[23] E. Ottaviano and M. Ceccarelli, "Optimal design of CaPaMan (Cassino Parallel Manipulator) with a specified orientation workspace," *Robotica*, vol. 20, no. 2, pp. 159–166, 2002.
[24] J.-P. Merlet, "Determination of the orientation workspace of parallel manipulators," *J. Intell. Robot. Syst.*, vol. 13, pp. 143–160, 1995.
[25] L. M. Bugayevskiy and J. P. Snyder, *Map Projections: A Reference Manual*. London, U.K.: CRC, 1995.
[26] S. Payandeh, T. Li, and H. Van Der Wal, "Devices for positioning implements about fixed points," US Patent Appl. 20040024387, Feb. 5, 2004.
[27] C. M. Gosselin and J. Angeles, "The optimum kinematic design of a spherical three-degree of freedom parallel manipulator," *J. Mech., Transmiss., Autom. Des.*, vol. 111, pp. 202–207, 1989.
[28] I. A. Bonev, D. Chablat, and P. Wenger, "Working and assembly modes of the Agile Eye," in *Proc. IEEE Int. Conf. Robot. Autom.*, Orlando, FL, May 2006, pp. 2317–2322.

Fault Tolerance of Parallel Manipulators Using Task Space and Kinematic Redundancy

Yong Yi, John E. McInroy, and Yixin Chen

***Abstract*—When a parallel manipulator suffers from failures, its performance can be significantly affected. Thus, fault tolerance is essential for task-critical applications or applications in which maintenance is hard to implement. In this paper, we consider three types of common strut failures corresponding to stuck joints, unactuated actuators, or the complete loss of struts, respectively. The impacts of different failures on the kinematics of a manipulator are examined, and the task space redundancy and kinematic redundancies are used to help overcome these failures. In addition, local measures of fault tolerance and their properties are analyzed. These measures can be helpful in architecture design and path planning.**

***Index Terms*—Fault tolerance, kinematic redundancy, parallel manipulators, task priority.**

I. Introduction

Fault tolerance is a major consideration for nuclear, military, and space applications. The dangerous and long-distance nature of these applications often makes maintenance very difficult and even impossible. Moreover, a single failure may jeopardize the entire mission or cause costly down-time.

Failures in robots come in various types. Fault-tolerance strategies usually tend to convert complex failures into several simple failure modes which are easy to deal with [1]. For example, if a motor behaves unpredictably or a joint is sluggish, the failure can be converted to a position failure with a brake. Three typical component failure modes are summarized here.

- *Position failure*. Position failure, also called a joint locked failure, refers to the case when one joint is locked in place and cannot move. This happens either because the motor failure directly results in an inability to move or because brakes are applied to prevent unpredictable behaviors due to the joint fatigue, uncontrollable motors, and failed sensors.
- *Torque failure*. Torque failure, also known as free swing failure [2], refers to a hardware or software fault in a robotic manipulator that causes the loss of torque (or force) on an actuator. Examples include a ruptured seal on a hydraulic actuator, the loss of electric power, and a mechanical failure in a drive system.
- *Hard failure*. Hard failure refers to the case when a strut is totally lost. It is mainly caused by mechanical fatigue or a blown-off strut. Parallel manipulators are superior at tolerating hard failures, since the manipulator may continue to implement the task even with some struts lost.

Fault tolerance typically incorporates a failure detection and identification scheme followed by failure recovery. This paper presents schemes to tolerate different failures once the failures are detected. Literature regarding fault tolerance can be found in [2] and [3] for serial manipulators, and [4] for parallel manipulators.

For serial manipulators, position failures can be easily tolerated. Once a joint is stuck, other joints can take over its workload and move the end-effector to its goal [3]. However, overcoming torque failures is more complicated. In [2], English and Maciejewski control the underactuated serial manipulator after torque failures by selecting the configurations which minimize a failure-susceptibility measure. Then, they improve the postfailure performance by converting torque failures to position failures through active braking [5].

Parallel manipulators have multiple struts working together. On the one hand, they have a better ability to endure the loss of actuations or constraints, since the struts back up each other. On the other hand, they also suffer from mutual conflict. Consequently, tolerating stuck joints is not as easy. McInroy and Chen suggest tolerating actuator position failure in a Gough–Stewart platform by sacrificing one redundant degree of freedom (DOF) [4].

This paper is different from the work mentioned above in the following ways.

- The impacts of different failures on the kinematics of a parallel manipulator and possible solutions using redundancy are investigated, and convenient ways to reconfigure the postfailure kinematics are found.
- Local measures of fault tolerance describing the impact of these failures on the manipulability are defined and determined by the nominal kinematics. The results can be useful in the design of fault-tolerant manipulators, as well as path planning for task-prioritized manipulators.

This paper is arranged as follows. First, the forward kinematics of a general parallel manipulator are derived in Section II. Then, in Section III, different failures are analyzed, and possible fault-tolerant strategies are explored. Furthermore, local measures of tolerance with these faults are defined and determined. Finally, in Section IV, these theories are applied on a case study, in which a nonredundant fast steering mirror (FSM) and two kinematically redundant FSMs are compared with respect to their fault-tolerance performance.

Notation:

- Given a matrix A, we use $\tilde{A}$ to denote the transposed annihilator of $A^T\,(A\tilde{A}=0)$. Its columns form an orthonormal basis for the right null space of A. Similarly, we use ${}^L\tilde{A}$ to denote the annihilator of $A\,({}^L\tilde{A}A=0)$. Its rows form an orthonormal basis for the left null space of A.
- Given a matrix A, we use $A_{i_1,i_2,\ldots,i_f}$ to denote the submatrix of A whose rows are composed of rows $i_1,i_2,\ldots,i_f$ of A.

II. Kinematic Model of a General Parallel Manipulator

In many applications, there are redundant DOFs in the task space, thus the DOFs can be sacrificed for some particular reasons, such as

Manuscript received April 1, 2005; revised October 11, 2005. This paper was recommended for publication by Associate Editor R. Roberts and Editors I. Walker and H. Arai upon evaluation of the reviewers' comments. This work was supported by the Missile Defense Agency and Army Research Office under Grant DAAD19-00-1-0153 and Grant DAAD19-02-1-0102.

Y. Yi and J. E. McInroy are with the Electrical and Computer Engineering Department, University of Wyoming, Laramie, WY 82071 USA (e-mail: yongyiyh@yahoo.com; mcinroy@uwyo.edu).

Y. Chen is with the Department of Computer Science, University of New Orleans, New Orleans, LA 70148 USA (e-mail: yixin@cs.uno.edu).

Digital Object Identifier 10.1109/TRO.2006.878973

enhancing system reliability, avoiding obstacles and singularities in the workspace, optimizing kinematic performance indices, or tolerating actuator failures. The manipulator performing a prioritized manipulation task is called a prioritized manipulator. The task-space DOFs of a prioritized manipulator can be divided into major DOFs (MDOFs), which are critical in performing a task, and secondary DOFs (SDOFs), which are less important.

For a general manipulator, let v be the task space velocity partitioned as $\left[\begin{smallmatrix} v=v_m \\ v_s \end{smallmatrix}\right]$, where v_m (v_s) corresponds to the major (secondary) DOFs. Notice that if the manipulator is not prioritized, v_s is empty. Let $\dot{\theta}_a$ ($\dot{\theta}_p$) denote the vector of actuator (passive joint) velocity. Then, we have the following theorem.

Theorem 1: The differential kinematics model of a parallel manipulator can be written as

$$\begin{bmatrix} v_m \\ v_s \end{bmatrix} = \begin{bmatrix} J_m \\ J_s \end{bmatrix} \dot{\theta}'_a + J_\xi \xi \tag{1}$$

with constraint

$$\dot{\theta}_a = T\dot{\theta}'_a \tag{2}$$

where T is an $N \times n\,(N \geq n)$ mapping matrix with orthonormal columns, $\dot{\theta}'_a$ and ξ are two free vectors, $\left[\begin{smallmatrix} J=J_m \\ J_s \end{smallmatrix}\right]$ is an $l \times n$ Jacobian matrix, and J_m (J_s) is the $m \times n$ ($(l-m) \times n$), $m \leq l$ submatrix of J corresponding to v_m (v_s).

Proof: For a general parallel manipulator, the differential kinematics can be written as [6]

$$v = J_{Ta}\dot{\theta}_a + J_{Tp}\dot{\theta}_p \tag{3}$$

$$J_{Ca}\dot{\theta}_a + J_{Cp}\dot{\theta}_p = 0 \tag{4}$$

where J_* denotes a Jacobian. Left multiplying (4) by ${}^L\tilde{J}_{Cp}$ on both sides and solving for $\dot{\theta}_a$ yields (2), with

$$T = \widetilde{{}^L\tilde{J}_{Cp}J_{Ca}}. \tag{5}$$

Note that $T = I$ when ${}^L\tilde{J}_{Cp}J_{Ca} = 0$. Then, solving (4) for $\dot{\theta}_p$ in terms of $\dot{\theta}_a$, and inserting $\dot{\theta}_p$ into (3), we have (1) with $J = \left[\begin{smallmatrix} J_m \\ J_s \end{smallmatrix}\right] = (J_{Ta} - J_{Tp}J^{\dagger}_{Cp}J_{Ca})T$ and $J_\xi = J_{Tp}\tilde{J}_{Cp}$. □

Remarks:

- If $J_{Tp}\tilde{J}_{Cp} \neq 0$, then the manipulator is underconstrained, meaning that unactuated task-space motion may occur due to insufficient independent constraints. This should be avoided, since the manipulator is at an unstable singularity.
- If ${}^L\tilde{J}_{Cp} \neq 0$, then the motions of actuators are constrained. This only occurs when the independent constraints outnumber the passive joints, as this makes ${}^L\tilde{J}_{Cp} \neq 0$ possible. It can occur in several ways:
  - — by actuating more joints than the minimum necessary to make $J_\xi = 0$ without altering the kinematic chain (termed *overactuation*);
  - — by adding additional struts, complete with actuators (termed *redundant strut*);
  - — by constraining some DOFs with additional fixtures (termed *overconstraint*).

  The first two cases are kinematically redundant. Kinematic redundancy is widely used in areas such as singularity avoidance [7], [8] and torque optimization [9]. However, its application in improving fault tolerance has not been fully explored.
- A manipulator does not lose DOFs by kinematic redundancy. In contrast, an overconstrained manipulator loses DOFs because of inappropriately exerted constraints. This should be nominally avoided for any well-designed manipulator, but could happen when failures occur, i.e., joints are stuck.
- If v_s and J_s are not empty, the manipulator has redundancy in its task space [4], [10]. Unlike kinematic redundancy, which comes from the manipulator itself, this redundancy in the task space depends on specific application.

In this paper, we will use the task-space redundancy and kinematic redundancy to improve tolerance to various strut failures.

III. Fault Tolerance

Once a failure occurs, it is essential to rederive the kinematic model and check whether the failure can be tolerated. A failure can be tolerated only if the system after the failure is kinematically stable and all of the desired DOFs are retained. Notice that even if a system is kinematically unstable, the structure may stay at a local equilibrium due to gravity and may be controlled within a subspace. However, it cannot resist a disturbance in the direction of unactuated task motion. In this paper, we always ensure kinematic stability so that the postfailure performance would not deteriorate much. Three typical failure modes are considered separately. The mixture of them can be easily calculated by combining the methods below.

A. Position Failure

Mathematically, position failure is represented as $\dot{\theta}_i = 0$, where θ_i is the failed joint. The work in [4] suggests rederiving the postfailure model by including the new constraint in (4). However, since the failed joints are often actuators [11], there is a more straightforward way to reconfigure the new Jacobian directly from the nominal one.

Suppose that f position failures happen in actuators $i_1, i_2, \ldots, i_f$, $f \leq n - m$. Let $T_f = T_{i_1,i_2,\ldots,i_f}$ consist of the rows of T corresponding to the failed actuators. Then, we have $T_f\dot{\theta}'_a = 0$. Solving for $\dot{\theta}'_a$ yields $\dot{\theta}'_a = \tilde{T}_f\dot{\theta}''_a$, where $\dot{\theta}''_a$ is an arbitrary $(n-f) \times 1$ vector. Notice that $\dot{\theta}''_a$ has a lower dimension than $\dot{\theta}'_a$, indicating that the motions of actuators are more constrained, and the manipulator loses DOFs. Then, the $l \times (n-f)$ postfailure Jacobian $\hat{J}$, and the $N \times (n-f)$ mapping matrix $\hat{T}$, can be reconfigured by

$$\hat{J} = J\tilde{T}_f \tag{6}$$

$$\hat{T} = T\tilde{T}_f. \tag{7}$$

It can be shown that $\hat{T}$ has orthonormal columns ($\hat{T}^T\hat{T} = I_{(n-f)\times(n-f)}$).

Due to the new constraints imposed by position failure, the postfailure manipulator becomes overconstrained and loses DOFs. Therefore, a position failure may be tolerated only if the nominal system has redundant DOFs.

B. Local Measure for Position Failure Tolerance

Suppose that there are $f \leq n - m$ independent position failures occurring in actuators $i_1, i_2, \ldots, i_f$. A local position failure-tolerance measure can be the relative manipulability index, defined as the percent of manipulability retained after failures [12] as

$$p_{i_1,i_2,\ldots,i_f} = \frac{w(\hat{J}_m)}{w(J_m)} \tag{8}$$

where $w(A) = \max(\sqrt{\det(AA^T)}, \sqrt{\det(A^TA)})$ is the classical measure of manipulability. When a manipulator is not at its singularity, it is just the product of the nonzero singular values. The measure defined by (8) examines the reduced manipulability index after failures relative to the nominal one. Notice that this quantity ranges from zero to one, and is independent of the scaling applied to the linear or rotational components of J due to the normalization. When $w(\hat{J}_m) = 0$, the manipulator is intolerant to the failure. Large $p_{i_1,i_2,\ldots,i_f}$ is desired, since it indicates that the manipulator would not have much reduction in its manipulability after a failure. Given that nominal manipulability requirements are satisfied, this measure can be used in designing manipulators or selecting configurations with the best position failure tolerance (maximize $p_{i_1,i_2,\ldots,i_f}$ over all configurations).

To determine the relative manipulability index, one can clearly calculate the reduced manipulability index for each actuator failure and then divide by the nominal manipulability index. However, it is possible to calculate this relative index directly from the mapping matrix T and the null space of the Jacobian J_m.

Theorem 2: Suppose that a task-prioritized parallel manipulator is in a nonsingular configuration, and that there are $f \leq n - m$ independent position failures occurring in actuators $i_1, i_2, \ldots, i_f$. Then, the local position failure-tolerance measure is given by

$$p_{i_1,i_2,\ldots,i_f} = \frac{w(\hat{J}_m)}{w(J_m)} = \frac{w\left((T\tilde{J}_m)_{i_1,i_2,\cdots,i_f}\right)}{w\left(T_{i_1,i_2,\ldots,i_f}\right)}. \quad (9)$$

Remarks:

- A similar approach that determines the position failure-tolerance measure for serial manipulators has been discussed in [12]. *Theorem 2* is the extension to parallel manipulators.
- When $f = 1$, (9) becomes

$$p_i = w\left(\frac{T_i}{\|T_i\|}\tilde{J}_m\right). \quad (10)$$

- The theorem has a physically intuitive interpretation. For the one-failure case, the magnitude of an actuator's contribution to the null space of J_m is a measure of how much redundancy resides in that particular actuator. Thus, the more redundancy associated with an actuator, the more tolerant the manipulator is to a failure in that actuator. For multiple failures, similar physical meaning holds. Notice that since the actuators are correlated, the impact of multiple actuator failures is different from the simple accumulation of individual failures.

To prove *Theorem 2*, the following theorem from [12] shall be used.

Theorem 3: Let A be an $m \times n\,(m < n)$ matrix with full rank and $\hat{A}$ be the matrix obtained from A by substituting columns $i_1, i_2, \ldots, i_f\,(f \leq n - m)$ with zeros. Then, $w(\hat{A})/w(A) = w((\tilde{A})_{i_1,i_2,\ldots,i_f})$.

Proof of Theorem 2: Let $T_f = T_{i_1,i_2,\ldots,i_f}$. Taking the singular value decomposition (SVD) of T_f, we get $T_f = U[S\ 0]\left[\begin{smallmatrix} V_1^T \\ V_2^T \end{smallmatrix}\right]$, where U and $V = [V_1\ V_2]$ are unitary matrices and S is a diagonal matrix. Notice that in (5), the mapping matrix T is not unique. Letting $\dot{\theta}'_a = V\dot{\theta}''_a$ generates another mapping matrix $T' = TV$, with new Jacobian $J'_m = J_m V$. If actuators $i_1, i_2, \ldots, i_f$ fail, then we have $0 = (\dot{\theta}_a)_{i_1,i_2,\ldots,i_f} = US(\dot{\theta}''_a)_{1,2,\ldots,f}$. Since U and S are $f \times f$ full-rank square matrices, it follows that elements $1, 2, \ldots, f$ of $\dot{\theta}''_a$ are zeros. Thus, the postfailure Jacobian can be reconfigured simply by substituting columns $1, 2, \ldots, f$ of J'_m with zeros. Then, according to *Theorem 3*, $p_{i_1,i_2,\ldots,i_f} = w((\tilde{J}'_m)_{1,2,\ldots,f}) = w(V_1^T\tilde{J}_m)$. Since $V_1^T = S^{-1}U^T T_f$, we have $p_{i_1,i_2,\ldots,i_f} = w(S^{-1}U^T T_f \tilde{J}_m) = w(T_f\tilde{J}_m)/w(T_f)$. □

C. Torque Failure

Torque failure happens when an actuator moves passively because its motor cannot produce torque (the torque exerted on the failed actuator is zero). Thus, the failed actuator becomes a passive joint [4].

After torque failures, the kinematic relationship of the manipulator remains the same. However, failed actuators should be excluded from the model since they become passive joints. Suppose there are $f \leq (N - n)$ independent torque failures occurring in actuators $i_1, i_2, \ldots, i_f$, and define $\hat{\theta}_a$ as the $(N - f) \times 1$ vector of the remaining actuators, and T_r as the $(N - f) \times n$ matrix composed of the rows of T associated with the remaining struts as follows:

$$T_r = T_{\{1,\ldots,N\}\setminus\{i_1,i_2,\ldots,i_f\}}.$$

Letting $T_r = [U_1\ U_2]\left[\begin{smallmatrix} \Sigma_1 \\ 0 \end{smallmatrix}\right]V^T$ be the SVD of T_r, and then defining $\dot{\hat{\theta}}_a = U_1\dot{\theta}''_a$, where $\dot{\theta}''_a$ is an $n \times 1$ free vector, yields the $l \times n$ postfailure Jacobian $\hat{J}$ and $(N - f) \times n$ mapping matrix $\hat{T}$

$$\hat{J} = JV\Sigma_1^{-1} \quad (11)$$

$$\hat{T} = U_1. \quad (12)$$

If T_r has full column rank ($\tilde{T}_r = 0$), then the motion of the failed actuators can be uniquely determined by the remaining actuators. Thus, the torque failure can be tolerated. Specifically, if T_r is a full-rank square matrix, then $\hat{T} = I$ and $\hat{J} = JT_r^{-1}$. On the other hand, if T_r degenerates, then the manipulator loses kinematic stability, and the failure cannot be tolerated due to the unactuated motion incurred.

D. Local Measure for Torque Failure Tolerance

A local torque failure-tolerance measure can be defined as

$$t_{i_1,i_2,\ldots,i_f} = \frac{w(J)}{w(\hat{J})} \quad (13)$$

where actuators $i_1, i_2, \ldots, i_f$ are involved in the failures. Since after torque failures the manipulator has increased velocity manipulability, it is clear that the measure ranges from zero to one. When the manipulator loses kinematic stability ($w(\hat{J})$ goes to infinity), it is intolerant to the failure. Like the position failure-tolerance measure, this measure can also be useful in designing manipulators with the best torque failure tolerance (maximize the minimum $t_{i_1,i_2,\ldots,i_f}$ over all configurations).

The following theorem gives an approach to derive $t_{i_1,i_2,\ldots,i_f}$ directly from T.

Theorem 4: Suppose a parallel manipulator is in a nonsingular configuration, and that there are $f \leq N - n$ independent torque failures involving actuators $i_1, i_2, \ldots, i_f$. Then, the local torque failure-tolerance measure is given by

$$t_{i_1,i_2,\ldots,i_f} = \frac{w(J)}{w(\hat{J})} = w(T_r) \quad (14)$$

where T_r denotes the $(N - f) \times n$ matrix composed of the rows of T associated with the remaining struts. Furthermore, if $f = N - n$, the following relationship holds:

$$\sum_{i_1 < i_2 < \cdots < i_f} t^2_{i_1,i_2,\ldots,i_f} = 1. \quad (15)$$

Proof: Equation (14) clearly follows from (11). If $f = N - n$, then, by the Binet–Cauchy theorem, we have $\sum_{i_1<i_2<\cdots<i_f} t^2_{i_1,i_2,\ldots,i_f} = \sum_{i_1<i_2<\cdots<i_f} w^2(T^T_{\{1,\ldots,N\}\setminus\{i_1,i_2,\ldots,i_f\}}) = w^2(T^T) = 1$. □

The physical interpretation of *Theorem 4* is that the degree of torque failure tolerance is determined by how much the remaining actuator velocities are related to the mapped actuator velocities.

E. Hard Failure

Hard failure is caused by mechanical fatigue or a blown-off strut. In either case, the system acts as if the failed strut is totally lost. Since a parallel manipulator consists of several struts, it is possible that the whole manipulator survives with one or more missing struts.

A hard failure can be equivalent kinematically to corresponding torque failures, on the condition that the failed strut itself has full manipulability at its tip. For example, it is typical that each strut of a 6-DOF manipulator has full manipulability (six DOFs) at its tip. For planar manipulators with redundant struts, each redundant strut also has full manipulability (three DOFs); for manipulators with only very restricted workspace (basically around one point), it is possible that at the particular working point, although the redundant strut of a manipulator does not have six DOFs, the constraints it enforces are dependent upon the preexisting manipulator constraints. In these cases, a strut with all its actuators having torque failures forms a passive connection to the manipulator. Thus, kinematically, it behaves like it is totally lost. The equivalence is beneficial, since the rederivation of kinematics after hard failure is time-consuming, while the kinematics after torque failure can be readily reconfigured by the procedure described in Section III-C.

However, there are many other manipulators wherein this equivalence does not exist. Then the postfailure kinematics should be derived from the beginning. There is no easy way of reconfiguring it from the nominal one. Generally, these manipulators would gain more DOFs, if they remain kinematically stable after failures.

Suppose a parallel manipulator is in a nonsingular configuration, and that there are f independent hard failures involving struts $i_1, i_2, \ldots, i_f$. We define the fault-tolerance measure as

$$h_{i_1,i_2,\ldots,i_f} = \frac{w(J)}{w(P_J \hat{J})} \tag{16}$$

where $P_J = J(J^T J)^{-1} J^T$ is a projection matrix which maps a vector into the space spanned by columns of J. This measure is useful to investigate the manipulability change in the nominal task space. If no DOF is gained after the failure, then $P_J \hat{J} = \hat{J}$, and the measure defined by (16) is the same as (13). Similar to torque failure, this measure ranges from zero to one. When the manipulator loses kinematic stability, it is intolerant to the failure.

One would notice that this measure does not reflect the status of the gained DOFs. Thus, there is a chance that the manipulator is unstable or close to singularity in the direction of the gained DOFs, while it still looks satisfactory if only the nominal DOFs are considered. Therefore, this measure represents just one aspect of the postfailure performance, and it is desirable to combine with other measures, such as maximum manipulability.

F. Discussion

So far, we have introduced the postfailure kinematics reconfiguration and the corresponding local measures of fault tolerance. In this section, we consider how a failure may be tolerated by redundancy. One may combine these strategies according to specific fault-tolerance requirements.

Position Failure: Due to the new constraints imposed by the stuck joints, a parallel manipulator would lose DOFs after position failure, and the failure can only be tolerated by redundancy in the task space. Notice that this is different from the position failure case for serial manipulators. For serial manipulators, a stuck joint only stops contributing to the manipulability, which can be made up by other redundant joints. However, for parallel manipulators, we cannot view a failed joint as just lost, as we can do for serial manipulators. Because each joint is one component of the closed loops, it can add new constraints to the manipulator when it is stuck. The case study in Section IV will show that if there is no redundancy in the task space, simply actuating some passive joints does not help the manipulator retain its manipulability, since the redundant actuation does not help the manipulator release any constraint.

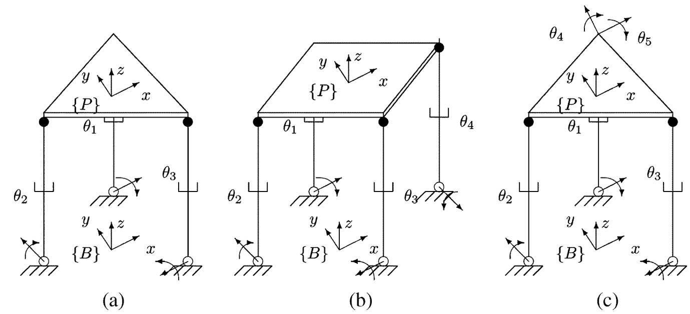

Fig. 1. FSMs. (a) Nonredundant FSM. (b) Four-strut FSM. (c) Overactuated FSM.

For manipulators having no redundant DOFs, two ways may be helpful to solve the problem. First, if a manipulator has less than six DOFs, then redundant actuators can be added to struts, such that the nominal manipulator gains redundant DOFs. Second, for a 6-DOF manipulator, redundant struts can be added to the nominal manipulator, such that the struts with stuck joints can be automatically disconnected when failures happen. In this case, a position failure is actually converted to a hard failure.

Torque failure: Torque failures may result in kinematic instability and can be tolerated by kinematic redundancy (overactuation or redundant strut) of the nominal manipulator. Notice that adding a redundant strut could also help maintain manipulability, since it provides redundant actuators.

Hard failure: We would like to point out that torque failure and hard failure have some common effects. They both tend to reduce constraints (kinematically, torque failure can be viewed as a controlled joint motion becoming free), thus causing instability. While it is obvious that adding redundant struts would improve hard-failure tolerance, notice that overactuation would also help, as long as it provides enough redundant actuations to prevent uncontrolled motion due to the loss of constraints. It is possible that an overactuated manipulator would gain more active DOFs after hard failure. Nevertheless, this causes no problem, on the condition that the kinematic stability is ensured.

The case study in Section IV will show that a manipulator with a redundant strut can tolerate a torque failure, and an overactuated manipulator can tolerate hard failure.

IV. Case Study

Here, we first take a four-strut FSM at its home configuration as an example to show the calculation of the postfailure kinematics and fault-tolerance measures. Then, a nonredundant FSM and two redundant FSMs are compared with respect to their fault-tolerance ability.

The FSMs shown in Fig. 1 are for pointing applications, in which only the two rotations around the x- and y-axes are critical. Fig. 1(a) shows a nonredundant three-strut FSM, Fig. 1(b) shows an FSM with one redundant strut, and Fig. 1(c) shows an overactuated FSM with two passive joints (θ_4 and θ_5) actuated. $\{B\}$ and $\{P\}$ are the task frame and the base frame, respectively. For each FSM, the struts are arranged around a circle symmetrically. Each strut has a prismatic actuator that can change the length of the strut. The struts connect to the payload with spherical joints and to the base with 1-DOF hinges, which allow rotations toward or away from the center. Let θ_i be the length of strut i, thus $\dot{\theta}_a$ is the actuator velocity vector composed of $\dot{\theta}_i$.

Let us take the four-strut FSM [Fig. 1(b)] at its home configuration as an example to show the calculation of the postfailure kinematics and fault-tolerance measures. At its home configuration, the manipulator has three DOFs. Let $v^T = [\omega_x \ \omega_y \ v_z]^T$, where ω_x, ω_y, v_z denote the

TABLE I
FAULT-TOLERANCE MEASURES FOR THREE MANIPULATORS AT HOME CONFIGURATION

Measure	PF	TF	HF
3-strut FSM	0.5775	0	0
4-strut FSM	0.5775	0.5	0.5
over-actuated FSM	0.6742[a] 0[b]	0.5164[a] 0.7746[b]	0.3464[c] 0[d]

[a] θ_1, θ_2, or θ_3 fails
[b] θ_4 or θ_5 failes
[c] Strut 2 or 3 fails
[d] Strut 1 fails

rotational velocity around the x- and y-axes and the translational velocity along the z-axis, respectively. Then, the Jacobian J (written in the base frame $\{B\}$) and the mapping matrix T are

$$J = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$$
$$T = \begin{bmatrix} 0.707 & 0 & 0.5 \\ 0 & 0.707 & 0.5 \\ -0.707 & 0 & 0.5 \\ 0 & -0.707 & 0.5 \end{bmatrix}.$$

When actuator 4 is stuck ($\dot{\theta}_4 = 0$), the new $\hat{J}$ and $\hat{T}$ can be calculated as $\hat{J} = J\tilde{T_4}$ and $\hat{T} = T\tilde{T_4}$, where T_4 is the fourth row of T. The position failure-tolerance measure p_4 is calculated by (10): $p_4 = w((T_4/\|T_4\|)\tilde{J}_m)$, where J_m consists of the first two rows of J corresponding to ω_x and ω_y.

Similarly, if actuator 4 has torque failure, the new kinematic parameters can be calculated as $\hat{J} = JT_r^{-1}$, and $\hat{T}$ is the identity matrix, where T_r consists of the first three rows of T. The torque failure-tolerance measure t_4 is calculated by (14): $t_4 = w(T_r)$.

At the home configuration, since the hinge on strut 4 enforces a dependent constraint, strut 4 forms a passive connection if it loses actuation. Hence, the hard failure on strut 4 is kinematically equivalent to the corresponding torque failure on actuator 4. However, generally this is not true for other configurations.

Fault-tolerance measures for the three manipulators at home configuration are listed in Table I, where PF, TF, and HF stand for position failure, torque failure, and hard failure.

Remarks:

- At home configuration, all of the manipulators have the same three DOFs. Since, for pointing application, only the two rotations ω_x and ω_y are critical, the manipulators have one redundant DOF (translation along the z-axis). After position failure, the manipulators retain the two MDOFs in most cases. However, translation along the z-axis is dependent on these two DOFs as a result of the new constraint. Note that the kinematic redundancy does not help in position failure. The failure is tolerated by sacrificing the z-axis translation. Thus, even a three-strut FSM can tolerate this failure.
- At the home configuration, the overactuated FSM can tolerate some hard failures and the four-strut FSM can also tolerate torque failure.
- At home configuration, the overactuated FSM sometimes outperforms the four-strut one. However, the overall performance is worse, since faults occurring on particular joints or struts cannot be tolerated. Moreover, it is not symmetric, which means that the control of this manipulator can be more complex. Thus, for high-precision applications in which only very limited workspace is involved, such as laser weapon pointing or high-precision motion control for telescopes, the four-strut architecture seems to be a better choice.
- At non-home configurations, the four-strut FSM has only two DOFs [either $(\omega_x, \dot{z})$ or $(\omega_y, \dot{z})$]. Thus, if the application requires a large workspace, the overactuated manipulator can be a better choice.
- The analysis of fault tolerance at non-home configurations is similar to the home configuration case. Note that for the four-strut FSM, hard failure is not equivalent to torque failure anymore, and the postfailure kinematics have to be rederived from the beginning. In this case, the manipulator gains one more DOF because the constraint enforced by the hinge of the lost strut is released.

V. CONCLUSION

Position failure results in the loss of DOFs and may be tolerated by task-space redundancy. Torque failure and hard failure may cause kinematic instability and may be tolerated by the kinematic redundancy of the original manipulator. In most cases, the postfailure kinematics can be directly derived from the nominal case by slight reconfiguration. Then, local measures of fault tolerance are defined by comparing the manipulability change after failures with the original manipulability. Convenient methods are developed for determining the local measures from the nominal kinematic parameters. The results can be useful in the design of fault-tolerant manipulators as well as in path planning for task-prioritized manipulators.

REFERENCES

[1] M. L. Visinsky and J. R. Walker, "Robotic fault detection and fault tolerance: A survey," *Reliable Eng. Syst. Safety*, vol. 46, pp. 139–158, 1994.

[2] J. D. English and A. A. Maciejewski, "Fault tolerance for kinematic redundant manipulators: Anticipating free-swing joint failures," *IEEE Trans. Robot. Autom.*, vol. 14, no. 4, pp. 566–575, Aug. 1998.

[3] C. L. Lewis and A. A. Maciejewski, "Fault tolerant operation of kinematically redundant manipulators for locked joint failures," *IEEE Trans. Robot. Autom.*, vol. 13, no. 4, pp. 622–629, Aug. 1997.

[4] Y. Chen, J. E. McInroy, and Y. Yi, "Optimal, fault tolerant velocity and static force mapping for prioritized manipulation," *IEEE Trans. Robot. Autom.*, vol. 19, no. 4, pp. 680–691, Aug. 2003.

[5] J. D. English and A. A. Maciejewski, "Failure tolerance through active braking: A kinematic approach," *Int. J. Robot. Res.*, vol. 20, no. 4, pp. 287–299, Apr. 2001.

[6] J. Wen and L. Wilfinger, "Kinematic manipulability of general constrained rigid multibody systems," *IEEE Trans. Robot. Autom.*, vol. 15, no. 3, pp. 558–567, Jun. 1999.

[7] S. J. Ryu *et al.*, "Eclipse: An overactuated parallel machanism for rapid machining," in *Proc. Symp. Robot., Manip., Syst.*, Paris, France, Jul. 1998, pp. 79–86.

[8] G. F. Liu and Y. L. Wu, "Analysis and control of redundant parallel manipulators," in *Proc. IEEE Int. Conf. Robot. Autom.*, Seoul, Korea, May 2001, pp. 3748–3754.

[9] C. A. Klein and S. Kittivatcharapong, "Optimal force distribution for the legs of a walking machine with friction cone constraints," *IEEE Trans. Robot. Autom.*, vol. 6, no. 1, pp. 73–85, Feb. 1990.

[10] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task priority based redundant control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 4, pp. 3–15, 1987.

[11] V. M. and S. Tosunoglu, "Fault tolerance for robotics and mechanical systems: An introduction survey for rapid machining," in *Proc. 3rd Biennial Joint Conf. Eng. Syst. Des. Anal.*, 1996, vol. 74, pp. 259–264.

[12] R. G. Roberts and A. A. Maciejewski, "A local measure of fault tolerance for kinematically redundant manipulators," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 543–553, Aug. 1996.