Kinematically Redundant Robot Manipulators*

J. Baillieul
Scientific Systems, Inc. and Boston University
Cambridge, MA 02140

R. Brockett
Harvard University
Cambridge, MA 02138

J. Hollerbach
Massachusetts Institute of Technology
Cambridge, MA 02139

D. Martin, R. Percy, and R. Thomas
Scientific Systems, Inc.
Cambridge, MA 02140

1.0 Introduction

This paper reports research on control, design and programming of kinematically redundant robot manipulators (KRRM). These are devices in which there are more joint space degrees of freedom than are required to achieve every position and orientation of the end-effector necessary for a given task in a given workspace. The technological developments described in this paper deal with:

- Kinematic programming techniques for automatically generating joint-space trajectories to execute prescribed tasks;
- Control of redundant manipulators to optimize dynamic criteria (e.g., applications of forces and moments at the end-effector that optimally distribute the loading of actuators);
- Design of KRRMs to optimize functionality in congested work environments or to achieve other goals unattainable with non-redundant manipulators.

We discuss kinematic programming techniques, showing that some pseudo-inverse techniques that have been proposed for redundant manipulator control fail to achieve the goals of avoiding kinematic singularities and also generating closed joint-space paths corresponding to close paths of the end effector in the workspace. The extended Jacobian is proposed as an alternative to pseudo-inverse techniques. It incorporates functional constraints in a straightforward way to resolve redundancy, and can meet a variety of spatially-varying optimality criteria. This method can generate manipulator trajectories that automatically avoid obstacles provided suitable distance functions are defined, and if the intersections of the constraint surfaces are characterized in a sufficiently simple way.

2.0 Design Issues

A six degree-of-freedom geometry can no longer be considered a general purpose manipulator. This geometry has fatal kinematic flaws that arise from singularities and restrictions on the workspace. The major flaw of six degree-of-freedom manipulators is the presence of singularities in the interior of the workspace. It is exceedingly difficult to plan trajectories that do not pass through or near singularities, given the complex transformation between end effector locations and joint angles. An extra degree of freedom makes functional interior workspace points in the sense that a nonlinear configuration can be found that will correspond to a given workspace point. Singular configurations will still arise, but they can be avoided through exercise of a self-motion to arrive at a new configuration. A self-motion is created by a redundancy and is defined as an internal motion of the linkage that does not move the endpoint. The trajectory planner must still be wary of interior singularities, but upon arriving at one, the motion can backtrack so as to evolve to a different configuration at the singular point. Thus a seven degree-of-freedom represents a minimal configuration (least complex geometry) that makes available all interior workspace points.

Seven degree-of-freedom geometries are complex and costly. Most industry efforts have therefore focused on seeking methods to mitigate the effects of singularities. Strict realization of the velocity requirements at the endpoint must be abandoned. Sometimes a self-motion at the singularity can be used to find an alternative configuration for which the possible endpoint velocities happen to coincide with the desired one [1], although the manipulator must effectively come to a stop for this self-motion to occur.

3.0 Resolution of Redundancy

Redundancy resolution schemes fall into two broad categories: local optimization or global optimization techniques. Within each category, the optimization may be done at the kinematic or at the dynamic level.

*Work conducted under Contract Number F33615-84-C-5131 SBIR Phase II, USAF AFWAL/MLTC

Most research has involved the instantaneous or local resolution of the redundancy through use of the pseudo-invers These local techniques deal with the instantaneous kinematics of motion, *i.e.*, motion which is locally optimized by increme tal movement from the current arm state.

Global optimization minimizes some performance index across a whole trajectory, and hence should perform bett than local optimization. Yet the complexity of problem formulation and the computational interactibility have restricted tl use of global optimization schemes for redundant manipulators.

The advantages of the local optimization methods over global methods are twofold: the simplicity of problem formul tion and the relatively small amount of computation required for the algorithm. The small amount of computation associat with local methods offers the possibility of real-time control of the manipulators. The local technique, however, may n always be desirable for controlling redundant arms. [2] showed motions of a redundant manipulator following closed ha trajectories are generally not closed in joint space trajectories. [3] proved that, without a modification, the generalized inver method need not even avoid kinematically singular configurations. Since the local optimization method only instantaneou minimizes a given criterion, it does not guarantee a global minimum and may even result in a disastrous manipulator moti [4].

On the other hand, the global optimization technique ensures a solution with a global minimum. Real-time cont based on global techniques is problematic, due to the heavy computational requirements. The global technique may be p fectly adequate for commonly encountered industrial problems requiring repetitive motion, since a specific solution will used over and over again.

3.1 Local Kinematic Resolution of Redundancy

Most local kinematic techniques resolve redundancy at the velocity level by using the pseudo-inverse J^T (also known the Moore Penrose generalized inverse) of the Jacobian J:

$$\dot{x} = J\dot{\theta}$$

$$\dot{\theta} = J^{*}\dot{x} + (I - J^{*}J)\,\dot{\phi}$$

$$J^T = J^T(JJ^T)^{-1}$$

where

$\dot{x}$ =	6 dimensional velocity vector of the manipulator end
θ=	$n > 6$ dimensional joint angle vector
$\dot{\phi}$=	arbitrary joint vector

$(I - J^TJ)\,\dot{\phi}$ is the projection of $\dot{\phi}$ into the null space of jacob and corresponds to self-motion of the linkage that does move the end effector.

This approach is attractive in two ways. First, the pseudo-inverse has a least squares property that can minimize exc sive joint velocities and make smoother motion. Second, the redundancy that is available is succinctly characterized by null-space of the Jacobian. Measures related to this formulation can be used to achieve some objective, *i.e.*, to avoid jo limits, singularities and obstacles [5,6,7]. A weighted pseudo-inverse (different from the null-space vector) can be used angers high and low priority of variables [8].

The Moore-Penrose generalized invoice is problematic, however, in that it is nonconservative [2]. Repetitive moti planned with the pseudo-inverse alone need not follow a repetitive path in joint-space.

3.2 Global Kinematic Resolution of Redundancy

Nakamura [11] presented a method based on Pontryagin's Maximum Principle for globally optimizing a given cost fu tion for problems involving both kinematics and dynamics. An integral performance index of the following type is minimi over a desired trajectory:

$$\int_{t_i}^{t_f} p(\underline{\theta}, t)\, dt$$

where t_i and t_f are the initial and final time respectively. For example, $p = \dot{\theta}^T\dot{\theta} + kw$, where k is a constant and w is the ma pulatability index, was used by [11]. Pontryagin's Maximum Principle is then applied to Equation 4 and Equation 2 whic treated as an ordinary optimal control problem of a dynamic system with $\underline{\theta}$ as an input vector. The Hamiltonian according a fixed time problem with a fixed time problem with a fixed left hand end-point and a free right hand endpoint is given by

$$H(\underline{\psi}, \underline{\theta}, t, \underline{\dot{\phi}}) = -p + \underline{\psi}^T \underline{\dot{\theta}} \quad (5)$$

where $\underline{\psi}$ is an auxiliary variable vector. The global solution is then given by choosing a $\underline{\dot{\phi}}$ that maximizes the Hamiltonian at every instant and solving the following 2n differential equations:

$$\underline{\dot{\theta}} = \left(\frac{\partial H}{\partial \underline{\psi}} \right)^T \quad (6)$$

$$\underline{\dot{\psi}} = -\left(\frac{\partial H}{\partial \underline{\theta}} \right)^T \quad (7)$$

where Equation 6 is the same as Equation 2.

3.3 Global Kinetic Resolution of Redundancy

For problems including dynamics, a state vector $\underline{\nu} = [\underline{\theta}^T \underline{\dot{\theta}}^T]^T$ was introduced in [11]. Using the inverse kinematics at the acceleration level, the kinematics equations are rewritten in the following form:

$$\underline{\dot{\nu}} = \underline{Q}(\underline{\nu}, t) + \underline{R}(\underline{\nu})\underline{\ddot{\phi}} \quad (8)$$

$$\underline{Q}(\underline{\nu}, t) = \begin{bmatrix} \underline{\dot{\theta}} \\ J\dagger(\ddot{x}(t) - \dot{J}\underline{\dot{\theta}}) \end{bmatrix} \quad (9)$$

$$\underline{R}(\underline{\nu}) = \begin{bmatrix} \underline{\theta} \\ I - J\dagger J \end{bmatrix} \quad (10)$$

Joint torques can now be written in terms of $\underline{\nu}$, $\underline{\ddot{\phi}}$, and t as

$$\underline{\tau}(\underline{\nu}, \underline{\ddot{\phi}}, t) = \underline{U}(\underline{\nu}, t) + \underline{V}(\underline{\nu})\underline{\ddot{\phi}} \quad (11)$$

$$\underline{U}, (\underline{\nu}, t) = HJ\dagger(\underline{\ddot{x}}(t) - \dot{J}\underline{\dot{\theta}}) + \underline{\dot{\theta}}.C.\underline{\dot{\theta}} + \underline{g} \quad (12)$$

$$\underline{V}(\underline{\nu}) = H(I - J\dagger J) \quad (13)$$

An integral performance index of the following type is then minimized:

$$\int_{t_i}^{t_f} (kp_o(\underline{\nu}) + \underline{\tau}^T \underline{\tau}) dt \quad (14)$$

where k is a non-negative scalar. For example, setting k to 0 minimizes the joint torques in a least squares sense. The optimization problem can be solved through Pontryagin's Maximum Principle. The solution requires solving 4n differential equations. The algorithms used in Nakamura's dynamic method and the global algorithm presented in this paper are theoretically equivalent, but different methods are used in the formulation.

4.0 Kinematic Programming Techniques

4.1 Pseudo-Inverse Techniques for Redundancy Resolution

The practical problem associated with planning joint-space motions for kinematically redundant manipulators is that of producing an arbitrary prescribed end-effector movement. To do so, the controller must choose among infinitely many corresponding joint space movements.

For any robot, each possible joint angle configuration defines a unique position of the end effector of the robot arm. This is expressed mathematically by an equation of the form $f(\theta) = x$, where x is a vector (typically six dimensional) defining the position and orientation of the end effector, and θ is a vector defining the joint angle configuration. By differentiating both sides of the equation $x(t) = f(\theta(t))$, we obtain the kinematic relation

$$\dot{x}(t) = \frac{\partial f}{\partial \theta}(\theta(t))\,\dot{\theta}(t) \quad (15)$$

from which we can compute $\dot{\theta}(t)$ in terms of the prescribed end effector trajectory x(t). One way to uniquely specify a joint velocity vector for each $\dot{x}(t)$ is to use the Moore-Penrose inverse given by

$$\dot{\theta}_0(t) = \frac{\partial f}{\partial \theta}(\theta(t))^{+}\dot{x}(t), \tag{16}$$

The joint velocities are minimized by this technique. But since joint velocities can become arbitrarily large near singular configurations [13], this technique appears to show promise for generating joint angle trajectories that automatically avoid singular configurations. However, analysis shows the Moore-Penrose inverse technique, without further restrictions, may generate trajectories which pass arbitrarily close to singular points in joint angle space. Thus singularities are not avoided in any practical sense. This result is in contrast to some claims that have been made in literature [2].

Modifications to the Moore-Penrose pseudo-inverse technique can be made to avoid singularities. An alternative to Equation 16 for defining joint angle trajectories uses a projection operator onto the null space:

$$\dot{\theta} += \frac{\partial f}{\partial \theta}(\theta)^{+}\dot{x} + \left[I - \frac{\partial f}{\partial \theta}(0)^{+}\frac{\partial f}{\partial \theta}(\theta)\right]\nu \tag{17}$$

ν is a (time varying) vector of the same dimension as θ which remains to be specified. This modification of the Moore-Penrose pseudo-inverse technique can generate trajectories which avoid singular configurations by appropriate choice of $\nu(.)$ in Equation (6).

4.1.1 Functional Constraints for Redundancy Resolutions

A second class of methods for resolving redundancy, quite distinct from the generalized inverse methods, is that of imposing differentiable (for smooth motion) functional constraint relationships on the joint angles:

$$\phi_r(\theta_1, \theta_2, \ldots, \theta_k) = 0$$

$$\vdots$$

$$\phi_r(\theta_1, \theta_2, \ldots, \theta_k) = 0 \tag{18}$$

In general, however, it might not be possible to choose ϕ so that $(\theta_1, \theta_2, \theta_3)$ satisfy the redundancy condition $\phi(\theta_1, \theta_2, \theta_3) = 0$ and depend continuously on the coordinates (x, y) of the end effector (a 2-d example of the method using a 3-bar resolute joint, linkage in the plane). It is possible to find ϕ if some arbitrarily small area A of the workspace is excluded from the conditions, hence resolving the redundancy in a continuous way.

4.1.2 Obstacle Avoidance

An optimality criterion defined in terms of a distance function will depend on how obstacles are represented. A simple way of representing manipulator links is to model them as live segments between adjacent joint coordinate systems. Obstructions in the workspace (modeled as, *e.g.*, primitives) can then be classified according to how and which links in the mechanism can be impeded. Analysis of various geometries will then indicate the cases in which the relative dimensions of the links represent undesirable designs.

There are two major issues in incorporating considerations of obstacle avoidance into the design of kinematically redundant manipulators. First, the basic geometry of the mechanism must be specified. Then, dimensions of the manipulator must be chosen to maximize some measure of its capacity to function in a congested workspace.

Each basic manipulator geometry will require specification of a figure-of-merit. One example of such a figure of merit could be the distance a manipulator could reach behind an obstacle in the workspace, or the area excluded from the workspace because of the obstacle. These figures can be based on manipulator characteristics, workspace and obstacle dimensions, or, if this is not known at the design stage, probabilistic models or parametric analyses.

4.2 Global Optimization Techniques

Our research developed practical numerical methods for resolving redundancy and solving the inverse kinematics problem, by minimizing a global (path integral) velocity criterion. These techniques are of interest because of the form in which the solutions are expressed is similar to that of the pseudo-inverse or Extended Jacobian techniques. This can be contrasted with other numerical techniques in which a repetitive and computationally costly process is used until the solution converges:

a nominal solution is assumed, the problem is linearized, the linear optimal solution is found by a backward and forward sweep, and the linear optimal solution is used to update the nominal solution.

Our method differs from these other numerical techniques:

(1) No approximations or linearizations are required;

(2) The solution is always in the form of a differential equation whose solution is always a feasible joint space trajectory;

(3) The "optimal" solution is found by searching over a relatively small number of parameters comprising the initial conditions of the differential equation; and

(4) The computational requirements of the solution for a particular set of initial conditions are comparable to those of the pseudo-inverse or Extended Jacobian techniques.

Our approach is to view this problem as a boundary value problem (the theoretical basis for this approach is due to Nakamura [11]. We choose to use the additional freedom to minimize an integral of the joint velocities over the path:

$$\text{Minimize} \int_0^T |\dot{\theta}(t)|^2 \, dt \tag{19}$$

subject to the constraint

$$x(t) = f(\theta(t)). \tag{19a}$$

The constraint expresses the requirement the end effector follow a prescribed path in space. It is also possible to express the constraints in terms of velocities:

$$\dot{x}(t) = \frac{\partial f}{\partial \theta}\dot{\theta}(t) = J\dot{\theta}(t) \tag{20}$$

Solutions to the problem Equation 19 are obtained from use of undetermined Lagrange multipliers and the Euler-Lagrange equations, and Equation 19 becomes

$$\text{Minimize } 0 = \int_0^T L(\theta, \dot{\theta}, \lambda)\frac{dt}{\theta}, \lambda \tag{21}$$

with

$$\frac{\partial L}{\partial \theta} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) = 0, \ \frac{\partial L}{\partial \lambda} = 0 \tag{22}$$

This leads to

$$J^T\lambda - \ddot{\theta} = 0 \tag{23a}$$

and

$$f(\theta) - x = 0 \tag{23b}$$

For a kinematically redundant manipulator, the dimensions of J as such Equation 23 overdetermines λ in terms of $\ddot{\theta}$. A direct consequence of this is the relation

$$n_j^T\ddot{\theta} = 0 \tag{24}$$

where n_j is any nullspace vector of J (*i.e.*, $Jn_j = 0$, $n_j^T n_j \neq 0$). Equation 24 is the necessary condition that was sought for joint space trajectories that extremize the integral of Equation (18). A solution for λ that is consistent with Equation 24 is $\lambda = (JJ^T)^{-1}J\ \ddot{\theta}$. When we substitute this solution back into Equation 20, we have

$$(J^T(JJ^T)^{-1}J - I)\ddot{\theta} = 0 \tag{25}$$

or, equivalently, $-P_J\ddot{\theta} = 0$ where P_J is the nullspace projection operator for J. Equations 24 and 25 are equivalent when $(JJ^T)^{-1}$ exists.

Equation 24 provides a second order differential equation that requires two boundary conditions to provide a particular solution.

Analysis of this case where $\phi(0)$ and $\theta(t)$ can vary, but are subject to kinematic constraints at the endpoints, leads to the consequence

$$n_j(\theta(0))^T\dot{\theta}(0) = 0$$
$$n_j(\theta(T)^T\dot{\theta}(T) = 0 \qquad (26)$$

This is the simple statement that, when the only boundary condition on θ is the kinematic relationship, $f(\theta) = x$, then a necessary condition for the cost to be at an extremum is the component of initial and final velocity in the nullspace of J be zero.

4.2.1 Differential Equations Describing Optimal Solutions

The equations of constraint, together with the results of the Euler-Lagrange equations just presented, can be used to derive differential equations for propagating the optimal $\theta(t)$. Such solutions must simultaneously satisfy Equation 21 and the kinematic constraint, $f(\theta) = x$. We have evaluated three ways to obtain differential equations for θ that meet these conditions. They differ in the implied computational requirements and some of the techniques introduce "removable" singularities to the computation of the solution. When singular behavior is not evident, all of the techniques provide the same solution to equivalent boundary value problems. Finally, it should be emphasized these differential equations are necessary but not sufficient for an optimal value of σ in Equation 18.

4.2.2 Direct Solution

The most direct way to obtain a second order differential equation meeting the criteria listed above is to differentiate the constraint equations twice with respect to time, to obtain

$$\ddot{x} = J\ddot{\theta} + \dot{J}\theta dot, \qquad (27)$$

When the pseudo-inverse solution for $\ddot{\theta}$ in terms of $\ddot{x}$ and $\dot{\theta}$ is examined,

$$\ddot{\theta} = J\dagger\left(\ddot{x} - \dot{J}\dot{\theta}\right) \qquad (28)$$

where $J\dagger = J^T(JJ^T)^{-1}$, can one observe that this solution to Equation (27) also satisfies Equation (24), since $n_j^T J\dagger = 0$. This means a joint space trajectory integrated from Equation (28) and appropriate boundary conditions will meet the necessary condition for optimality. Note that, for this resolution to exist, $(JJ^T)^{-1}$ must exist everywhere along the trajectory. This is the equivalent to the requirement there be no kinematic singularities on the trajectory. This does not mean optimal trajectories do not include singularities; it is possible to specify boundary conditions, for example, that are kinematically singular. "Optimal" solutions for such problems exist, but they are not a consequence of Equation (24) or (27).

4.2.3 Reduced Order Solutions

In order to obtain solutions to Equation (28) one must integrate a second order differential equation in a number of variables equal to the dimension of the joint angle vector. In principle, not all of these quantities need to be integrated, as some of them are already determined by the restraint of the kinematic relationship. There are two approaches that take advantage of this situation. The first approach introduces a parameter used to resolve the redundancy explicitly. The second approach uses the nullspace velocity as its parameter. In the latter case, the parameter is not obviously related to the configuration of the manipulator at a particular time, but offers the advantage of introducing no removable or extraneous singularities in the differential equation. In the manipulators examined so far, the number of redundant degrees of freedom is one, but all methods presented can be extended to the case of multiple degrees if freedom.

Both techniques can be derived in precisely the same way, and differ only in the particular functional relationship used to resolve the redundancy.

4.2.3.1 The Reduction Resolution Technique

In the redundancy resolution (RR) technique, a redundancy resolution parameter, $\phi = l(\theta)$, is introduced to resolve the ambiguity remaining after the constraint $f(\theta) = x$ is met.

Specifying both x and ϕ should provide enough information to compute θ. A velocity relationship can be obtained by differentiation:

$$\dot{\phi} = \partial\frac{l}{\partial}\theta\dot{\theta} = m^T\dot{\theta} \qquad (29)$$

In the null space velocity approach, the additional equation is defined directly in terms of the nullspace velocity component,

$$\dot{\alpha} = n_J^T \dot{\theta} \tag{30}$$

These two equations have the same form, and the analysis of each is similar, with substitution of appropriate parameters as required.

Applying kinematic constraints on joint velocity, and solving the resulting set of equations, we obtain a second-order differential equation in a scalar parameter that represents either ϕ or α. The inverse of Extended Jacobian can provide an explicit relationship for $\dot{\theta}$ in terms of $\dot{x}$ and this scalar parameter so that the two together provide the reduced order. If n is the dimension of θ, and n-1 the dimension of x, the two relationships comprise n+2 coupled, first order nonlinear differential equations that must be integrated. This can be compared with Equation (28), which is equivalent to $2n$ differential equations.

The principle advantage of the RR approach is that ϕ is simply related to the configuration of the manipulator, and can be found directly. The principle disadvantage of the RR approach is that many "optimal" trajectories, depending on the particular conditions or boundary values, encounter singularities under certain conditions of the parameters. The Extended Jacobian technique removes this singularity algebraically and there is, then, the possibility further work with the RR technique can eliminate this disadvantage.

4.2.3.2 The Nullspace Velocity (NV) Technique

The alternative technique to the RR technique just described is the resolution of the redundancy by a velocity constraint, in particular, the specification of the velocity component in null space. An advantage to this approach is the lack of the "removable" singular points associated with the RR technique. The NV technique uses the same basic information used in the RR technique, rather than ϕ and $\frac{\partial l}{\partial \theta}$. The computational cost of integrating a particular solution from specified initial conditions using the NV formulation requires an amount of computation that is at least comparable to the pseudo-inverse and Extended Jacobian techniques.

The disadvantage of the NV technique, relative to the RR technique, is the parameter α has little to do with the configuration of the manipulator at any given time. Its first derivative, $\dot{\alpha}$, is related to the nullspace velocity. By implication, one might assume α is related to the nullspace velocity. By implication, one might assume α is related to some distance traveled in the nullspace direction, but this is a path dependent integral, so α need not necessarily take on the same value for the same manipulator configuration if the trajectories are not identical. One available option is to integrate a subsidiary equation, such as $\dot{\phi} = m^T\dot{\theta}$, rather then integrating $\dot{\alpha}$ to obtain α, since α is not required in the formulation. This would provide a history of the self-motion of the manipulator over the trajectory.

4.3 Boundary Value Problems

With the computationally efficient methods for obtaining solutions to Problem (15) in hand, the next issue is that of obtaining particular solutions associated with given initial conditions or boundary values. The sections that follow will pose each type of boundary problem in turn, and provide a numerical method for obtaining solutions to the problem.

4.3.1 Initial Boundary Value Problem (IBVP)

The initial boundary value problem is the simplest problem. The initial orientation and velocity of the manipulator is specified by the user, subject to the kinematic constraints. It is useful to specify the initial joint angles with a redundancy resolution parameter or parameters to avoid imposing a requirement on the user to specify a full joint angle set consistent with the kinematic restraint. This allows the user to specify the workspace position and manipulator orientation in its self-motion at that position independently, rather than forcing the user to compute a joint angle set corresponding to the desired configuration. The initial position, then, is specified by the kinematic constraint in conjunction with a user-specified initial value of θ.

The "optimality" of the solutions generated by all the initial value techniques presented must be verified. This is a direct consequence of the fact the Euler-Lagrange equations from which they are derived are only necessary, but insufficient, conditions for optimality. A solution generated from an arbitrary set of initial conditions may well be a locally maximum cost solution, or may correspond to a solution that is first order stationary, but for which large changes in trajectory produce lower cost solutions.

4.3.2 Natural Boundary Value Problem

The "natural" boundary value problem occurs when there are essentially no conditions on the configuration of the manipulator at either endpoint, and we wish to find initial and final configurations that yield the least-cost solution. A necessary condition for the solution to the natural boundary value problem is the nullspace joint velocity be zero at the initial and final configurations.

The approach developed to solve this boundary value problem uses the solution to the IBVP. The NVBP solution, then, can be reduced to finding the zeros of a function that is computed by solving the IBVP. This approach provides solutions that satisfy the necessary conditions for optimum solutions to the NBVP, but to find the actual optimum all solutions must be examined.

The computational requirements imposed by the requirement to examine the entire range of solutions to the IBVP is obvious. The worst case computation cost of the solution can be immense. Rather than integrating the NV equations once, as was required for the IBVP, the NBVP requires, in principle, infinitely many such evaluations. However, many practical motion profiles give rise to a relatively smooth function for the nullspace velocity component $\dot{\alpha}^T$, and the zeros of this function can be isolated with a small number of evaluations of the IBVP.

A final aspect of this solution technique is poor performance, as might be expected, when the initial (or final) configuration is itself near a kinematic singular point.

4.3.3 Two Point Boundary Value Problem (TPBVP)

Solutions to the two point boundary value problem can be obtained by a method analogous to that used for the NBVP. In this problem, ϕ_0 and ϕ_T, or equivalent information is given. The solution to Equation (19) is required and can be found by making use of the IBVP solution. The TPBVP approach takes ϕ_0 as the configuration initial condition and searches for a velocity initial condition, $\dot{\phi}_0$, leading to a solution with ϕ_T as the final value of ϕ.

In general, it is likely that ϕ will completely resolve the redundancy. Specification of additional parameters should allow θ_T to be known unambiguously.

4.3.4 Periodic Boundary Value Problem (PVBP)

This is the problem of finding the least cost periodic motion for $\theta(t)$ corresponding to a workspace motion $x(t)$ that is also periodic, or cyclic. That is, we have a situation where $x(0)=x(T)$, and we wish to find $\theta(t)$ that is a solution to the problem of Equation (1), and meets the additional constraints $\theta(0)=\theta(T)$ and $\dot{\theta}(0)=\dot{\theta}(T)$. This results in a joint angle time history that follows the desired trajectory, is periodic, and is low cost in the sense of Equation 19.

This problem differs from the previous boundary problems as it requires a search in two variables, ϕ_0 and α_0, for the simultaneous zeros of two expressions that specify the problem. Intersections of plots of solutions to these expressions will correspond to solutions, but spurious solutions will have to be rejected.

4.4 Summary

This section has presented a new technique for generating globally optimal solutions (in a velocity-magnitude squared sense) to the inverse kinematics of redundant manipulators, due to Nakamura. The section discussed the computational requirements of the techniques and showed derivations of two reduced order methods. It presented solutions related four different types of boundary problems. The techniques presented are a practical off-line means of finding good solutions to the inverse kinematics of redundant manipulators.

5.0 Dynamics and Control

This section presents a local and a global optimization method for minimizing torque leading at the joints in the least-squares sense. The local optimization technique minimizes torque by specifying a null space vector using a generalized inverse applied to accelerations. The local method is compared to a straightforward pseudo-inverse and an inertial-weighted pseudo-inverse. The global optimization method is formulated through the use of calculus of variations, and is compared with the local algorithms.

5.1 Local Torque Optimization

Redundancy resolution using local torque optimization can reduce joint torque and avoid joint torque limits throughout the manipulator movement. An effective approach is to keep the joint torques close to the midpoint of their upper and lower torque limits. This is done in a least squares sense by minimizing a vector that combines a vector of the upper limits of the joint torques and the vector of lower limits. For simplicity, these limits are assumed motion independent. The idea of different available torque ranges is easily solved by using a weighting matrix with proper representation of the available torque ranges.

The algorithms we investigated are:

- Unweighted pseudo-inverse algorithm (UPI)

$$\underline{\tau} = HJ\dagger(\ddot{x} - \dot{J}\underline{\dot{\theta}}) + \underline{C} + \underline{g} \tag{31}$$

- Inertia-weighted pseudo-inverse algorithm (IWPI)

$$\underline{\tau} = HJ^{\dagger}_{H}(\ddot{x} - \dot{J}\underline{\dot{\theta}}) + \underline{C} + \underline{g} \tag{32}$$

- Unweighted null-space algorithm (UNS)

$$\underline{\tau} = HJ\dagger\left(\underline{\ddot{x}} - \dot{J}\underline{\dot{\theta}}\right) + \underline{C} + \underline{g} + H\,[\,H(I - J\dagger J)\,]\dagger\frac{\left(\underline{\tau}^{+} + \underline{\tau}^{-}\right)}{2} \tag{33}$$

- Weighted null-space algorithm (WNS)

$$\tau = HJ\dagger(\,\underline{\alpha} - \dot{J}\underline{\dot{\theta}}\,) + \underline{C} + \underline{g} + H\,[\,W^{\frac{1}{2}}H(\,I - J^{\frac{1}{2}}J\,)\,]\dagger\,W^{\frac{1}{2}}(\underline{\tau}^{+} + \underline{\tau}^{-})\frac{}{2} \tag{34}$$

The unweighted pseudo-inverse algorithm derives the joint torques without the null space component yielding a solution with minimum $\dot{\theta}^T\dot{\theta}$. Presumably, this should keep joints from moving too fast if started at rest, possibly yielding a more controllable motion. The inertia weighted pseudo-inverse algorithm [9, 10] yields a minimum kinetic energy solution. The unweighted and weighted null-space algorithms are the proposed methods presented in the previous section.

5.1.1 Results

Performances of the unweighted null-space (UNS), unweighted pseudo-inverse (UPI) and inertia-weighted pseudo-inverse (IWPI) algorithms were compared for representative trajectories and assumed characteristics of a basic three-link planar rotary manipulator.

For a short movement, the UNS dramatically reduces the joint torques over the UPI, with the IWPI falling somewhere between the two. A dramatic reduction in joint torque of the UNS is the main contribution to the overall increase in performance. For a medium length movement, the UNS still shows a dramatic reduction over the USI, with the IWPI again falling in between.

The situation changes considerably for a long movement. Both the UNS the IWPI algorithms show unexpected instability near the end of the movement. The instability seems to be caused by the alignment of the second and third links and the large joint velocities associated at the time of alignment. The redundancy of the arm is partially lost in the first joint at the alignment, and the large joint velocities require extremely large joint torques to keep the manipulator on the desired trajectory. Evidently, the UNS and IWPI algorithms always show instablility for relatively long trajectories.

The UPI algorithm appears to be more stable. There were a few trajectories where only the WPI showed the instability. The UPI algorithm goes through a partial loss of redundancy in the third joint near the movement midpoint, and another loss of redundancy in the second joint near the end of the movement. These losses of partial redundancy together with the large joint velocities seemed to have caused the instability of the UPI algorithm.

In the WNS for the same trajectories, the third joint torque is pulled much closer to its midpoint at the expense of the first and second joints. However, all the joint torques are well within their ranges. Unfortunately, the WNS also shows this instability in long movements. There were even movements where the instability is shown only by the WNS. The characteristics of the instability in the weighted case were identical to those of unweighted cases.

5.1.2 Discussion

Local algorithms show dramatic improvement over the unweighted and pseudo-inverse algorithms in trajectories of short length. However, for long trajectories, the null-space algorithms and the IWPI algorithm all have stability problems. Only the UPI algorithm was generally well-behaved, although it too showed instabilities.

It seems local tampering with the energetics of movement leads to global disaster. The instability shown by the IWPI and the UNS and WNS seems to be caused by the line-up of links 2 and 3 together with high joint velocities at this configuration. Since the use of the null-space vector adds to the joint acceleration vector, the joint torques are minimized at the cost of large joint velocities. These large joint velocities eventually caused the manipulator's second and third links to line up, resulting in a partial loss of redundancy with the inability of joint 1 torque to vary. The UPI the instability less often. Since the UPI gives a solution of minimum joint accelerations in a least square sense, the joint velocities are restrained from causing links 2 and 3 to line up. Nevertheless, the UPI can go through another type of manipulator configuration with partial loss of redundancy and eventually go unstable.

5.2 Global Optimization

The undesirable behavior of the local optimization techniques has led to development of a global method for optimizing joint torques. The method parameterizes the redundancy of a manipulator and uses the calculus of variations. This formulation requires explicit inverse kinematic solutions and extra time derivatives of the variables involved, as opposed to Nakamura's [11] use of Lagrange multipliers and Pontryagin's Maximum Principle. However, only a single fourth-order ordinary differential equation needs to be solved, instead of 4n elementary differential equations required by Pontryagin's Principle for a manipulator with one degree of redundancy. The global optimization algorithm is formulated using a variable ϕ that parameterizes the redundancy of the manipulator. The hand variables x with ϕ specifies a joint configuration. Therefore, given a desired trajectory $\underline{x}(t)$, the corresponding trajectory of the manipulator can be solved in terms of $\underline{x}(t)$ and $\phi(t)$.

The objective of this technique is to place the joint torques closest to the midpoint of the joint torque movements over the entire movement. This is done in a least squares sense by minimizing an integral of joint torques over the entire trajectory. The performance index to be optimized is expressed as a function of ϕ and its first two derivatives and t over the time of movement. The problem is to find a $\phi(t)$ that minimizes this performance index. This is a straight-forward problem in calculus of variations whose solution is given by an Euler-Lagrange equation with appropriate boundary conditions.

The Euler-Lagrange equation can be expressed as fourth-order ordinary differential equation in ϕ, and four boundary conditions are needed to solve for the optimal solution. Two of these are readily obtained from the initial manipulator configuration, namely $\phi(0)$ and $\dot{\phi}(0)$. The remaining two conditions are given by the transversality condition at $t = t_f$. The problem then becomes a two point boundary problem which can be solved numerically. However, since there are only two unknown initial values, $\ddot{\phi}(0)$ and $d^3\frac{\phi}{d}t^3(0)$, the space of these two unknown values can be searched for the optimal solution. Various well-known initial value integration methods may then be used to search for the solution with the minimal performance index.

5.2.1 Results

As expected, the global solution of the short movement closely resembles the UNS algorithm. However, for trajectories where the UNS showed instability, global solutions more closely resemble the stable UPI solution.

5.2.2 Discussion

The unacceptable performance of the local algorithm in minimizing actuator demands over the whole trajectory has led to the development of a global algorithm formulated through the parameterization of the redundancy and the method of calculus of variations. (Even though the global methods are computationally infeasible for real-time control, they could be used in repetitive motions commonly found in industry.) Results of the global algorithms are very promising; a solution was found that outperformed all the local algorithms in movements of all lengths, even those long movements where the local algorithm showed the instability.

Whether the local kinematic methods may be modified to avoid the instabilities is not known. One possibility is to weight the local optimization criterion with a kinematic term to avoid high velocity buildup. For staying within torque bounds, linear programming, rather than least squares, may be more satisfactory. The broader question is whether any local algorithms can ever be completely successful, or whether ultimately only a global resolution of redundancy can be guaranteed problem-free.

More questions should be addressed before these algorithms can actually be applied to the control of redundant manipulators. First, the joint angle, velocity and torque limits cannot be enforced with the algorithms formulated. The global algorithm may be formulated in terms of Pontryagain's Maximum Principle to incorporate the joint torque limits; however, the joint angle and velocity limits still cannot be addressed. State-space search may have to be used for the enforcement of all manipulator constraints. Second, even though the manipulator starts are rest and ends at zero hand velocity, the resulting joint velocities may not be zero; that is, the manipulator continues to move at the end of the movement. The algorithms may continuously be applied at the movement end to keep the hand from moving; however, this does not guarantee the arm will eventually come to rest. For any reasonable tasks, this is highly undesirable; therefore, the algorithms should be modified in some ways so the manipulator comes to a complete stop or a desired configuration at the movement end.

6.0 References

[1] Sugimoto, K. and Duffy, J., 1981, "Special Configurations of Industrial Robots," Proc. 11th Int. Symp. Industrial Robots, Tokyo, October 7-9, pp.309-316

[2] Klein, C.A., and Huang, C.H., 1983, Review of Pseudo-Inverse Control for Use with Kinematically Redundant Manipulators," IEEE Trans. Systems, Man, Cybern, SMC-13, pp. 245-250.

[3] Baillieul, J., 1985, "Kinnematic Programming Alternatives for Redundant Manipulators," Proc. IEEE Conf. on Robotics and Automation, San Francisco, April 7-10, pp. 1698-1704.

[4] Chang, P.H., 1986, "A Closed-Form Solution of the Control of Manipulators with Kinematic Redundancy," Proc. IEEE Inc. Conf. Robotics and Automation, pp. 9-14.

[5] Liegois, A., 1977, "Automated Supervisory Control of the Configuration and Behavior of Multibody Mechanisms," IEEE Trans. Systems, Man, Cybern., SMC-7, pp. 868-871.

[6] Yoshikawa, T., 1984, "Analysis and Control of Robot Manipulators with Redundancy," Robotics Research: The First International Symposium, edited by M. Brady and R. Paul, Cambridge, MA, MIT Press, pp.735-748.

[7] Klein, C.A., 1985, "Use of Redundancy in the Design of Robotic Systems," Robotics Research: The Second International Symposium, edited by H. Hanafusa and H. Inoue, Cambridge, MA, MIT Press, pp. 207-214.

[8] Hanafusa, H., Yoshikawa, T., and Nakamura Y., 1981, "Analysis and Control of Articulated Robot Arms with Redundancy," Prep. 8th IFAC World Congress, August, pp. XIV-78-83.

[9] Khatib, O., 1980, Commande Dynamique dans l'Espace Operational des Robots Manipulateurs en Presence d'Obstacles, Docteur Ingenieur Thesis, L'Ecole Nationale Superieure de l'Aeronautique et de l'Espace.

[10] Khatib, O., 1983, "Dymanic Control of Manipulators in Operational Space," 6th IFToMM Congress on Theory of Machines and Mechanisms, New Delhi, December 15-20, pp. 1123-1131.

[11] Nakamura, Y., 1985, Kinematic Studies on the Trajectory Control of Robot Manipulators, PhD. Thesis, Kyoto University, June.

[12] Nakamura, Y. and Hanafusa,H., 1985, "Task Priority Based Redundancy Control of Robot Manipulators," Robotics Research: The Second International Symposium, edited by H. Hanafusa and H. Inoue, Cambridge, MA, MIT Press, pp. 155-162.

[13] Baillieul, J. et al, 1984, "Programming and Control of Kinematically Redundant Robot Manipulators," Proc. 23rd IEEE Conference on Decision and Control, Las Vegas, NV, December 12-14, pp. 768-774.