. Although the maximum errors of one third of an inch and half of a degree may appear slightly large, the proposed goal oriented resolved rate control law is self correcting. For the kinematic simulation of Run 5, the PORs of the two arms are not constrained to be coincident. However, for an actual hardware implementation, the PORs of the two arms would be constrained to be coincident while grasping the same object. Since the control law is resolved rate rather than resolved motion, rate sensory feedback will reduce the positional errors between the two arms, thereby reducing the opposing forces and torques. This error reduction is the self-correcting effect mentioned earlier.

As an aside, notice that the joint and joint rate time histories for joint 4 remain relatively unchanged. This can be explained by the nature of the maneuver, which is primarily translational. For this particular trajectory, joint 4 pitch is critical to attain desired POR translation.

CONCLUSIONS

A generalized control law for a spatial arm with 7 or more degrees of freedom (DOF) has been presented. This control law was derived using Lagrange multipliers to optimize a function consisting of squared joint rate terms. This function can be physically interpreted as the summed individual link rotational kinetic energies when the A weighting values are set to the link inertias.

Results from a simulation implementation utilizing this control law were presented to clarify the effects of the associated weighting matrix. Despite the variance in joint angle solutions and corresponding rates, the linear response of POR in task space is maintained. This ability is ideal for multiple arm control implementations utilizing independent control laws for each arm.

The relative rather than absolute magnitudes of the A weighting matrix are important. Specifically, preliminary results indicate the "magic 100" ratio to be a good engineering solution. Increasing a single value of the weighting matrix tends to limit motion of the corresponding joint; however, this effect is reduced when applied to a higher number of joints.

Although the current analysis is based upon Whitney's formulation, other formulations might be more applicable to minimizing practical kinematic quantities, such as joint travel. Implementation of additional formulations within the RST will be an immediate subject of future research. Integrating and analyzing collision avoidance and path planning algorithms for 7 or more DOF robotic arms is also planned.

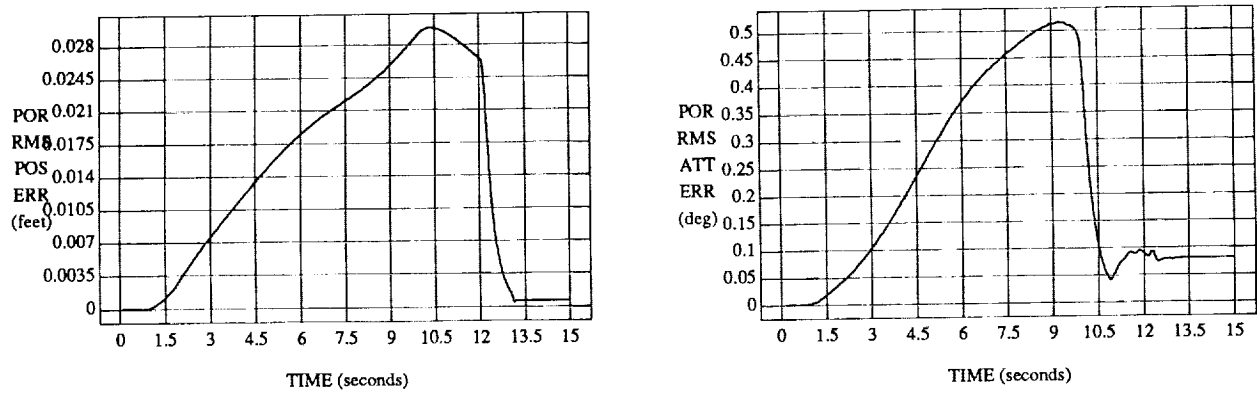


Figure 9 - RMS POR Errors Between Arms 1 and 2 for Run 5

REFERENCES

[1] Whitney, D. E., "Resolved Motion Rate Control of Manipulators and Human Prostheses," *IEEE Transactions on Man-Machine Systems*, Vol. MMS-10, No. 2, 1969, pp. 47 - 53.

[2] Whitney, D. E., "The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 122, 1972, pp. 303 - 309.

[3] Liegeois, A., "Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms, *IEEE Transactions on Systems, Man, and Cybernetics*, n Vol. SMC-7, No. 12, 1977, pp. 868 - 871.

[4] Klein, C. A., and Huang, C., "Review of Pseudoinverse Control for Use with Kinematically Redundant Manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 3, 1983, pp. 245 - 250.

[5] Bourgeois, B. J., "An Optimal Resolved Rate Law for Kinematically Redundant Manipulators," *First Annual Workshop on Space Operations Automation and Robotics (SOAR 87)*, 1987, pp. 457 - 464.

[6] Maciejewski, A. A., and Klein, C. A., "Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments," *International Journal of Robotics Research*, Vol. 4, No. 3, 1985, pp. 109 - 117.

[7] Suh, K. C., and Hollerbach, J. M., "Local versus Global Torque Optimization in Redundant Manipulators," *IEEE Conference on Robotics and Automation*, Vol. 1, 1987, pp. 619 - 624.

[8] Karlen, J. P., Thompson, J. M., and Farrell, J. D., "Design and Control of Modular, Kinematically-Redundant Manipulators," *Second AIAA/NASA/USAF Symposium on Automation, Robotics, and Advanced Computing for the National Space Program*, March 1987.

[9] Cleghorn, T. F., and Bailey, R. W., "Task Space Coordination of Multiple Robotic Arms Manipulating the same Payload in an Orbital Environment," *Satellite Servicing Workshop IV*, June 1989, proceedings to be published.

[10] Mardson, J., and Tromba, A., Vector Calculus, W. H. Freeman and Company, San Fransisco, 1981.

[11] Baker, D. R., and Wampler, C. W., "Some Facts Concerning the Inverse Kinematics of Redundant Manipulators," *IEEE Conference on Robotics and Automation*, Vol. 1, 1987, pp. 604 - 609.