Measuring and Reducing the Euclidean-Space Effects of Robotic Joint Failures

James D. English, *Member, IEEE,* and Anthony A. Maciejewski, *Member, IEEE*

***Abstract*—Robotic joint failures are directly characterized and measured in joint space. A locking failure, for example, is one for which a joint cannot move, and it gives an error equal to the desired value minus the locked value. This article extends the joint-space characterization to Euclidean space by measuring a failure's effect there. The approach is based on a rudimentary measure of point error that can be defined to be distance or path length. It is used to form comprehensive measures through weighted integration over Euclidean-space regions. For kinematically redundant manipulators, minimizing the measures using the redundancy is a method to induce failure tolerance. This can be applied both before a failure to reduce the likelihood of collision-induced damage and after a failure to reduce end-effector error. Examples for both cases are given.**

***Index Terms*—Fault/failure tolerance, free-swinging joint failure, kinematically redundant, kinematics, locked joint failure, manipulators, redundant robots/manipulators, robots.**

I. Introduction

SOME EXAMPLES of robotic joint-failure types are as follows: 1) locking, where the joint cannot move [1]–[3]; 2) free swinging, where actuator torque is lost [4]; and 3) calibration, where the joint value is perturbed by an unknown, possibly time-varying, value. These failures all eventually express themselves through joint position error. Errors may involve multiple joints (as, for example, when a hydraulic system loses pressure), but it is typical—and is assumed for this work—that a failure-induced error is isolated to one joint. This joint error is an imprecise measure of the effect of the failure, however, even for the same joint on the same manipulator. This is illustrated in Fig. 1, where identical joint errors produce significantly different errors as measured in Euclidean space.

This article addresses the effect of failure-induced joint position error in Euclidean space. It is motivated by applications in remote environments, such as in space exploration or hazardous waste remediation, that place a premium on safety. In these applications, it is typical to place limits on the maximum velocity of a manipulator's motion and yet still require that joint failures not result in collisions with the environment. This restriction precludes the use of dynamic failure recovery schemes that command high velocities for unfailed joints in order to compensate for the motion of failed joints. An alternative that does not require rapid response is to prepare for any possible uncommanded joint motion due to a particular failure by continuously configuring the manipulator in such a way as to minimize the likelihood of failure-induced collision. For example, the prefailure manipulator configuration on the right-hand side of Fig. 1 is preferable to that on the left-hand side, because should the actuator on joint one lose torque, the volume of the workspace through which the manipulator falls is much smaller. In addition, the Euclidean error experienced by the end-effector is significantly smaller. Thus, the ability to guarantee that geometric properties, such as the swept volume, resulting from a joint failure are minimized provides an additional safety mechanism.

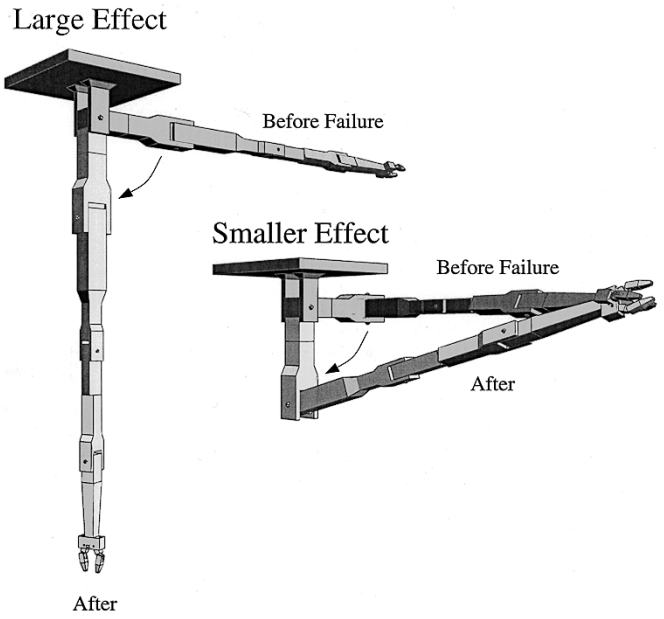


Fig. 1. A manipulator in two configurations before and after a joint-one error of $\pi/2$. The upper-left robot experiences extensive arm and end-effector displacement, while the lower-right robot experiences significantly less. This article presents methods to measure and reduce these kinematic effects that transcend joint error. (The error shown here would correspond to that caused by a loss of joint-one actuator torque.)

In addition to requiring mechanisms that reduce the immediate impact of failures, robots employed in remote environments are frequently required to continue operating, albeit in a reduced capacity, even after failures occur, i.e., such robots must "gracefully degrade" in performance. While there are techniques that actively use the dynamics of failed joints [5], this work focuses on how the commanded motion of unfailed joints

Manuscript received March 5, 1999; revised September 24, 1999 This paper was recommended for publication by Associate Editor I. Walker and Editor A. De Luca upon evaluation of the reviewers' comments. This work was supported by a NASA Graduate Student Research Fellowship under Grant NGT9-2 and by Sandia National Laboratories under Contract AL-3011. This paper was presented in part at the 1997 IEEE International Conference on Robotics and Automation.

J. D. English is with MÄK Technologies, Cambridge, MA 02138 USA.

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

Publisher Item Identifier S 1042-296X(00)03024-X.

can be used to minimize the effects of the motion of an unreliable joint. (The unreliable joint motion may be due to either an actuator failure or a sensor failure.) A kinematic analysis of the Euclidean error induced by an existing joint failure can be used to identify optimal configurations for minimizing this error, as well as for providing bounds on the accuracy to which a manipulator can complete its assigned task.

There is no one natural way to measure the kinematic aspects of rigid-body motions with a scalar [6]. However, physical objects do allow a focusing of motion to form scalar measures (an example of which is the "volume of a swept volume" [7], among others [8]), and this concept is used here. The idea is to first define a measure of the motion of a point after a joint failure—a simpler task—and then extend it to find the motion of an object by integrating a weighting of the point measure squared over the object. The object measure is further extended to a manipulator-wide measure by incorporating multiple objects in multiple coordinate frames.

The presented measures are, in general, functions of the joint variables, and a primary goal of this work is to enable reduction of the measures in kinematically redundant manipulators using self motion. The aim is to achieve a degree of failure tolerance by either best configuring a manipulator in anticipation of a failure or reconfiguring it for failure recovery, possibly in conjunction with other recovery methods [9]. Failure tolerance is especially important for manipulators used in hazardous or remote environments [10]–[14], and kinematically redundant manipulators have been proposed for use there [2], [15]–[21].

Methods for optimizing criterion functions using self motion include the augmented Jacobian technique [22], [23] for tracking a desired value, the extended Jacobian technique [24] for tracking critical points, and the gradient-projection method [25], [26] for tracking extrema. Because these methods all require knowledge of the function's gradient, methods for calculating the gradients of the error measures are given.

II. Point Error

The foundation of the Euclidean-space measures is established here by defining a rudimentary measure of the error of a point. This measure will be extended to objects in Sections III and IV.

Let an n-degree-of-freedom manipulator with joint variables $\boldsymbol{q}$ have a failure at joint i, with an error in the failed joint variable of $\tilde{q}_i$ (i.e., $\tilde{q}_i$ equals the actual value of q_i, entry i of $\boldsymbol{q}$, minus its desired value). For some failure modes, $\tilde{q}_i$ will be a function of $\boldsymbol{q}$. (Examples will be given in Section VI.) Let point r, whose Euclidean-space error is of interest, lie at the tip of vector $\vec{r}$ in Denavit–Hartenberg (D–H) frame k, $1 \leq k \leq n$. (This article uses the frame-labeling scheme of Paul [27].) The point so chosen is completely general—any location in any frame. Then, the point error e_i represents a measure of point r's motion caused by $\tilde{q}_i$. Two possible values, path length assuming stationary healthy joints and Euclidean distance, are shown in Fig. 2.

Let $\vec{r}^{\,*}_{i-1}$ be the perpendicular vector from the line passing through $\hat{z}_{i-1}$, the z-axis of D–H frame $i-1$, to the tip of $\vec{r}$,

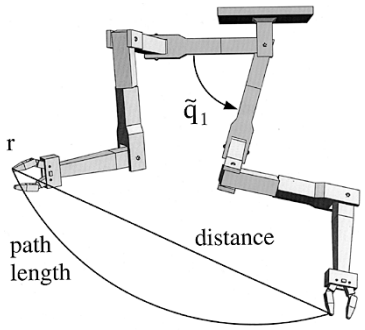


Fig. 2. A manipulator before and after a failure of the first joint. The joint error $\tilde{q}_1$ induces motion of the interest point r (a point on the hand in this case). Two possible measures of the point's motion, denoted e_1, are shown: path length assuming stationary healthy joints and Euclidean distance.

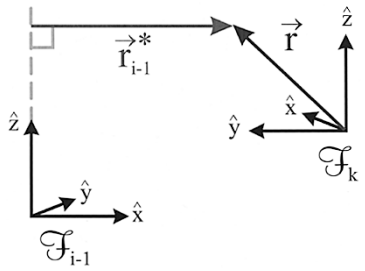


Fig. 3. For use in finding the distance error of the tip of $\vec{r}$ caused by an error in joint i, $\vec{r}^{\,*}_{i-1}$ is defined as the perpendicular vector from the line coinciding with the axis of joint i to the tip of $\vec{r}$.

as shown in Fig. 3. Then, in its general form, the point error is defined using a static, nonnegative function $\phi(\cdot)$ as follows:

$$e_i^2 = \begin{cases} \phi(\tilde{q}_i)\|\vec{r}^{\,*}_{i-1}\|^2, & i \leq k, \text{ joint } i \text{ rotational} \\ \tilde{q}_i^2, & i \leq k, \text{ joint } i \text{ prismatic} \\ 0, & i > k. \end{cases} \tag{1}$$

This general form allows e_i to be defined as either path length assuming stationary healthy joints, using

$$\phi(\tilde{q}_i) = \tilde{q}_i^2 \tag{2}$$

or Euclidean distance, using

$$\phi(\tilde{q}_i) = 2(1 - \cos(\tilde{q}_i)). \tag{3}$$

Path length is appropriate when the focus is on the process of the failure, and Euclidean distance is appropriate when the focus is on the result of the failure. Examples showing when each of these is applicable are given in Section VI. In the ensuing text, however, $\phi(\cdot)$ will be used in the general sense and not restricted to either of these values.

The scalar $\|\vec{r}^{\,*}_{i-1}\|$ can be found through

$$\|\vec{r}^{\,*}_{i-1}\| = \|(\vec{p}_{i-1\to k} + \vec{r}) \times \hat{z}_{i-1}\| \tag{4}$$

where $\vec{p}_{\ell\to k}$ is the vector from the origin of D–H frame ℓ to D–H frame k. This equation is in coordinate-free form and can be calculated in any frame.

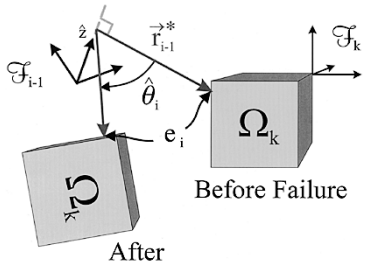


Fig. 4. To find the object error for object Ω_k rigidly attached to D–H frame k, the weighted point error squared is integrated over the object.

III. Object Error

Different points on the manipulator (typically) move different distances after a failure. This section expands the point-based measure to a region—or object—rigidly attached to one D–H frame. (This object may comprise several disjoint sets.) The object-based measure is found by integrating the square of the point error times a weighting function over the object.

Let Ω_k be the object rigidly attached to frame k, and let $\rho_k(\vec{r}, t)$ be a possibly time-varying weighting function for which there exists some (preferably small) integer N_k such that it can be decomposed as follows:

$$\rho_k(\vec{r}, t) = \sum_{i=1}^{N_k} \rho'_{k,i}(\vec{r})\rho''_{k,i}(t). \tag{5}$$

This decomposition will allow a reduction in computation by eliminating the need for online integration.

The object-based measure o_i is established by integrating the product of $\rho_k(\vec{r}, t)$ and $[c_i(\vec{r}, \tilde{q}_i, \boldsymbol{q})]^2$ over Ω_k

$$o_i(\Omega_k, \tilde{q}_i, \boldsymbol{q}, t) = \int_{\Omega_k} \rho_k(\vec{r}, t)[c_i(\vec{r}, \tilde{q}_i, \boldsymbol{q})]^2 \, d\omega_k \tag{6}$$

where $d\omega_k$ is a differential volume, area, or distance element when Ω_k is a solid, surface, or curve, respectively. This concept is illustrated in Fig. 4, where Ω_k is a solid. When c_i is Euclidean distance, (6) corresponds to the object norm of Kazerounian and Rastegar [28].

Equation (6) is in general a computationally expensive calculation that cannot be performed online. This calculation has been identified as a potential drawback of Kazerounian and Rastegar's method [8], [29]. However, it will be shown below that for the given assumptions, there exists a set of integrations independent of the joint variables that can be performed only once, and the results used in lieu of integrating $[\rho_k(\vec{r}, t)c_i^2]$ each time step. When N_k [as used in (5)] is sufficiently small, this allows real-time calculation.

A. Calculating o_i for Rotational Joint i

Using (1) for joint i rotational, (6) gives

$$o_i = \begin{cases} \phi_i \displaystyle\int_{\Omega_k} \rho_k \|\vec{r}^{*}_{i-1}\|^2 \, d\omega_k, & i \le k \\ 0, & i > k \end{cases} \tag{7}$$

where $\phi_i = \phi(\tilde{q}_i)$. Thus, the problem becomes one of finding $\int_{\Omega_k} \rho_k \|\vec{r}^{*}_{i-1}\|^2 \, d\omega_k$.

This integral can be efficiently evaluated by decomposing it using the configuration-independent components $m_{\Omega_k} = \int_{\Omega_k} \rho_k \, d\omega_k$, $\vec{h}_{\Omega_k} = \int_{\Omega_k} \rho_k \vec{r} \, d\omega_k$, and $\boldsymbol{I}_{\Omega_k} = \int_{\Omega_k} \rho_k \boldsymbol{R}^T \boldsymbol{R} \, d\omega_k$, where $\boldsymbol{R}$ is the cross-product matrix for $\vec{r}$; i.e., $\boldsymbol{R}\vec{v} = \vec{r} \times \vec{v}$ for all $\vec{v}$. These components can be evaluated by first performing offline calculation of the following time-independent quantities:

$$ {}_i m'_{\Omega_k} = \int_{\Omega_k} \rho'_{k,i}(\vec{r}) \, d\omega_k \tag{8}$$

$$ {}_i \vec{h}'_{\Omega_k} = \int_{\Omega_k} \rho'_{k,i}(\vec{r}) \vec{r} \, d\omega_k \tag{9}$$

$$ {}_i \boldsymbol{I}'_{\Omega_k} = \int_{\Omega_k} \rho'_{k,i}(\vec{r}) \boldsymbol{R}^T \boldsymbol{R} \, d\omega_k. \tag{10}$$

The definitions in (8)–(10) are analogous to the rigid-body inertial parameters of mass, first moment of inertia, and second moment of inertia, respectively. Using these, m_{Ω_k}, $\vec{h}_{\Omega_k}$, and $\boldsymbol{I}_{\Omega_k}$ can be calculated through the following online summations:

$$m_{\Omega_k}(t) = \sum_{i=1}^{N_k} \rho''_{k,i}(t) \, {}_i m'_{\Omega_k} \tag{11}$$

$$\vec{h}_{\Omega_k}(t) = \sum_{i=1}^{N_k} \rho''_{k,i}(t) \, {}_i \vec{h}'_{\Omega_k} \tag{12}$$

$$\boldsymbol{I}_{\Omega_k}(t) = \sum_{i=1}^{N_k} \rho''_{k,i}(t) \, {}_i \boldsymbol{I}'_{\Omega_k}. \tag{13}$$

These allow $\int_{\Omega_k} \rho_k \|\vec{r}^{*}_{i-1}\|^2 \, d\omega_k$ to be expressed in coordinate-free form as

$$\begin{aligned} &\int_{\Omega_k} \rho_k \|\vec{r}^{*}_{i-1}\|^2 \, d\omega_k \\ &\quad = m_{\Omega_k} \|\vec{p}_{i-1\to k} \times \hat{z}_{i-1}\|^2 + 2\hat{z}_{i-1} \times (\vec{p}_{i-1\to k} \times \hat{z}_{i-1}) \\ &\qquad \cdot \vec{h}_{\Omega_k} + \hat{z}_{i-1} \cdot \boldsymbol{I}_{\Omega_k} \hat{z}_{i-1} \end{aligned} \tag{14}$$

using the identity $\|\vec{r} \times \hat{z}_{i-1}\|^2 = \hat{z}_{i-1} \cdot \boldsymbol{R}^T \boldsymbol{R} \hat{z}_{i-1}$ with (4). This used in (7) allows calculation of (6) with no online integration. Equation (14) can be efficiently calculated in frame k.

B. Calculating o_i for Prismatic Joint i

Substituting (1) for joint i prismatic into (6) gives

$$o_i = \begin{cases} \tilde{q}_i^2 m_{\Omega_k}, & i \le k \\ 0, & i > k \end{cases} \tag{15}$$

with m_{Ω_k} calculated using (11).

C. Calculating the Gradient of o_i

For reducing o_i, its gradient is typically required. The gradient is formed from the partial derivatives with respect to the joint variables, and methods for calculating these are presented here.

1) Joint i Rotational: When joint i is rotational and $i > k$, from (7) $o_i = 0$ and therefore

$$\frac{\partial}{\partial q_j} o_i = 0 \tag{16}$$

and when $i \leq k$, taking the partial derivative of (7) gives

$$\frac{\partial}{\partial q_j} o_i = \phi_i \frac{\partial}{\partial q_j} \int_{\Omega_k} \rho_k \|\vec{r}^*_{i-1}\|^2 \, d\omega_k + \left[\frac{\partial}{\partial q_j} \phi_i\right] \int_{\Omega_k} \rho_k \|\vec{r}^*_{i-1}\|^2 \, d\omega_k. \tag{17}$$

In finding $(\partial/\partial q_j) \int_{\Omega_k} \rho_k(\vec{r}) \|\vec{r}^*_{i-1}\|^2 \, d\omega_k$ for $i \geq j$ or $k < j$, $\|\vec{r}^*_{i-1}\|$ does not change as a function of q_j and thus

$$\frac{\partial}{\partial q_j} \int_{\Omega_k} \rho_k(\vec{r}) \|\vec{r}^*_{i-1}\|^2 \, d\omega_k = 0. \tag{18}$$

This leaves only the case $i < j \leq k$. When joint j is rotational, applying the product rule to (14) and using the fact that outboard, fixed-length vectors move according to the rule $(\partial/\partial q_j)\vec{v} = \hat{z}_{j-1} \times \vec{v}$, gives

$$\begin{aligned} &\frac{\partial}{\partial q_j} \int_{\Omega_k} \rho_k(\vec{r}) \|\vec{r}^*_{i-1}\|^2 \, d\omega_k \\ &\quad = 2(((m_{\Omega_k}\vec{p}_{i-1\to k} + \vec{h}_{\Omega_k}) \times \hat{z}_{i-1}) \cdot ((\hat{z}_{j-1} \times \vec{p}_{j-1\to k}) \times \hat{z}_{i-1}) \\ &\qquad + \hat{z}_{i-1} \times (\vec{p}_{i-1\to k} \times \hat{z}_{i-1}) \cdot \hat{z}_{j-1} \times \vec{h}_{\Omega_k} \\ &\qquad - \hat{z}_{j-1} \times \hat{z}_{i-1} \cdot \boldsymbol{I}_{\Omega_k} \hat{z}_{i-1}). \end{aligned} \tag{19}$$

When joint j is prismatic, only $\vec{p}_{i-1\to k}$ changes with q_j, according to the rule $(\partial/\partial q_j)\vec{p}_{i-1\to k} = \hat{z}_{j-1}$, giving

$$\begin{aligned} &\frac{\partial}{\partial q_j} \int_{\Omega_k} \rho_k(\vec{r}) \|\vec{r}^*_{i-1}\|^2 \, d\omega_k \\ &\quad = 2(\hat{z}_{j-1} \times \hat{z}_{i-1}) \cdot \left(\left(m_{\Omega_k}\vec{p}_{i-1\to k} + \vec{h}_{\Omega_k}\right) \times \hat{z}_{i-1}\right). \end{aligned} \tag{20}$$

The computation of $(\partial/\partial q_j)\phi_i$ as required in (17) is straightforward using the appropriate value of $(\partial/\partial q_j)\tilde{q}_i$ with (2) or (3).

2) Joint i Prismatic: When joint i is prismatic, from (15)

$$\frac{\partial}{\partial q_j} o_i = \begin{cases} 2m_{\Omega_k} \tilde{q}_i \dfrac{\partial}{\partial q_j} \tilde{q}_i, & i \leq k \\ 0, & i > k. \end{cases} \tag{21}$$

IV. Multi-Object Error

Section III presented a measure of joint error that assigned a scalar to the movement of an object rigidly attached to one frame. This was, however, restrictive. For example, if secondary damage caused by a moving manipulator after a failure is of concern, then the entire arm should be taken into account. This section presents a measure for this purpose and gives an efficient calculation method. The measure is established through a weighted integration of the point error squared over multiple objects in multiple frames.

In particular, for all k, $1 \leq k \leq n$, let Ω_k be the object rigidly attached to D–H frame k, with $\rho_k(\vec{r}, t)$ the weighting function for Ω_k. Then integrating the weighting of e_i^2 over all the regions is equivalent to summing n single-object-based measures. Using $o_i = 0$ for $i > k$, the multi-object measure $\hat{o}_i$ becomes

$$\hat{o}_i = \sum_{k=i}^{n} o_i(\Omega_k). \tag{22}$$

With this formulation, $\hat{o}_i$ could be found by repeated application of (7) and (15). However, this can be an inefficient approach if $\hat{o}_i$ is calculated for multiple values of i, as would be the case if a failure were anticipated for multiple joints. A procedure is given below that allows calculation of $\hat{o}_i$ for all i, $1 \leq i \leq n$, in order $O(n)$ time once the $\tilde{q}_i$ are known.

A. Calculating $\hat{o}_i$ for Rotational Joint i

Using (7) in (22) and factoring out ϕ_i gives

$$\hat{o}_i = \phi_i \sum_{k=i}^{n} \int_{\Omega_k} \rho_k \|\vec{r}^*_{i-1}\|^2 \, d\omega_k. \tag{23}$$

Now, if $\boldsymbol{I}^{\ell}_{\Omega_k}$ is defined as

$$\boldsymbol{I}^{\ell}_{\Omega_k} = m_{\Omega_k} \boldsymbol{P}^T_{\ell\to k} \boldsymbol{P}_{\ell\to k} + \boldsymbol{P}^T_{\ell\to k} \boldsymbol{H}_{\Omega_k} + \boldsymbol{H}^T_{\Omega_k} \boldsymbol{P}_{\ell\to k} + \boldsymbol{I}_{\Omega_k} \tag{24}$$

where $\boldsymbol{P}_{\ell\to k}$ is the cross-product matrix for $\vec{p}_{\ell\to k}$ and $\boldsymbol{H}_{\Omega_k}$ is the cross-product matrix for $\vec{h}_{\Omega_k}$, and $\boldsymbol{I}^*_{\ell}$ is defined as

$$\boldsymbol{I}^*_{\ell} = \sum_{k=\ell+1}^{n} \boldsymbol{I}^{\ell}_{\Omega_k} \tag{25}$$

then, using (14), (23) becomes

$$\hat{o}_i = \phi_i \hat{z}_{i-1} \cdot \boldsymbol{I}^*_{i-1} \hat{z}_{i-1}. \tag{26}$$

The matrix $\boldsymbol{I}^*_{\ell}$ as defined through (24) and (25) is analogous to the second moment of composite rigid-body inertia [30]; it can be calculated through the following procedure. Let the net outboard interest parameter be $\mho_{\ell}$, calculated using

$$\mho_{\ell} = m_{\Omega_{\ell}} + \mho_{\ell+1} \qquad \mho_n = m_{\Omega_n}. \tag{27}$$

And let $\vec{h}^*_{\ell}$ be the outboard interest vector, calculated using

$$\begin{aligned} \vec{h}^*_{\ell} &= {}^{\ell}\boldsymbol{R}_{\ell+1} \left(\vec{h}^*_{\ell+1} + \vec{h}_{\Omega_{\ell+1}} + \mho_{\ell+1} \vec{p}_{\ell+1}\right) \\ \vec{h}^*_n &= \vec{0} \end{aligned} \tag{28}$$

where for calculation, $\vec{h}_{\Omega_{\ell+1}}$ and $\vec{p}_{\ell+1}$ are both expressed in D–H frame $\ell + 1$. Then, $\boldsymbol{I}^*_{\ell}$ can be found through the following recursion:

$$\begin{aligned} \boldsymbol{I}^*_{\ell} = {}^{\ell}\boldsymbol{R}_{\ell+1} \, &(\boldsymbol{I}^*_{\ell+1} + \boldsymbol{I}_{\Omega_{\ell+1}} - \boldsymbol{P}_{\ell\to\ell+1}(\boldsymbol{H}^*_{\ell+1} + \boldsymbol{H}_{\Omega_{\ell+1}}) \\ &- (\boldsymbol{H}^*_{\ell+1} + \boldsymbol{H}_{\Omega_{\ell+1}}) \boldsymbol{P}_{\ell\to\ell+1} - \mho_{\ell+1} \boldsymbol{P}_{\ell\to\ell+1} \boldsymbol{P}_{\ell\to\ell+1}) {}^{\ell}\boldsymbol{R}^T_{\ell+1} \end{aligned} \tag{29}$$

$$\boldsymbol{I}^*_n = \boldsymbol{0}. \tag{30}$$

Each of (27), (28), and (29) with (30) can be calculated in $O(n)$ time.

B. Calculating $\hat{o}_i$ for Prismatic Joint i

When joint i is prismatic, $\hat{o}_i$ is given by

$$\hat{o}_i = \tilde{q}_i^2 \mho_i \tag{31}$$

with $\mho_i$ calculated using (27).

C. Calculating the Gradient of $\hat{o}_i$

The technique used to find the value of $\hat{o}_i$ in $O(n)$ time will be used here to establish a method for finding the gradients. It will enable calculation of $\nabla \hat{o}_i$ for all i in $O(n^2)$ time once the values for $\tilde{q}_i$ and $\nabla \tilde{q}_i$ are known.

1) Joint i Rotational: Straightforward application of the chain rule to (26) is simplified if partial derivatives are evaluated in frame i. In this case, $(\partial/\partial q_j)\hat{z}_{i-1} = 0$ so that

$$\frac{\partial}{\partial q_j}\hat{o}_i = \phi_i \hat{z}_{i-1} \cdot \left[\frac{\partial}{\partial q_j} \boldsymbol{I}^*_{i-1}\right] \hat{z}_{i-1} + \left[\frac{\partial}{\partial q_j}\phi_i\right] \hat{z}_{i-1} \cdot \boldsymbol{I}^*_{i-1} \hat{z}_{i-1}. \tag{32}$$

When $j \leq i$, $\boldsymbol{I}^*_{i-1}$ does not change in frame i as q_j changes, and (32) gives

$$\frac{\partial}{\partial q_j}\hat{o}_i = \left[\frac{\partial}{\partial q_j}\phi_i\right] \hat{z}_{i-1} \cdot \boldsymbol{I}^*_{i-1} \hat{z}_{i-1}. \tag{33}$$

When $j > i$ and joint j is rotational, $(\partial/\partial q_j)\boldsymbol{I}^*_{i-1}$ can be found using the analogy with composite rigid-body inertia (a formula for partial derivatives is given in [4])

$$\begin{aligned}\frac{\partial}{\partial q_j}\hat{o}_i =& 2\phi_i(\hat{z}_{i-1} \times \hat{z}_{j-1} \cdot \boldsymbol{I}^*_{j-1}\hat{z}_{i-1} \\ &+ (\hat{z}_{i-1} \times \vec{p}_{i-1\to j-1}) \cdot (\vec{h}^*_{j-1} \times (\hat{z}_{j-1} \times \hat{z}_{i-1}) \\ &- \hat{z}_{j-1} \times (\vec{h}^*_{j-1} \times \hat{z}_{i-1}))) \\ &+ \left[\frac{\partial}{\partial q_j}\phi_i\right] \hat{z}_{i-1} \cdot \boldsymbol{I}^*_{i-1}\hat{z}_{i-1}.\end{aligned} \tag{34}$$

Similarly, when $j > i$ and joint j is prismatic

$$\frac{\partial}{\partial q_j}\hat{o}_i = 2\mho_j \phi_i (\vec{p}_{i-1\to j-1} \times \hat{z}_{i-1}) \cdot (\hat{z}_{j-1} \times \hat{z}_{i-1}) + \left[\frac{\partial}{\partial q_j}\phi_i\right] \hat{z}_{i-1} \cdot \boldsymbol{I}^*_{i-1}\hat{z}_{i-1}. \tag{35}$$

2) Joint i Prismatic: Independent of joint j type, when joint i is prismatic, from (31)

$$\frac{\partial}{\partial q_j}\hat{o}_i = 2\mho_i \tilde{q}_i \frac{\partial}{\partial q_j}\tilde{q}_i. \tag{36}$$

V. Three-Link Planar Example

This first example uses a three-link planar arm with link lengths unity, link masses unity, and the center of mass of each link at the link center. The arm operates in the vertical plane with the task of end-effector positioning only. For this task, it has one degree of redundancy. The regions of interest (Ω_1, Ω_2, and Ω_3) are line segments running the length of each link. The interest density is a constant of $1m^{-1}$, where the unit of length is m, and the choice of point error is Euclidean distance with $\phi(\cdot)$ given by (3).

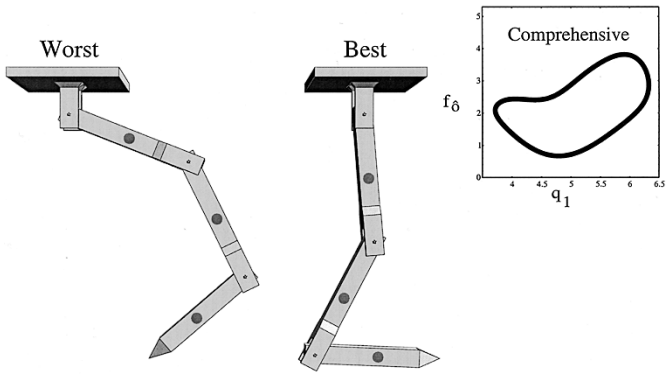


Fig. 5. The multi-object-error-based worst-case and best-case configurations for a stationary three-link planar revolute manipulator with anticipation of a failure in any joint. The manipulator on the left has the largest cost-function value using the comprehensive measure for end-effector position (0.61, −1.92), and the manipulator on the right has the smallest. The curve to the far right shows the cost-function value parameterized by q_1.

TABLE I
D–H Parameters for the RRC K-1207i

link	a (m)	d (m)	α (rad)	θ (rad)
1	-0.1016	0.0000	-1.5708	q_1
2	0.1016	0.0000	1.5708	q_2
3	-0.0857	0.5461	-1.5708	q_3
4	0.0857	0.0000	1.5708	q_4
5	-0.0591	0.5461	-1.5708	q_5
6	0.0591	0.0000	1.5708	q_6
7	0.0000	0.1778	0.0000	q_7

For this example, a possible free-swinging failure in any of the three joints is anticipated. An appropriate cost function is $f_{\hat{o}} = \sum_{i=1}^{3} \hat{o}_i$. For this cost function, with an end point (0.61, −1.92), worst-case and best-case configurations are shown in Fig. 5, together with a curve showing the value of the cost function parameterized by q_1. The gradient-projection technique [25], [26] was used to find both the worst-case (using the negative of the gradient) and best-case configurations.

Fig. 5 illustrates the tradeoff that must be made when optimizing a cost function that includes the possibility of multiple joint failures. The best-case configuration would actually give greater motion after a joint-three failure than the worst-case configuration. However, this is more than compensated by the fact that the best-case configuration gives much less motion after a joint-one failure. From the worst-case to the best-case configuration, the value of $f_{\hat{o}}$ is improved by more than a factor of five. The plot of $f_{\hat{o}}$ as a function of q_1 shows the existence of multiple local minima, as is commonly the case for functions of configuration. Methods for working with multiple minima are given in [31].

VI. Seven-Link Examples

In this section, the Robotics Research Corporation K-1207i manipulator is used as an example arm to illustrate a spatial redundant arm's ability to minimize the Euclidean effects of a joint failure. Its D–H parameters are given in Table I, and the

TABLE II
JOINT LIMITS IN RADIANS FOR THE RRC K-1207i

joint	upper limit	lower limit
1	3.1410	-3.1410
2	-0.0543	-3.0510
3	0.0000	-6.2800
4	0.0000	-3.0510
5	6.2800	-6.2800
6	0.6100	-2.9670
7	6.2800	-6.2800

software joint limits are given in Table II. This seven-degree-of-freedom arm has one degree of redundancy for the task of hand positioning and orienting. It is this extra degree of freedom that allows reduction of the error measures.

A. A Single-Object Calibration-Error Example

Here, an object rigidly attached to the K-1207i's end-effector is used as Ω_n to measure the effect of a joint calibration error. This could represent a number of error types for which the value of the error is not a known function of configuration. To focus on a calibration error, the joint error $\tilde{q}_i$ is fixed at a nonzero value that represents a bound on the possibly time-varying error. The object of interest is chosen, as an example, to be L-shaped, formed by joining four cubes of edge length 0.06 m. The top of the L-shaped object lies 0.10 m from the end frame along the z-axis.

For this example, errors for all points of the object are considered equally important so ρ_n is assigned to a constant, $\rho_n = 1\ \mathrm{m}^{-3}$. Additionally, Euclidean distance is chosen as a rudimentary measure to focus on the final effect of the error [i.e., $\phi(\cdot)$ is given by (3)]. The end frame is constrained to a pose given by the following homogeneous transformation matrix relative to the base frame:

$$\boldsymbol{H} = \begin{bmatrix} 0 & 0 & -1 & -0.25 \\ 0 & 1 & 0 & -0.60 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{37}$$

Under these conditions, the gradient-projection technique [25], [26] was used to find the worst-case and best-case configurations for tolerating a joint-five failure. These are shown in Fig. 6.

The potential error in the L-shaped object caused by an error in joint five is greatly reduced by reconfiguring. In fact, o_5 for the best-case configuration of Fig. 6 is two orders of magnitude less than that of the worst case. Fig. 7 shows the errors in the object caused by a 0.1-rad error in joints two through five using best and worst-case configurations. The improvement, using the best configuration, for the joint two and joint five failures is especially substantial, but improvement is evident in all cases. If a calibration-type failure was experienced in one of these joints, reconfiguring might allow a task to be completed that would otherwise be impossible. (It is assumed here that direct compensation of the error is not possible, as would be the case if the error value were not well known. If some compensation were possible, it could further improve the results.)

The cases for joints one and seven are not shown in Fig. 7 because, for static $\tilde{q}_i$, o_1 and o_n do not change once the hand is fixed. This will be the case for any manipulator where the focus is on the error induced on a single object in the last frame. The case for joint six is not shown because o_6 here changes very little with reconfiguring.

This technique of assuming a fixed bound on $\tilde{q}_i$ is also useful for addressing general failure modes in manipulators with software or hardware error checking. Error checking is commonly used in robotic controllers to stop the arm when a joint deviates excessively from its expected value, and the techniques presented in this article allow enhancement of this safety feature. If the excessive-error cutoff value for the K-1207i were set to 0.1 rad and the manipulator maintained an optimal configuration, it is clear from Fig. 7 that the L-shaped object would be less exposed to collision-induced damage.

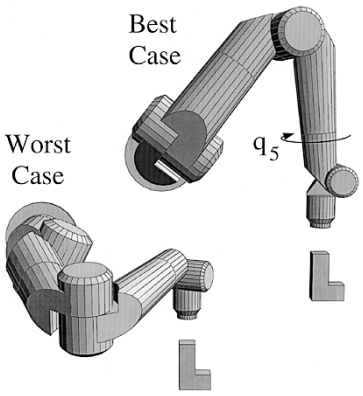


Fig. 6. Worst-case and best-case configurations of the RRC K-1207i for reducing the effect of joint-five error on the L-shaped object under the constraint of end-effector position/orientation given by (37). In the worst case, the line passing through the axis of the fifth joint lies far from all points on the object, and in the best case, the line passing through the axis of the fifth joint actually passes through the object. The motion of the object for these two configurations after a 0.1 rad error is shown at the bottom of Fig. 7.

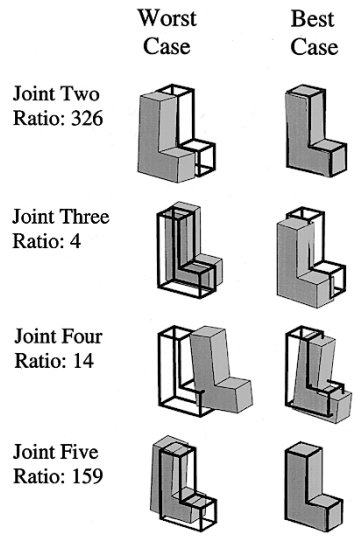


Fig. 7. Under the constraint of hand pose given by (37), the K-1207i was placed in worst-case and best-case configurations for each of joints two through five. (The configurations for joint five are shown in Fig. 6.) Then an error of 0.1 rad was imposed on the focus joint. The resulting error in the L-shaped object is shown here. In each case, the black frame represents where the object would be if there were no error. Given to the left of each set is the ratio of the worst-case to best-case values of o_i. Some joint-failure effects are more amenable to reconfiguration than others—for joint two the error is reduced by a factor of 326, while for joint three the error is reduced by only a factor of four.

TABLE III
MASSES AND CENTERS OF MASS FOR THE RRC K-1207i

link	mass (kg)	c_x (m)	c_y (m)	c_z (m)
1	19.051	0	0	-0.0030
2	9.299	0	0	0.3239
3	11.113	0	0	0.0064
4	5.897	0	0	0.3200
5	4.536	0	0	0.0127
6	2.381	0	0	0.1219
7	0.325	0	0	-0.0200

B. Multi-Object Free-Swinging Failure Example

In the previous example, the joint error was assumed to be bounded by a fixed, typically small, value. This is not always the case, however, and for this section's example, the joint failure is of the free-swinging type, i.e., one where actuator torque is lost. After a free-swinging failure, the arm moves under the influence of gravity. If the failed joint does not hit a stop, it settles into a configuration with the center of mass of the outboard links at its lowest point relative to the gravitational field [4], and this motion is a function of configuration.

For this example, the point error is path length assuming stationary healthy joints, with $\phi(\cdot)$ given by (2). This choice is appropriate for reducing the likelihood of collision-induced damage after a failure, and the stationary-healthy-joint assumption is an approximation for a slow-moving manipulator. (Manipulators used in remote and hazardous environments are typically slow moving.) Since collision with any part of the arm is of concern, the objects Ω_k are chosen to be the CAD models of the K-1207i links used to make the images of the robot for this article. All points on the arm are equally important, so the weighting function is chosen to be a constant, $\rho = 1\ \mathrm{m}^{-3}$. The values for the free-swinging joint error and its partial derivatives are calculated using the techniques given in [4].

For the hand pose given by (37) and link masses and centers of mass given in Table III, the worst-case and best-case configurations for tolerating a free-swinging failure of the first joint as found using the gradient-projection technique [25], [26] are shown in Fig. 8, both before and after a failure. With $\hat{o}_1$ equal to 0.384, the motion for the worst-case configuration is kinematically equivalent to translating a 1-m cube by 62 cm. In contrast, with $\hat{o}_1$ equal to 0.038, the motion for the best-case configuration is kinematically equivalent to translating a 1-m cube by 19 cm. It is clear from Fig. 8 that this reduced motion corresponds to a reduced likelihood of collision with the environment.

Even for this case, which is made more difficult by its focus on free-swinging failures, optimizing the configuration does not require excessive computation time. For the RRC K-1207i with a focus on free-swinging failures, $\hat{o}_i$ and $\nabla\hat{o}_i$ can be calculated for all i in less than 4 ms on a Sun Microsystems SPARC 10 workstation. This allows even the most general measures, including those incorporating all possible joint failures, to be minimized in real time.

VII. A 12-LINK EXAMPLE

This section presents an example of a highly redundant manipulator to illustrate the occurrence of algorithmic redundancy

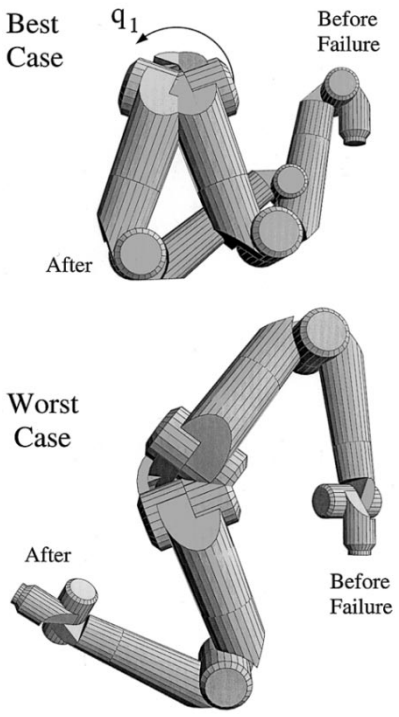


Fig. 8. Under the constraint of hand pose given by (37), the K-1207i was placed in worst-case and best-case configurations for a free-swinging failure of the first joint. Configurations before and after a failure are shown for both cases. The value of $\hat{o}_1$ for the best case is 0.038 and for the worst case is 0.384.

with respect to fault tolerance. Algorithmic redundancy occurs when the manipulator retains degrees of freedom in self motion while achieving a secondary criterion. For the secondary criterion of optimizing the fault-tolerance measures presented in this article, explicit conditions for algorithmic redundancy can be established in certain cases.

In particular, for rotational joint i, when the joint error $\tilde{q}_i$ is independent of configuration and the integrated scalar parameter $m_{\Omega_k} > 0$, the global minimum of the object error o_i has a direct geometrical interpretation. Under these conditions, using (7) and (14), o_i can be expressed as

$$o_i = \phi_i m_{\Omega_k} \left\| \left(\vec{p}_{i-1\to k} + \frac{1}{m_{\Omega_k}} \vec{h}_{\Omega_k} \right) \times \hat{z}_{i-1} \right\|^2 + \phi_i \left(\hat{z}_{i-1} \cdot \boldsymbol{I}_{\Omega_k} \hat{z}_{i-1} - m_{\Omega_k} \left\| \frac{1}{m_{\Omega_k}} \vec{h}_{\Omega_k} \times \hat{z}_{i-1} \right\|^2 \right) \quad (38)$$

giving that o_i is at its global minimum when $\hat{z}_{i-1}$ lies on the line $(1/m_{\Omega_k})\vec{h}_{\Omega_k} + \alpha\vec{v}_3$ as parameterized by α, where $\vec{v}_3$ is an eigenvector associated with the smallest eigenvalue of $\boldsymbol{I}_{\Omega_k} - (1/m_{\Omega_k})\boldsymbol{H}_{\Omega_k}^T \boldsymbol{H}_{\Omega_k}$.

Because there are four degrees of freedom in specifying a line, this implies minimizing a single-object-based measure subject to a positioning/orienting hand constraint will not, in general, resolve redundancy for manipulators with more than ten degrees of freedom, thus allowing the simultaneous optimization of additional criteria. To illustrate this, let a 12-link example manipulator have the geometry given in Table IV.

TABLE IV
D–H PARAMETERS FOR THE 12-LINK MANIPULATOR

link	a (m)	d (m)	α (rad)	θ (rad)
Odd, 1–11	0	0	$\frac{\pi}{2}$	q_i
Even, 2–10	0	$\frac{1}{2}$	$\frac{\pi}{2}$	q_i
12	0	$\frac{1}{4}$	0	q_{12}

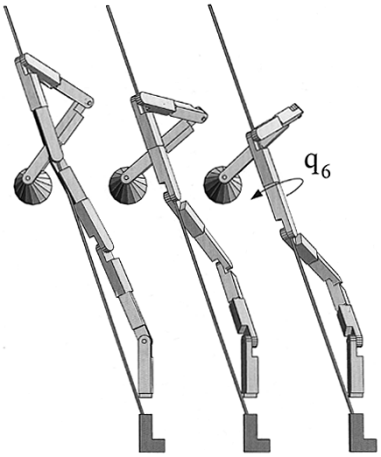


Fig. 9. Under the constraint of hand pose given by (39), the 12-link example arm has two degrees of freedom in minimizing o_6, the error in the L-shaped object caused by a fixed joint-six error. Shown here is the arm in three configurations, each with o_6 globally minimized. The set of all solutions globally minimizing o_6 can be characterized by those for which the joint-six axis lies on the line shown.

For the same L-shaped object as was used in Section VI and the end frame constrained to

$$\boldsymbol{H} = \begin{bmatrix} 0 & 0 & -1 & -1.10 \\ 0 & 1 & 0 & -0.60 \\ 1 & 0 & 0 & 0.40 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{39}$$

this manipulator can globally minimize o_6 yet retain two degrees of freedom. In Fig. 9, the arm is shown in three configurations as joint six moves along the line corresponding to a global minimum. This motion is achieved using joint rates lying in the null space of $[\boldsymbol{J}^T|\boldsymbol{H}_6]^T$, where $\boldsymbol{H}_6$ is the Hessian of o_6. In practice, algorithmic redundancy can be used to achieve other desirable criteria, such as the optimization of independent functions of configuration or joint rates [32].

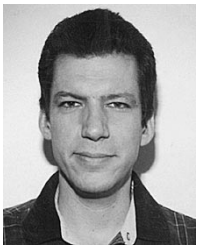
VIII. SUMMARY

This article defined measures of joint failures using Euclidean-space objects rigidly attached to a manipulator's links. The objects were used to expand point-error-based measures by integrating a weighting of the measures squared over the objects. Efficient ways of calculating the measures were given, and for the purpose of instilling fault tolerance in redundant manipulators, ways to calculate the measures' gradients were also presented. Examples showed how the measures could be used to prepare for a failure by reducing the likelihood of a collision or compensate for a failure by reducing the task error. Many of the reductions in collision likelihood and task error were substantial.

REFERENCES

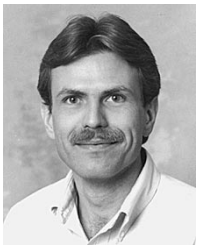
[1] A. A. Maciejewski, "Fault tolerant properties of kinematically redundant manipulators," in *Proc. 1990 IEEE Int. Conf. Robot. Automat.*, Cincinnati, OH, May 13–18, 1990, pp. 638–642.
[2] C. J. J. Paredis, W. K. F. Au, and P. K. Khosla, "Kinematic design of fault tolerant manipulators," *Comput. Elect. Eng.*, vol. 20, no. 3, pp. 211–220, May 1994.
[3] R. G. Roberts and A. A. Maciejewski, "A local measure of fault tolerance for kinematically redundant manipulators," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 543–552, Aug. 1996.
[4] J. D. English and A. A. Maciejewski, "Fault tolerance for kinematically redundant manipulators: Anticipating free-swinging joint failures," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 566–575, Aug. 1998.
[5] J. D. English and A. A. Maciejewski, "An example of failure tolerance through active braking," in *Proc. 2nd Int. Conf. Recent Adv. Mechatron. (ICRAM)*, Istanbul, Turkey, May 24–26, 1999, pp. 181–186.
[6] F. C. Park and R. W. Brockett, "Kinematic dexterity of robotic mechanisms," *Int. J. Robot. Res.*, vol. 13, no. 1, pp. 1–15, Feb. 1994.
[7] D. Blackmore and M. Leu, "Analysis of swept volume via Lie groups and differential equations," *Int. J. Robot. Res.*, vol. 11, no. 6, pp. 516–537, December 1992.
[8] J. M. R. Martinez and J. Duffy, "On the metrics of rigid body displacements for infinite and finite bodies," *ASME J. Mech. Design*, vol. 117, pp. 41–47, Mar. 1995.
[9] M. L. Visinsky, J. R. Cavallaro, and I. D. Walker, "Robotic fault detection and fault tolerance: A survey," *Rel. Eng. Syst. Safety*, vol. 46, pp. 139–158, 1994.
[10] R. Colbaugh and M. Jamshidi, "Robot manipulator control for hazardous waste-handling applications," *J. Robot. Syst.*, vol. 9, no. 2, pp. 215–250, 1992.
[11] E. Wu, J. Hwang, and J. Chladek, "Fault-tolerant joint development for the space shuttle remote manipulator system: Analysis and experiment," *IEEE Trans. Robot. Automat.*, vol. 9, pp. 675–684, Oct. 1993.
[12] M. L. Visinsky, J. R. Cavallaro, and I. D. Walker, "A dynamic fault tolerance framework for remote robots," *IEEE Trans. Robot. Automat.*, vol. 11, pp. 477–490, Aug. 1995.
[13] I. D. Walker and J. R. Cavallaro, "Failure mode analysis for a hazardous waste clean-up manipulator," *Rel. Eng. Syst. Safety*, vol. 53, no. 3, pp. 277–290, Sept. 1996.
[14] M. L. Leuschen, I. D. Walker, and J. R. Cavallaro, "Investigation of reliability of hydraulic robots for hazardous environments using analytic redundancy," in *Proc. 1999 Annu. Rel. Maintainability Symp.*, Washington, DC, 1999, pp. 22–128.
[15] Y. Ting, S. Tosunoglu, and D. Tesar, "A control structure for fault-tolerant operation of robotic manipulators," in *Proc. 1993 IEEE Int. Conf. Robot. Automat.*, Atlanta, GA, May 2–6, 1993, pp. 684–690.
[16] C. L. Lewis and A. A. Maciejewski, "Dexterity optimization of kinematically redundant manipulators in the presence of failures," *Comput. Elect. Eng.*, vol. 20, no. 3, pp. 273–288, May 1994.
[17] Y. Ting, S. Tosunoglu, and R. Freeman, "Torque redistribution and time regulation methods for actuator saturation avoidance of fault-tolerance parallel robots," *J. Robot. Syst.*, vol. 12, no. 12, pp. 807–820, Dec. 1995.
[18] R. Hooper, D. Sreevijayan, D. Tesar, J. Geisinger, and C. Kapoor, "Implementations of a four-level mechanical architecture for fault tolerant robots," *Rel. Eng. Syst. Safety*, vol. 53, no. 3, pp. 237–246, Sept. 1996.
[19] C. J. J. Paredis and P. K. Khosla, "Fault tolerant task execution through global trajectory planning," *Rel. Eng. Syst. Safety*, vol. 53, no. 3, pp. 225–235, Sept. 1996.
[20] 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.
[21] S. Tosunoglu and V. Monteverde, "Kinematic and structural design issues in the development of fault-tolerant manipulators," *Intell. Automat. Soft Computing*, vol. 4, no. 3, pp. 261–268, 1998.
[22] O. Egeland, "Task-space tracking with redundant manipulators," *IEEE J. Robot. Automat.*, vol. RA-3, pp. 471–475, Oct. 1987.
[23] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Robot. Automat.*, vol. 5, pp. 472–490, Aug. 1989.
[24] 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.
[25] 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.

[26] 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.
[27] R. P. Paul, *Robot Manipulators: Mathematics, Programming, Control*. Cambridge, MA: MIT Press, 1981.
[28] K. Kazerounian and J. Rastegar, "Object norms: A class of coordinate and metric independent norms for displacements," in *Proc. Flex. Mech., Dyn., Anal.*, Scottsdale, AZ, Sept. 13–16, 1992, pp. 271–275.
[29] F. C. Park, "Distance metrics on the rigid-body motions with applications to mechanical design," *ASME J. Mech. Des.*, vol. 117, pp. 48–54, Mar. 1995.
[30] M. W. Walker and D. E. Orin, "Efficient dynamic computer simulation of robotic mechanisms," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 104, pp. 205–211, Sept. 1982.
[31] J. D. English, "Free-swinging failure tolerance for robotic manipulators," Ph.D. dissertation, Purdue University, West Lafayette, IN, 1996.
[32] D. N. Nenchev, "Redundancy resolution through local optimization: A review," *J. Robot. Syst.*, vol. 6, no. 6, pp. 769–798, 1989.



James D. English (S'95–M'97) received the B.S. degree from Virginia Polytechnic Institute and State University, Blacksburg, the M.E. degree from the University of Florida, Gainesville, and the Ph.D. degree from Purdue University, West Lafayette, IN, all in electrical engineering, in 1988, 1991, and 1996, respectively.

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



Anthony A. Maciejewski (S'82–M'87) received the B.S.E.E., M.S., and Ph.D. degrees in electrical engineering from Ohio State University, Columbus, in 1982, 1984, and 1987, respectively. In 1985–1986, he was an American Electronics Association Japan Research Fellow at the Hitachi Central Research Laboratory in 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.