Open access • Journal Article • DOI:10.1109/3468.844350

On the implementation of velocity control for kinematically redundant manipulators

James D. English, Anthony A. Maciejewski

Institutions: Purdue University

Published on: 01 May 2000 - Systems, Man and Cybernetics

Topics: Jacobian matrix and determinant, Linear system, Basis (linear algebra) and Vector-valued function

On the Implementation of Velocity Control for Kinematically Redundant Manipulators

James D. English and Anthony A. Maciejewski

***Abstract*—The velocity control of kinematically redundant manipulators has been addressed through a variety of approaches. Though they differ widely in their purpose and method of implementation, most are optimizations that can be characterized by Liégeois's method. This characterization is used in this article to develop a single framework for implementing different methods by simply selecting a scalar, a function of configuration, and a joint-rate weighting matrix. These quantities are used to form a fully constrained linear system by row augmenting the manipulator Jacobian with a weighted basis of its nullspace and augmenting the desired hand motion with a vector function of the nullspace basis. The framework is shown to be flexible, computationally efficient, and accurate.**

***Index Terms*—Control, kinematically redundant, kinematics, manipulators, redundant robots/manipulators, velocity control.**

I. Introduction

FOR a given manipulator's vector of joint values $\boldsymbol{q}$ and a given representation of hand motion $\dot{\boldsymbol{x}}$, a corresponding manipulator Jacobian $\boldsymbol{J}(\boldsymbol{q})$ exists such that

$$\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}}. \tag{1}$$

When the dimension of the task space (the dimension of $\dot{\boldsymbol{x}}$), m, is less than the number of joints, n, (1) is underdetermined and the manipulator is kinematically redundant.

A well-known general method for resolving this redundancy was presented by Liégeois [1]. His method is to cast a secondary desideratum in the form of a vector of joint rates $\boldsymbol{z}$, then find joint rates $\dot{\boldsymbol{q}}$ that approximate $\boldsymbol{z}$ while producing a prescribed $\dot{\boldsymbol{x}}$ using

$$\dot{\boldsymbol{q}} = \boldsymbol{G}_1\dot{\boldsymbol{x}} + (\boldsymbol{I} - \boldsymbol{G}_2\boldsymbol{J})\boldsymbol{z} \tag{2}$$

where $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ are generalized inverses of $\boldsymbol{J}$.

Liégeois's method will be used as the foundation for the implementation framework suggested in this article. It will be shown that many well-known velocity-control techniques can be cast using a particular representation of (2) (exceptions are methods which do not give the exact prescribed $\dot{\boldsymbol{x}}$, such as damped least-squares methods [2], [3] and Jacobian transpose schemes [4]). This will allow easy implementation and evaluation of the various velocity-control techniques.

Manuscript received July 5, 1997; revised January 23, 2000. This work was supported by a NASA graduate student research fellowship (Grant NGT9-2) and by Sandia National Laboratories under Contract AL-3011. This paper was recommended by Associate Editor W. A. Gruver.

J. D. English is with MÄK Technologies, Cambridge, MA 02139 USA (e-mail: jde@mak.com)

A. A. Maciejewski is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285 USA.

Publisher Item Identifier S 1083-4427(00)03704-8.

II. The Framework

Let $\boldsymbol{W}(\boldsymbol{q})$ be a symmetric positive-definite weighting matrix defining a joint-rate measure through $\dot{\boldsymbol{q}}^T\boldsymbol{W}\dot{\boldsymbol{q}}$, $f(\boldsymbol{q})$ be a function (or measure) of the joint values, and α be a scalar. Then Liégeois's method can be restricted without changing its ability to represent most velocity-control methods by assigning $\boldsymbol{z}$ as

$$\boldsymbol{z} = -\alpha\boldsymbol{W}^{-1}\nabla f \tag{3}$$

where ∇f is the gradient of f, and by assigning $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ as

$$\boldsymbol{G}_1 = \boldsymbol{G}_2 = \boldsymbol{G} \tag{4}$$

a $\boldsymbol{W}$-weighted generalized inverse.[1]

Weighted generalized inverses have been applied to robotics for some time [6], and recent coverage is given in [7] and [8]. Provided the Jacobian has full row rank, the unique $\boldsymbol{W}$-weighted generalized inverse can be calculated using

$$\boldsymbol{G} = \boldsymbol{W}^{-1}\boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{W}^{-1}\boldsymbol{J}^T)^{-1}. \tag{5}$$

When $\boldsymbol{J}$ has full row rank, the solution from (2)–(5) is equal to that from the following:

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{J} \\ \hline \boldsymbol{N}_J^T\boldsymbol{W} \end{bmatrix}^{-1} \begin{bmatrix} \dot{\boldsymbol{x}} \\ \hline -\alpha\boldsymbol{N}_J^T\nabla f \end{bmatrix} \tag{6}$$

where $\boldsymbol{N}_J$ is any $n \times (n-m)$ matrix whose columns are a spanning set of the null space of $\boldsymbol{J}$. This is established through (1) and left-multiplying (2) by $\boldsymbol{N}_J^T\boldsymbol{W}$.

Equation (6) is a framework for velocity control, and its application extends beyond simple imitation of (2)–(5). Provided

$$\boldsymbol{N}_J^T\boldsymbol{W}\boldsymbol{N}_J > \boldsymbol{0} \tag{7}$$

i.e., $\boldsymbol{N}_J^T\boldsymbol{W}\boldsymbol{N}_J$ is positive definite, (6) can be used even when $\boldsymbol{W}$ is not positive definite. This is the strength of (6) that allows it to be applied to many existing methods.

When (7) holds and $\boldsymbol{J}$ has full row rank, it follows from the Lagrange Multiplier Theorem that (6) gives a $\dot{\boldsymbol{q}}$ that minimizes $(1/2)\dot{\boldsymbol{q}}^T\boldsymbol{W}\dot{\boldsymbol{q}} + \alpha\dot{f}$ while satisfying $\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}}$.[2] Thus, (6) makes a tradeoff based on α between minimizing the joint-*rate* measure and extremizing the derivative of the joint-*value* measure.

[1] A $\boldsymbol{W}$-weighted generalized inverse $\boldsymbol{G}$ satisfies the following: If $\boldsymbol{J}\dot{\boldsymbol{q}} = \dot{\boldsymbol{x}}$ has a solution for $\dot{\boldsymbol{q}}$, the unique solution minimizing $\dot{\boldsymbol{q}}^T\boldsymbol{W}\dot{\boldsymbol{q}}$ is given by $\boldsymbol{G}\dot{\boldsymbol{x}}$ [5].

[2] Similarly, ∇f could be replaced by any vector function of $\boldsymbol{q}$ to optimize a general quadratic criterion function subject to $\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}}$.

TABLE I
FOR n DEGREES OF FREEDOM AND m DEGREES OF TASK SPACE, $n > m$, THE COMPUTATIONAL COST OF THE METHODS GIVEN BY (2)–(5) AND BY (6) ONCE $\boldsymbol{J}$, $\boldsymbol{W}$, AND ∇f ARE KNOWN. VALUES ARE GIVEN FOR $\boldsymbol{W} = \boldsymbol{I}$ AND FOR GENERAL $\boldsymbol{W}$. THESE COSTS ARE BASED ON THE EFFICIENCIES MENTIONED IN THE TEXT

Method	multiplies/divides	adds/subtracts
(2)–(5), $\mathbf{W} = \mathbf{I}$	$\frac{2m+9m^2+m^3+6n+15mn+3m^2n}{6}$	$\frac{-10m+3m^2+m^3+15mn+3m^2n}{6}$
(2)–(5), any $\mathbf{W}$	$\frac{2m+9m^2+m^3+8n+18mn+3m^2n+9n^2+3mn^2+n^3}{6}$	$\frac{3m^2-10m+m^3-7n+12mn+3m^2n+6n^2+3mn^2+n^3}{6}$
(6), $\mathbf{W} = \mathbf{I}$	$\frac{-6m-9m^2+3m^3+4n+9mn-9m^2n+6n^2+6mn^2+2n^3}{6}$	$\frac{3m^2+3m^3-5n-3mn-9m^2n+3n^2+6mn^2+2n^3}{6}$
(6), any $\mathbf{W}$	$\frac{-6m-9m^2+3m^3+4n+9mn-9m^2n+6n^2+8n^3}{6}$	$\frac{3m^2+3m^3-5n+3mn-9m^2n-3n^2+8n^3}{6}$

III. ESTABLISHED VELOCITY-CONTROL METHODS

The last section presented a framework for velocity control. Its generality will be demonstrated here by casting existing velocity-control methods into its form.

For $\boldsymbol{G}$ set numerically to the Moore–Penrose pseudoinverse $\boldsymbol{J}^+$, Liégeois's method has been used for joint-limit avoidance [1], [9]; singularity avoidance [10]; torque minimization [11]; obstacle avoidance [12], [13]; fault tolerance [14], [15]; and many other applications. A discussion is given in [16]. For these, (6) can be used with $\boldsymbol{W} = \boldsymbol{I}$ and the respective choice of α and f. Note that $\boldsymbol{W} = \boldsymbol{I}$, though useful for examples and conceptual experiments, in practice is rarely ideal. It implies an equality between the effects of the joints that a practical manipulator design usually precludes.

The augmented Jacobian technique [17], [18], for which the gradients of $n - m$ functions are augmented to $\boldsymbol{J}$ and their desired derivatives are augmented to $\dot{\boldsymbol{x}}$, can be cast for functions $g_i(\boldsymbol{q})$ with desired derivatives of $\overline{g}_i$ as

$$\boldsymbol{W} = \sum_{i=1}^{n-m} \gamma_i \nabla g_i (\nabla g_i)^T \tag{8}$$

$$f = \sum_{i=1}^{n-m} \gamma_i \overline{g}_i g_i \tag{9}$$

and

$$\alpha = -1 \tag{10}$$

where the γ_i's are any positive scalars. Using $\boldsymbol{W}$ of the form given in (8) with $\alpha = 0$ gives an algorithm that inherits the repeatability properties of the corresponding augmented Jacobian forms [19].

Baillieul's extended Jacobian technique [20] ([21] for general redundancy), which maintains a function $g(\boldsymbol{q})$ at an extremum, can be cast as

$$\boldsymbol{W} = \left(\frac{\partial(\overline{\boldsymbol{N}}_J^T \nabla g)}{\partial \boldsymbol{q}} \right)^T \frac{\partial(\overline{\boldsymbol{N}}_J^T \nabla g)}{\partial \boldsymbol{q}} \tag{11}$$

e.g., $\boldsymbol{W} = \nabla(\overline{\boldsymbol{N}}_J^T \nabla g)(\nabla(\overline{\boldsymbol{N}}_J^T \nabla g))^T$ for a single degree of redundancy, and

$$\alpha = 0 \tag{12}$$

where $\overline{\boldsymbol{N}}_J(\boldsymbol{q})$ is an $n \times (n-m)$ function of configuration whose columns are differentiable and form a spanning set of the null space of $\boldsymbol{J}$. This $\overline{\boldsymbol{N}}_J$ need not equal $\boldsymbol{N}_J$ as used in (6). Setting α to a nonzero value and $f = g$ allows a feedback term to be introduced that will drive g specifically to a minimum or a maximum.

Other weightings amenable to (6) include $\boldsymbol{W} = \boldsymbol{K}_\theta$; a joint compliance matrix [22] $\boldsymbol{W} = \boldsymbol{M}$; the manipulator inertia matrix [$(1/2)\dot{\boldsymbol{q}}^T \boldsymbol{M} \dot{\boldsymbol{q}}$ measures kinetic energy] [6], [23], [24]; and $\boldsymbol{W} = \boldsymbol{K}_p$, where $\boldsymbol{K}_p$ measures the kinetic energy of a virtual load [25]. The modified Moore–Penrose solution of Mussa-Ivaldi and Hogan [26] uses $\boldsymbol{W} = \boldsymbol{C}^{-1} - \boldsymbol{\Gamma}$ where $\boldsymbol{C}$ is a compliance matrix and $\boldsymbol{\Gamma}$ is a matrix of second derivatives of the transformation from configuration values to end-effector coordinates. These are just a few of the joint-rate weightings that have been proposed and can be used in (6).

A strength of (6) is its ability to implement many of these approaches without explicitly calculating $\boldsymbol{W}$ or ∇f. For the augmented Jacobian technique with a single degree of redundancy, $\boldsymbol{N}_J^T \boldsymbol{W}$ can be calculated as $(\boldsymbol{N}_J^T \nabla g)(\nabla g)^T$, for example. And when $\boldsymbol{W} = \boldsymbol{M}$, the manipulator inertia matrix, $\boldsymbol{N}_J^T \boldsymbol{W}$ can be calculated by treating the columns of $\boldsymbol{N}_J$ as joint accelerations and calculating the resulting joint torques with an inverse-dynamics algorithm. Directional derivatives of f along the columns of $\boldsymbol{N}_J$ can be used to find $\boldsymbol{N}_J^T \nabla f$ in lieu of explicit calculation. For problems in which $\boldsymbol{W}$ or ∇f are very costly to evaluate, the ability to find the solution without them can be the most important characteristic of (6).

Methods which are cast using a nonpositive-definite $\boldsymbol{W}$ can be made more robust by moderating—or damping—the joint rates. A damped version of the augmented Jacobian method with one degree of redundancy could use $\boldsymbol{W} = \tilde{\boldsymbol{W}} + \nabla g (\nabla g)^T$, and a damped version of Baillieul's method for one degree of redundancy could use $\boldsymbol{W} = \tilde{\boldsymbol{W}} + \nabla(\boldsymbol{n}_J^T \nabla g)(\nabla(\boldsymbol{n}_J^T \nabla g))^T$, with $\tilde{\boldsymbol{W}}$ a positive-definite matrix with a small relative norm. This is a method to address the occurrence of algorithmic singularities. Examples of using this damping technique to reduce task errors are given in [27].

IV. COMPUTATIONAL COSTS

To compare the computational costs of the method using (2)–(5) and that using (6), a floating point count[3] is used here. This count is based on the cost after $\boldsymbol{J}$, $\boldsymbol{W}$, and ∇f are explicitly established. Explicit calculation is not always necessary for (6), as discussed previously. The results are given in Table I. Note the normal-equation method with (5) is a representative example of one of many ways to solve (2) with

[3]A floating-point count is acknowledged as an imperfect measure of implementation cost and is used here only as a rough gauge of the algorithm.

(4) [28], [29]. It is used here for comparative purposes because it is relatively fast when efficiently applied and is frequently discussed in robotics literature.

In establishing Table I, efficiencies were exploited for both techniques. For (2)–(5), the optimized procedure of Klein and Huang [16] was modified to incorporate a weighting matrix:

- Cholesky decomposition was used to find $\boldsymbol{L}$ giving $\boldsymbol{W} = \boldsymbol{L}\boldsymbol{L}^T$;
- $\boldsymbol{L}$ was used to form $\hat{\boldsymbol{J}} = (\boldsymbol{L}^{-1}\boldsymbol{J}^T)^T$ and $\hat{\boldsymbol{z}} = \boldsymbol{L}^{-1}(-\alpha\nabla f)$ using forward substitution on the rows of $\boldsymbol{J}$ and ∇f;
- the symmetry of $\hat{\boldsymbol{J}}\hat{\boldsymbol{J}}^T$ was exploited in its calculation from $\hat{\boldsymbol{J}}$;
- Cholesky decomposition was used to solve $\boldsymbol{y}$ from $(\hat{\boldsymbol{J}}\hat{\boldsymbol{J}}^T)\boldsymbol{y} = \dot{\boldsymbol{x}} - \hat{\boldsymbol{J}}\hat{\boldsymbol{z}}$;
- a final backsubstitution using $\boldsymbol{L}^T$ was performed on $\hat{\boldsymbol{J}}^T\boldsymbol{y} + \hat{\boldsymbol{z}}$ to find $\dot{\boldsymbol{q}}$.

And for (6), the following were performed:

- an LU decomposition with column pivoting was performed on rectangular $\boldsymbol{J}$ ($\boldsymbol{L}$ is $m \times m$ lower triangular, $\boldsymbol{U}$ is $m \times n$ upper triangular);
- the matrix $\boldsymbol{N}_J$ was found by assigning rows $m+1$ through n to equal the rows of the $(n-m)\times(n-m)$ identity matrix and using backsubstitution with $\boldsymbol{U}$ to complete each column of $\boldsymbol{N}_J$ to a null-space vector;
- the LU decomposition of $\boldsymbol{J}$ was extended with the rows formed by $\boldsymbol{N}_J^T\boldsymbol{W}$;
- this final LU decomposition was used to solve $\dot{\boldsymbol{q}}$ using a forward substitution then a backsubstitution on $[\dot{\boldsymbol{x}}^T | -\alpha(\nabla f)^T\boldsymbol{N}_J]^T$.

Computations involving the weighting matrix were excluded from the count for the $\boldsymbol{W} = \boldsymbol{I}$ cases.

For $n = 7$ and $m = 6$, common values for commercially available redundant manipulators, the floating-point counts are as given in Table II. Here, for $\boldsymbol{W} = \boldsymbol{I}$, (6) requires less than 60% of the cost of (2)–(5). The difference is even more significant when $\boldsymbol{W} \neq \boldsymbol{I}$. Equation (6) requires fewer floating-point operations for a general $\boldsymbol{W}$ than (2)–(5) with $\boldsymbol{W} = \boldsymbol{I}$.

When $\boldsymbol{W} = \boldsymbol{I}$, (6) is similar in concept to the dual projection method [30] and the projection methods given in [31] for one degree of redundancy and [32] and [33] for general redundancy. Though somewhat faster, it is comparable with these in its computational cost. Also, note that null-space bases have been previously used as an explicit way to complement particular solutions [34].

V. Numerical Accuracy

Equation (6) tends to suffer less from numerical errors than (2)–(5) because conditioning problems with $\boldsymbol{J}$ are compounded in forming $\boldsymbol{J}\boldsymbol{W}\boldsymbol{J}^T$. For example, if $\boldsymbol{W} = \boldsymbol{I}$, then the two-norm condition number of $\boldsymbol{J}\boldsymbol{W}\boldsymbol{J}^T$ equals the square of the condition number of $\boldsymbol{J}$.

To illustrate the relative numerical accuracy of the two methods, a planar three-link manipulator with unit link lengths is used here in a simulation experiment. For the task of planar positioning, 10 000 pseudorandom configurations and unit-norm end-effector velocity directions were chosen. Then both methods were used with $\boldsymbol{W} = \boldsymbol{I}$ and $\alpha = 0$ to calculate, in single precision, joint rates to achieve the desired velocity. The two norm of the error given by the difference between the desired and the actual velocity was calculated, and the values are shown in Fig. 1. The statistics for these results are given in Table III, showing (6) to be more accurate.

TABLE II
Floating-Point Costs for $n = 7$ and $m = 6$ Once $\boldsymbol{J}$, $\boldsymbol{W}$, and ∇f Are Known, Using the Efficiencies Mentioned in the Text

Method	mults/divides	adds/subtracts
(2)–(5), $\mathbf{W} = \mathbf{I}$	330	275
(2)–(5), any $\mathbf{W}$	631	499
(6), $\mathbf{W} = \mathbf{I}$	195	154
(6), any $\mathbf{W}$	244	196

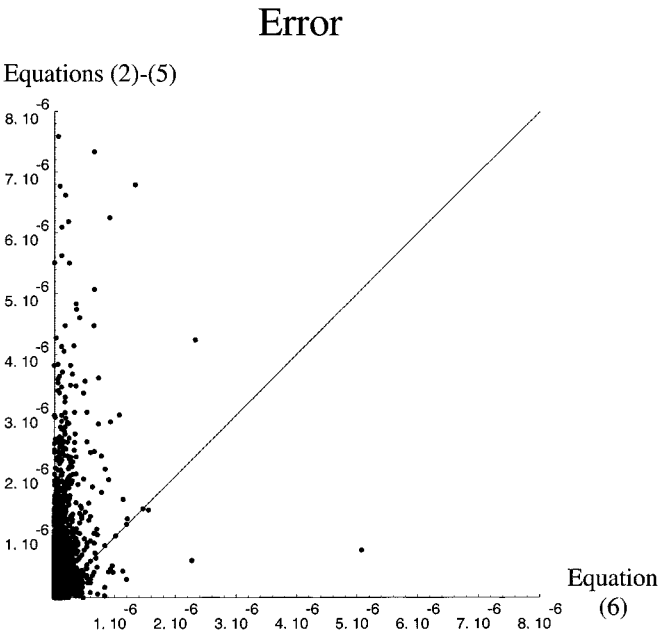


Fig. 1. Errors in end-effector velocity using 10 000 configurations and unit-norm velocity directions. The error for the methods of (2)–(5) and of (6) are shown against the vertical and horizontal scales, respectively. The diagonal line separates the regions where one method outperforms the other. Statistics for these errors are given in Table III. Points with errors greater than 8.0×10^{-6} for the method of (2)–(5) are not shown.

TABLE III
Error Statistics for the 10 000 Samples. The Percentage Error Measure is the Percentage of Points for Which One Method Performed Worse Than the Other—9% of the Time Both Methods Had the Same Error

Error	Eqns. (2)–(5)	Eqn. (6)
Average	2.7×10^{-7}	6.4×10^{-8}
Worst Case	3.9×10^{-5}	5.1×10^{-6}
Percentage	79	12

VI. When $\boldsymbol{J}$ is not of Full Rank

When the Jacobian is not of full rank, the augmented form of (6) can still be used, but with modification. In particular, when $\boldsymbol{J}$ does not have full rank, $\dot{\boldsymbol{q}}$ can be found as follows:

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{R}_J^T\boldsymbol{J}^T\boldsymbol{W}_{\hat{x}}\boldsymbol{J} \\ \boldsymbol{N}_J^T\boldsymbol{W} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{R}_J^T\boldsymbol{J}^T\boldsymbol{W}_{\hat{x}}\dot{\boldsymbol{x}} \\ -\alpha\boldsymbol{N}_J^T\nabla f \end{bmatrix} \tag{13}$$

where $\boldsymbol{W}_{\dot{x}}$ is a weighting matrix for the $\dot{\boldsymbol{x}}$-space and $\boldsymbol{R}_J$ is a matrix with columns that complete the columns of $\boldsymbol{N}_J$ to a spanning set of the entire space. Equation (13) minimizes $(1/2)\dot{\boldsymbol{q}}^T\boldsymbol{W}\dot{\boldsymbol{q}}+\alpha\dot{f}$ subject to $(\boldsymbol{J}\dot{\boldsymbol{q}}-\dot{\boldsymbol{x}})^T\boldsymbol{W}_{\dot{x}}(\boldsymbol{J}\dot{\boldsymbol{q}}-\dot{\boldsymbol{x}})$ being minimized.

Note (13) does not solve the problem of kinematic singularities. Equation (6) will typically not give a physically realizable joint-rate solution near a kinematic singularity, and the same is true of the method comprising (2)–(5). To address this issue, a method that does not give the desired $\dot{\boldsymbol{x}}$, such as time scaling or damped least squares, must be used.

VII. Summary

Many velocity-control techniques have the same structure and can be cast in this article's framework simply as different choices of scaling factor (α), configuration measure (f), and joint-rate weighting matrix ($\boldsymbol{W}$). As a conceptual tool, this provides a basis for common analysis. Any technique so cast can be verified against the meaning of minimizing $(1/2)\dot{\boldsymbol{q}}^T\boldsymbol{W}\dot{\boldsymbol{q}}+\alpha\dot{f}$ in the Jacobian's null space. As a method of implementation, the framework allows easy changes between parameters in software and is efficient and accurate. It also allows the possibility of avoiding explicit calculation of ∇f and $\boldsymbol{W}$ by calculating $(\boldsymbol{N}_J^T\nabla f)$ and $(\boldsymbol{N}_J^T\boldsymbol{W})$ directly.

References

[1] A. Liégeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 868–871, Dec. 1977.

[2] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *ASME J. Dynamic Syst., Meas., Contr.*, vol. 108, no. 3, pp. 163–171, Sept. 1986.

[3] C. W. Wampler II, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-16, pp. 93–101, Jan./Feb. 1986.

[4] P. Chiacchio and B. Siciliano, "A closed-loop Jacobian transpose scheme for solving the inverse kinematics of nonredundant and redundant wrists," *J. Robot. Syst.*, vol. 6, no. 5, pp. 601–630, 1989.

[5] A. Ben-Israel and T. N. E. Greville, *Generalized Inverses: Theory and Applications*. New York: Wiley Interscience, 1974.

[6] D. E. Whitney, "Resolved motion rate control of manipulators and human prostheses," *IEEE Trans. Man–Mach. Syst.*, vol. MMS-10, no. 2, pp. 47–53, June 1969.

[7] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Reading, MA: Addison-Wesley, 1991.

[8] K. L. Doty, C. Melchiorri, and C. Bonivento, "A theory of generalized inverses applied to robotics," *Int. J. Robot. Res.*, vol. 12, no. 1, pp. 1–19, Feb. 1993.

[9] C. A. Klein and A. I. Chirco, "Dynamic simulation of a kinematically redundant manipulator system," *J. Robot. Syst.*, vol. 4, no. 1, pp. 5–23, 1987.

[10] R. V. Mayorga and A. K. C. Wong, "A singularities avoidance approach forthe optimal local path generation of redundant manipulators," in *Proc. 1988 IEEE Int. Conf. Robotics and Automation*, Philadelphia, PA, Apr. 24–29, 1988, pp. 49–54.

[11] J. M. Hollerbach and K. C. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE J. Robot. Automat.*, vol. RA-3, pp. 308–316, Aug. 1987.

[12] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. Robot. Res.*, vol. 4, no. 3, pp. 109–117, Fall 1985.

[13] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 2, pp. 3–15, Summer 1987.

[14] C. L. Lewis and A. A. Maciejewski, "Fault tolerant operation of kinematically redundant manipulators for locked joint failures," *IEEE Trans. Robot. Automat.*, vol. 13, pp. 622–629, Aug. 1997.

[15] C. J. J. Paredis and P. K. Khosla, "Designing fault-tolerant manipulators: How many degrees of freedom?," *Int. J. Robot. Res.*, vol. 15, no. 6, pp. 611–628, Dec. 1996.

[16] C. A. Klein and C. H. Huang, "Review of pseudoinverse control for use with kinematically redundant manipulators," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, pp. 245–250, Mar./Apr. 1983.

[17] O. Egeland, "Task-space tracking with redundant manipulators," *IEEE J. Robot. Automat.*, vol. RA-3, pp. 471–475, Oct. 1987.

[18] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Robot. Automat.*, vol. 5, pp. 472–490, Aug. 1989.

[19] R. G. Roberts and A. A. Maciejewski, "Repeatable generalized inverse control strategies for kinematically redundant manipulators," *IEEE Trans. Automat. Contr.*, vol. 38, pp. 689–699, May 1993.

[20] J. Baillieul, "Kinematic programming alternatives for redundant manipulators," in *Proc. 1985 IEEE Int. Conf. Robot. Automat.*, St. Louis, MO, Mar. 25–28, 1985, pp. 722–728.

[21] H. Seraji and R. Colbaugh, "Improved configuration control for redundantrobots," *J. Robot. Syst.*, vol. 7, no. 6, pp. 897–928, 1990.

[22] J.-O. Kim, P. Khosla, and W.-K. Chung, "Static modeling and control of redundant manipulators," *Robot. Comput.-Integr. Manufact.*, vol. 9, no. 2, pp. 145–157, 1992.

[23] B. H. Hu, C. L. Teo, and H. P. Lee, "Local optimization of weighted joint torques for redundant robotic manipulators," *IEEE Trans. Robot. Automat.*, vol. 11, pp. 422–425, June 1995.

[24] K. L. Doty, C. Melchiorri, E. M. Schwartz, and C. Bonivento, "Robot manipulability," *IEEE J. Robot. Automat.*, vol. 11, pp. 462–468, June 1995.

[25] K. Sugimoto, "On the manipulability and singularity of manipulators," in *Robotics Research: The 5th Int. Symp.*, H. Miura and S. Arimoto, Eds, 1989, pp. 301–308.

[26] F. A. Mussa-Ivaldi and N. Hogan, "Integrable solutions of kinematic redundancy via impedance control," *Int. J. Robot. Res.*, vol. 10, no. 5, pp. 481–491, Oct. 1991.

[27] J. D. English, "Free-swinging failure tolerance for robotic manipulators," Ph.D. dissertation, Purdue Univ., West Lafayette, IN, 1996.

[28] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Englewood Cliffs, NJ: Prentice-Hall, 1974.

[29] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins Univ. Press, 1983.

[30] M. Z. Huang and H. Varma, "Optimal rate allocation in kinematically redundant manipulators—The dual projection method," in *Proc. 1991 IEEE Int. Conf. Robotics and Automation*, Sacramento, CA, Apr. 9–11, 1991, pp. 702–707.

[31] R. V. Dubey, J. A. Euler, and S. M. Babcock, "An efficient projection optimization scheme for a seven-degree-of-freedom redundant robot with spherical wrist," in *Proc. 1988 IEEE Int. Conf. Robotics and Automation*, Philadelphia, PA, Apr. 24–29, 1988, pp. 28–36.

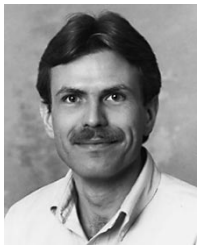
[32] C. Chevallereau and W. Khalil, "A new method for the solution of the inverse kinematics of redundant robots," in *Proc. 1988 IEEE Int. Conf. Robotics and Automation*, Philadelphia, PA, Apr. 24–29, 1988, pp. 37–42.

[33] H. Zghal, R. V. Dubey, and J. A. Euler, "An efficient gradient projection optimization for manipulators with multiple degrees of redundancy," in *Proc. IEEE 1990 Int. Conf. Robotics and Automation*, Cincinnati, OH, May 13–18, 1990, pp. 1006–1011.

[34] R. P. Podhorodeski, A. A. Goldenberg, and R. G. Fenton, "Resolving redundant manipulator joint rates and identifying special arm configurations using Jacobian null-space bases," *IEEE Trans. Robot. Automat.*, vol. 7, pp. 607–618, Oct. 1991.

James English received the B.S. degree from Virginia Tech, Blacksburg, in 1988, the M.E. degree from the University of Florida, Gainesville, FL, in 1991, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 1996, all in electrical engineering.

He has been with the Center for Mathematical System Theory, University of Florida, Gainsville; The Voice of America, Washington, DC and Tangier, Morocco; and the Raytheon Systems Company, Tucson, AZ. He is currently with MÄK Technologies, Cambridge, MA. His primary research interests lie in the simulation and design of remote automatic systems.



Anthony Maciejewski received the B.S.E.E., M.S., and Ph.D. degrees in electrical engineering from The Ohio State University, Columbus, in 1982, 1984, and 1987, respectively. From 1985 to 1986, he was an American Electronics Association Japan Research Fellow at the Hitachi Central Research Laboratory, Tokyo, Japan.

He is currently a Professor of electrical and computer engineering at Purdue University, West Lafayette, IN. His primary research interests center on the simulation and control of kinematically redundant and failure tolerant robotic systems.