Over-constrained rigid multibody systems: differential kinematics, and fault tolerance

Yong Yi, John E. McInroy and Yixin Chen

Dept. of Electrical Enginerring, University of Wyoming,
Laramie, WY 82071

ABSTRACT

Over-constrained parallel manipulators can be used for fault tolerance. This paper derives the differential kinematics and static force model for a general over-constrained rigid multibody system. The result shows that the redundant constraints result in constrained active joints and redundant internal force. By incorporating these constraints, general methods for overcoming stuck legs or even the complete loss of legs are derived. The Stewart platform special case is studied as an example, and the relationship between its forward Jacobian and its inverse Jacobian is also found.

Keywords: Over-constrained, kinematics, fault tolerance

1. INTRODUCTION

A multibody system is a kind of parallel mechanism, which consists of multiple serial manipulators connected together with multiple closed kinematic loops. For a generally constrained multibody mechanism, typically the number of the independent constraints equals the number of the passive joints so that the system has the maximum degrees of freedom (DOF's) without unactuated internal motion. If there are not enough constraints to keep the mechanism kinematically stable at the current configuration, then the system is under-constrained. On the other hand, if there are too many constraints, then the system is over-constrained. The system might lose DOF's because of the extra constraints. Many discussions of the generally constrained multibody mechanism are based on the assumptions that the system is neither under-constrained nor over-constrained. For most applications, it is nonsense to discuss the under-constrained case since the system is unstable. However, the over-constrained case is useful for singularity avoidance,[1] for improvement in force control, for kinematic calibration[2],[3] etc. This paper exploits the redundancy of over-constrained systems to tolerate faults.

Fault tolerance is a major consideration for military, space and some manufacturing applications, because a single failure can jeopardize the entire mission or cause costly down-time. Basically, there are two kinds of failures: soft failures and hard failures, depending on whether the failed struts (the serial manipulators which form the multibody system) are present or not.

Soft failures refer to the cases in which the failed struts are present but work improperly. Soft failures can be characterized as two types: position failures and torque failures. Position failure occurs when some joints (both active joints and passive joints are possible) are stuck and cannot move. This can be viewed as adding more constraints to the system since the velocities of the stuck joints must be zero, and the resulting system might become over-constrained. Torque failure occurs when an active joint cannot be actuated actively and becomes a passive joint. In this case, the number of the passive joints increases while the number of the constraints remains unchanged. Consequently, to keep the system stable after failure, the nominal system should be designed as over-constrained. Some work has been done to tolerate position failures. Ref. 4 can tolerate active joint position failure by choosing sacrificed DOF's from redundant DOF's and then reconfiguring the Jacobian matrix. Then Ref. 5 extends the work by using all the redundant DOF's to reconfigure the Jacobian matrix. Ref. 6 provides a general method to re-derive kinematics by treating position failures as increased constraints.

Further author information: (Send correspondence to John. E. McInroy)
John. E. McInroy: E-mail: mcinroy@uwyo.edu, Telephone: 1 307 766 6137

Hard failures refer to the case when some struts are totally lost. It is mainly caused by mechanical fatigue or a blown-off strut. Hard failures will increase the number of passive joints and possibly decrease the number of independent constraints. This can result in kinematic instability. To maintain kinematic stability, when hard failures occur, a high reliable system should be designed as over-constrained.

From the above analysis, we can see that a system may become over-constrained because of position failures. To tolerate hard or torque failures, the nominal system must be over-constrained. However, a general method to analyze an over-constrained system is still unavailable. In an over-constrained system, the active joints are constrained because of the extra constraints. If the joint torque is not properly exerted, the struts will fight against each other, wasting energy and exerting large internal force. Based on the work of Ref. 7 and Ref. 8, the first part of this paper will discuss the kinematics and static force models for over-constrained multibody systems.

This paper is arranged as follows. In section 2, the kinematics and static force models for an over-constrained multibody system will be derived, and the relationship between the forward Jacobian and the inverse Jacobian for a Stewart platform will be found. Soft failures and hard failures are considered in section 3. Finally, these theories are verified on the UW hexapod.

2. KINEMATICS AND STATIC FORCE MODELS

Terminology and Notation

- Spatial velocity at a given frame means the 6×1 vector of $\begin{bmatrix} angular & velocity \\ linear & velocity \end{bmatrix}$, and Spatial force at a given frame means the 6×1 vector of $\begin{bmatrix} torque \\ force \end{bmatrix}$.
- Given a matrix A, we use ${}^L\tilde{A}$ to denote the annihilator of A(${}^L\tilde{A}A = 0$) and ${}^R\tilde{A}$ to denote the transpose of annihilator of $A^T(A\,{}^R\tilde{A} = 0)$.

2.1. Kinematics model

First the method for analyzing differential kinematics will be derived. For a general kinematically constrained rigid multibody system, the differential kinematics model developed by Wen and Wilfinger[8] becomes:

$$V = J_T(\theta)\dot{\theta} \tag{1}$$

with constraint

$$J_C(\theta)\dot{\theta} = 0, \tag{2}$$

where V is the spatial velocity, $\dot{\theta}$ is the joint velocity vector, and θ represents joint positions.

Without loss of generality, θ may be partitioned into active joints (θ_a) and passive joints (θ_p):

$$\theta = \begin{bmatrix} \theta_a \\ \theta_p \end{bmatrix}, J_T = \begin{bmatrix} J_{Ta} & J_{Tp} \end{bmatrix}, J_C = \begin{bmatrix} J_{Ca} & J_{Cp} \end{bmatrix}.$$

Then the model can be rewritten as

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

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

Solving for $\dot{\theta}_p$ in terms of $\dot{\theta}_a$ from (4), we obtain

$$\begin{cases} \dot{\theta}_p = -J_{Cp}^{+}J_{Ca}\dot{\theta}_a + {}^R\tilde{J}_{Cp}\xi \\ {}^L\tilde{J}_{Cp}J_{Ca}\dot{\theta}_a = 0 \end{cases}. \tag{5}$$

where ξ is an unconstrained vector. Inserting (5) into (3), we get

$$V = J\dot{\theta}_a + J_{Tp}\,{}^R\tilde{J}_{Cp}\xi \tag{6}$$

with constraint

$${}^L\tilde{J}_{Cp}J_{Ca}\dot{\theta}_a = 0, \tag{7}$$

where $J = J_{Ta} - J_{Tp}J_{Cp}^{+}J_{Ca}$.

Typically, J_{Cp} is a square matrix with ${}^L\tilde{J}_{Cp} = 0$ and ${}^R\tilde{J}_{Cp} = 0$. Physically, this means that the number of independent constraints is the same as the number of passive joints. In this case, the motion of active joints can uniquely determine the motion of passive joints.

If J_{Cp} is not full column rank (fat), then ${}^R\tilde{J}_{Cp} \neq 0$. This happens when the number of independent constraints is not enough to uniquely determine $\dot{\theta}_p$ given $\dot{\theta}_a$. In other words, the passive joints can move even if the active joints are locked. This will cause unactuated task motion if $J_{Tp}\,{}^R\tilde{J}_{Cp} \neq 0$. In this case, the system is under-constrained. This paper will not consider this unstable situation in detail, but concentrate on the over-constrained case. In the following discussion, we assume that there is no unactuated task motion ($J_{Tp}\,{}^R\tilde{J}_{Cp} = 0$).

On the other hand, if J_{Cp} is not full row rank (tall), then the system is over-constrained, meaning the number of independent constraints is greater than the number of the passive joints. In this case, the motion of active joints must be restricted, because for some $\dot{\theta}_a$, $\dot{\theta}_p$ may have no solution. This over-constrained case will be used in this paper to add fault tolerance to the system.

Generally, the following two cases should be considered.

Case 1. ${}^L\tilde{J}_{Cp}J_{Ca} = 0$.

Obviously, this case includes the situation when J_{Cp} is square with full rank. Another possible situation is, although ${}^L\tilde{J}_{Cp} \neq 0$, ${}^L\tilde{J}_{Cp}J_{Ca} = 0$. This means that the system has redundant but dependent constraints. This case is not considered as over-constrained because it does not provide independent constraint to affect the system's motion. The advantage of having dependent constraints is that when one of them cannot be enforced, such as in hard failures, others can replace it. In this paper, when we talk about redundant constraints, we always refer to the independent constraints.

When ${}^L\tilde{J}_{Cp}J_{Ca} = 0$, (7) is always satisfied, so $\dot{\theta}_a$ is not constrained. Since $J_{Tp}\,{}^R\tilde{J}_{Cp} = 0$ also, the kinematics model becomes

$$V = J\dot{\theta}_a. \tag{8}$$

Case 2. ${}^L\tilde{J}_{Cp}J_{Ca} \neq 0$.

It is clear that ${}^L\tilde{J}_{Cp}J_{Ca}$ cannot be full column rank. Otherwise, $\dot{\theta}_a$ must equal zero in order to satisfy constraint (7). It is meaningless to discuss this case, so we always assume ${}^L\tilde{J}_{Cp}J_{Ca}$ is not full column rank.

To satisfy (7), $\dot{\theta}_a$ should be chosen to be in the null space of ${}^L\tilde{J}_{Cp}J_{Ca}$. Solving for $\dot{\theta}_a$, we get

$$\dot{\theta}_a = {}^R(\widetilde{{}^L\tilde{J}_{Cp}J_{Ca}})\dot{\bar{\theta}}_a = T\dot{\bar{\theta}}_a, \tag{9}$$

where $\dot{\bar{\theta}}_a$ is an arbitrary vector, and $T = {}^R(\widetilde{{}^L\tilde{J}_{Cp}J_{Ca}})$.

The above equation means that because of the redundant constraints on the system, the motions of active joints are constrained. The number of constrained active joints (redundant actuators) equals the number of redundant constraints.

T is a mapping matrix. It maps from a lower dimension, unconstrained space to the active joint space. T has some important attributes:

- T is a tall matrix with full column rank.

- All the column vectors of T are orthonormal.
- $T^T T = I_{n\times n}$, where n is the number of columns in T.
- Because all the singular values of T are 1's, $\|a\|=\|Ta\|$, where a is any $n \times 1$ vector.

Inserting $\dot{\theta}_a$ into (6), we get

$$V = (JT)\dot{\bar{\theta}}_a = \bar{J}\dot{\bar{\theta}}_a, \tag{10}$$

where $\bar{J} = JT$. $\bar{J}$ reveals the relation between the velocities of constrained active joints and the spatial velocity.

2.2. Static force balance

Using the principle of virtual work, the static force model is Ref. 8

$$\begin{bmatrix} \tau \\ 0 \end{bmatrix} = J_T^T f + J_C^T f_C, \tag{11}$$

where f is the spatial force, τ is the active torque vector, and f_C is the internal force. Note that because the passive joints cannot be actuated, the torques corresponding to the passive joints are zero. Internal force f_C is the force that enforces constraints. It can be unambiguously determined if $\mathcal{N}(J_C^T) = \{0\}$, which implies that there is no dependent constraint in the system.

Partitioning J_T and J_C according to active and passive joints, the above equation is rewritten as

$$\begin{array}{l} \tau = J_{Ta}^T f + J_{Ca}^T f_C \\ 0 = J_{Tp}^T f + J_{Cp}^T f_C \end{array}. \tag{12}$$

Left multiplying by $({}^R\tilde{J}_{Cp})^T$ on both sides of the second equation yields

$$(J_{Tp}\,{}^R\tilde{J}_{Cp})^T f = 0. \tag{13}$$

This is the condition that must be satisfied in order to make (12) hold. If $J_{Tp}\,{}^R\tilde{J}_{Cp} = 0$, then the condition is always satisfied. If $J_{Tp}\,{}^R\tilde{J}_{Cp} \neq 0$, then there exists some task space spatial forces ($f \in \mathcal{N}((J_{Tp}\,{}^R\tilde{J}_{Cp})^T)$) that cannot be resisted by internal forces and active torques. This is exactly the case of unstable singularity. So we assume that $J_{Tp}\,{}^R\tilde{J}_{Cp}$=0.

From the constraint (12), we can solve for f_C given f:

$$f_C = -(J_{Tp}J_{Cp}^+)^T f + ({}^L\tilde{J}_{Cp})^T f_\eta, \tag{14}$$

where f_η is an arbitrary vector. In (14), the internal force f_C is decomposed into two orthogonal parts. The first part consists of forces that are required to maintain kinematic stability. The other part consists of redundant internal forces that correspond to the redundant constraints. Redundant internal forces imply fighting among struts, thus causing constrained active joints.

Inserting f_C into (12), we get

$$\tau = J^T f + ({}^L\tilde{J}_{Cp}J_{Ca})^T f_\eta, \tag{15}$$

where $J = (J_{Ta} - J_{Tp}J_{Cp}^+ J_{Ca})$.

Corresponding to the kinematics analysis, we consider the same two cases.

Case 1. ${}^L\tilde{J}_{Cp}J_{Ca} = 0$.

Now (15) becomes

$$\tau = J^T f. \tag{16}$$

In this case, the redundant internal force is zero, which means that there is no fight among struts. All the components of the active torques contribute to generate the spatial force. This is consistent with the fact that all the active joints can move freely.

Case 2. ${}^{L}\tilde{J}_{Cp}J_{Ca} \neq 0$.

If ${}^{L}\tilde{J}_{Cp}J_{Ca}$ is full column rank, then according to (15), any spatial force or any torque exerted on active joints can be resisted only by the internal force. So the system cannot move. Here we only discuss the case when ${}^{L}\tilde{J}_{Cp}J_{Ca}$ is not full column rank.

Remember that we have defined $T = {}^{R}\widetilde{({}^{L}\tilde{J}_{Cp}J_{Ca})}$. Since $T^T({}^{L}\tilde{J}_{Cp}J_{Ca})^T = 0$, we can get the relation between the torque on the active joints and the spatial force by left multiplying T^T on both sides of (15):

$$T^T\tau = T^T J^T f = (JT)^T f. \tag{17}$$

The torque τ exerted on the active joints can be decomposed into two parts

$$\tau = T\bar{\tau} + ({}^{L}\tilde{T})^T\tau_\eta. \tag{18}$$

These two parts are orthogonal to each other. They can be obtained by $\bar{\tau} = T^T\tau$ and $\tau_\eta = {}^{L}\tilde{T}\tau$. Inserting the decomposed τ into (17), and using the fact that $T^T T = I$, we obtain the static force model:

$$\bar{\tau} = (JT)^T f = \bar{J}^T f. \tag{19}$$

In a general mechanism, internal force may determine if a constraint can be enforced. For example, in a multifinger grasp with frictional contacts, each contact force needs to be in the friction cone to ensure that the contact can be sustained. Solving for f_C from(15) yields

$$f_\eta = (({}^{L}\tilde{J}_{Cp}J_{Ca})^T)^+(\tau - J^T f) + {}^{R}\widetilde{({}^{L}\tilde{J}_{Cp}J_{Ca})}^T\xi, \tag{20}$$

where ξ is arbitrary. The second term stands for unresolvable internal force. Previously, from (11), we found that f_C can be unambiguously solved if J_C^T has full column rank, which means that there is no dependent constraint. Actually, this is equivalent to that $({}^{L}\tilde{J}_{Cp}J_{Ca})^T$ has full column rank. This can be proved as follows. Suppose we can find a nonzero vector $y \in \mathcal{N}(({}^{L}\tilde{J}_{Cp}J_{Ca})^T)$. According to the attributes of the annihilator, ${}^{L}\tilde{J}_{Cp}$ has full row rank, thus $\mathcal{N}(({}^{L}\tilde{J}_{Cp})^T) = 0$. Then there exists a nonzero vector $x = ({}^{L}\tilde{J}_{Cp})^T y$ that belongs to $\mathcal{N}(J_C^T)$, because $J_C^T x = \begin{bmatrix} J_{Ca}^T \\ J_{Cp}^T \end{bmatrix} ({}^{L}\tilde{J}_{Cp})^T y = 0$. This implies if J_C^T has full column rank, then $({}^{L}\tilde{J}_{Cp}J_{Ca})^T$ also has full column rank.

Generally, J_C^T has full column rank. So ${}^{R}\widetilde{({}^{L}\tilde{J}_{Cp}J_{Ca})}^T = 0$. Then f_η can be unambiguously solved as:

$$f_\eta = (({}^{L}\tilde{J}_{Cp}J_{Ca})^T)^+(\tau - J^T f). \tag{21}$$

However, if $\mathcal{N}(J_C^T) \neq \{0\}$, then there exists unresolvable internal force because of dependent constraints. Combining (21) with (14) and rearranging terms, the internal force, f_C, can be obtained as

$$f_C = (({}^{L}\tilde{J}_{Cp}J_{Ca})^{+\,L}\tilde{J}_{Cp})^T\tau - [J({}^{L}\tilde{J}_{Cp}J_{Ca})^{+\,L}\tilde{J}_{Cp} + (J_{Tp}J_{Cp}^+)]^T f. \tag{22}$$

The virtual work principle becomes:

$$\tau^T\dot{\theta} = (T\bar{\tau} + ({}^{L}\tilde{J}_{Cp}J_{Ca})^T\tau_\eta)^T T(\dot{\bar{\theta}}_a) = \bar{\tau}^T\dot{\bar{\theta}}_a = f^T V.$$

In the two parts of τ, the first part is the torque that must be exerted on the active joints to balance the force f exerted on the payload. Only this part of the torque does work. The second part will be balanced by the redundant internal force instead of generating spatial force. Typically, we choose $\tau_\eta = 0$ to save energy. However, in some applications where the internal force is important, such as multifinger grasp, the second part can be nonzero to generate certain internal forces.

2.3. Over-constrained velocity and force ellipsoid

In general, systems without redundant constraints can be considered as a special case of over-constrained systems with 0 redundant constraints. So the differential kinematics models of the two cases can be combined together. Then we get

$$V = \bar{J}\dot{\bar{\theta}}_a \tag{23}$$

and

$$\bar{\tau} = \bar{J}^T f, \tag{24}$$

where $\bar{J} = JT$, $\dot{\theta}_a = T\dot{\bar{\theta}}_a$ and $\tau = T\bar{\tau} + ({}^L\tilde{T})^T\tau_\eta$. T is defined as

$$T = \begin{cases} {}^R(\widetilde{{}^L\tilde{J}_{Cp}J_{Ca}}), & {}^L\tilde{J}_{Cp}J_{Ca} \neq 0 \\ I, & {}^L\tilde{J}_{Cp}J_{Ca} = 0 \end{cases}. \tag{25}$$

The DOF of the system is defined to be the maximum rank of $\bar{J}$ at all configurations.[8] If $\bar{J}$ loses rank at some configurations, then we say the system is at an unmanipulable singularity.

The velocity manipulability ellipsoid is an indication of the relative movement capability in task space. Due to the constrained active joints, it is reasonable to define the ellipsoid as the set of task velocities generated by a unit ball in $\dot{\bar{\theta}}_a$ space. Notice that $\|\dot{\theta}_a\| = \|\dot{\bar{\theta}}_a\|$. So the velocity ellipsoid can be defined as

$$\varepsilon_V = \left\{V : V = \bar{J}\dot{\bar{\theta}}_a, \|\dot{\bar{\theta}}_a\| = 1\right\} \tag{26}$$

Similarly, we can define the force ellipsoid as the set of spatial forces that can be applied by the mechanism with active torques constrained on the unit ball of $\bar{\tau}$ space. Since τ_η does not contribute to f, it can be ignored in the ellipsoid calculation. Note that when $\tau_\eta = 0$, $\|\tau\| = \|\bar{\tau}\|$. The force ellipsoid is defined as

$$\varepsilon_F = \{f : \bar{\tau} = \bar{J}^T f, \|\bar{\tau}\| = 1\} \tag{27}$$

2.4. Relationships between the forward Jacobian and the inverse Jacobian for over-constrained Stewart platforms

The Gough-Stewart platform(figure 1) is one of the most popular parallel mechanisms. It is easy to calculate the inverse Jacobian for a Stewart platform[9]:

$$\dot{\theta} = MV, \tag{28}$$

$$M = \begin{bmatrix} [({}^B_PR\,{}^Pp_1\times)\,{}^B\hat{u}_1]^T & {}^B\hat{u}_1^T \\ \vdots & \vdots \\ \left[({}^B_PR\,{}^Pp_n\times)\,{}^B\hat{u}_n\right]^T & {}^B\hat{u}_n^T \end{bmatrix}, \tag{29}$$

where M is the inverse Jacobian matrix relating the spatial velocity V and the active joint velocity $\dot{\theta}_a$. u_i is the unit vector along strut i, ${}^B_PR$ is the rotation matrix from the payload coordinate system to the base coordinate system, p_i is the payload connection point of strut i, and the prescription denotes the coordinate system of reference. This section finds the relation between the inverse Jacobian, M, and the forward Jacobian, $\bar{J}$.

Note that for a stable steward platform, if the spatial velocity $V = 0$, then all the joints are locked (both $\dot{\theta}_a$ and $\dot{\theta}_p$ equal 0). Since T is full column rank, $\dot{\bar{\theta}}_a$ also equals zero. This implies that $\mathcal{N}(\bar{J}) = \{0\}$ or $\bar{J}$ is always full column rank. Next, let's consider the row rank of $\bar{J}$.

Case 1. $\bar{J}$ is full row rank.

In this case $\bar{J}$ is square and full rank. This means that the system has 6 DOF's at the current configuration. From (23), we can solve for $\dot{\theta}_a$

$$\dot{\bar{\theta}}_a = \bar{J}^{-1}V.$$

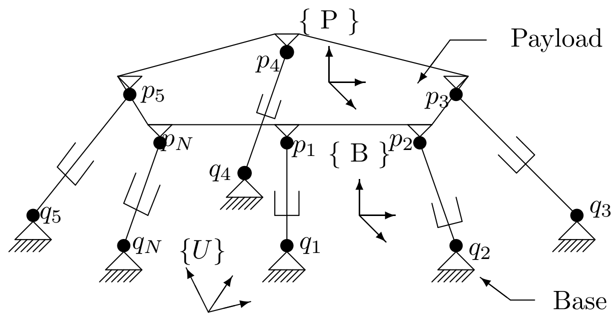

Figure 1. 6-legged Stewart Platform(Hexapod). $\{P\}$ is a Cartesian coordinate frame located at, and rigidly attached to, the payload's center of mass. $\{B\}$ is the frame attached to the (possibly moving) base, and $\{U\}$ is a Universal inertial frame of reference.

Consequently, we can get

$$\dot{\theta}_a = T\bar{J}^{-1}V.$$

Since V is a free vector, (28) yields

$$M = T\bar{J}^{-1}. \tag{30}$$

If the Stewart platform is not over-constrained, then $T = I$, and $M = \bar{J}^{-1}$.

Case 2. $\bar{J}$ is not full row rank.

In this case, the system has less than 6 DOF's and the spatial velocity must belong to the range space of $\bar{J}$. This occurs, for instance, when a joint gets stuck in one position. Substituting $V = \bar{J}\dot{\bar{\theta}}_a$ into (28), we get

$$\dot{\theta}_a = M\bar{J}\dot{\bar{\theta}}_a. \tag{31}$$

Recall that $\dot{\theta}_a = T\dot{\bar{\theta}}_a$, and $\dot{\bar{\theta}}_a$ is a free vector, so we can derive that

$$M\bar{J} = T. \tag{32}$$

If $T = I$, then $M\bar{J} = I$.

3. SOFT FAILURES AND HARD FAILURES

Most failures can be characterized as either soft failures or hard failures depending on whether the failed struts are present or not. This paper extends the work of Ref. 6 by including hard failures.

Soft failure is caused by an abnormal but present strut. It is soft in nature. There are two types of soft failures: position failures and torque failures.

3.1. Position failure

Position failure occurs when some joints are stuck. The result is that the failed joints cannot move (the velocities of the failed joints are zeros). It is equivalent to removing the failed joints from the mechanism. If the stuck joint is an active joint, then the torque exerted on the failed joint should be zero, since it can only generate internal force. The model can be re-derived simply by adding constraints, which restrict the velocities of the failed joints to be zeros, to the constraint equation (2). The nature of the position failure is an increase of the number of the constraints. Thus the system will have more constrained active joints and lose degrees of freedom. Position failures won't cause unactuated task motion. But the desired DOF's may be lost. So redundant DOF's of nominal system provide the potential to tolerate position failures. A position failure can be tolerated only if the resulting system still has the desired DOF's.

3.2. Torque failure

Torque failure happens when some actuators cannot produce torques (the torques exerted by the failed actuators are zeros) due to failures in motors, electronics, etc. Note that this does not mean the corresponding active joints cannot move. Instead, they can move freely and passively. Thus the effect of the torque failure is that the corresponding active joints become passive joints.

Since some active joints become passive joints, we repartition (1) and (2) according to the new active joints and passive joints, then the new models can be easily derived. Due to the increase of passive joints, the system may not have enough constraints to keep its kinematic stability. Therefore, the number of redundant constraints of the nominal system should be at least as many as the number of failed joints. Specifically, if they are equal, then the remaining active joints are not constrained anymore.

3.3. Hard failure

Hard failure is caused by mechanical fatigue or blown-off struts. In either case, the system acts as if the failed struts are totally lost. One way of obtaining the models is to re-derive the models from the beginning for the mechanism without the lost struts.

However, we can view hard failures in another way. Let's consider the case of torque failures, in which all the active joints on a strut become passive joints. If the tip of the strut has 6 DOF's or has dependent constraints, then it forms a passive connection to the mechanism and won't enforce any independent constraint. In other words, the strut won't affect the motion of the whole system, because it moves passively with the other parts of the system. This is exactly the same situation as hard failure. In this case, the hard failure is kinematically the same as the torque failure in nature. Note that these two failures are dynamically different because in torque failure, the failed strut still plays a role in the dynamics of the whole system.

So an easy way to consider hard failures is to check DOF's of the lost strut first. If the tip of the strut does not have 6 DOF's, find out the constraints that restrict the motion of the tip, and remove these constraints from constraint (2). Then let all the active joints on the lost strut become passive joints. Repartition (1) and (2) according to the new active joints and passive joints, and the new models can be derived.

Hard failures result in an increase of passive joints and possibly a decrease of the number of constraints. This may make the system kinematically unstable. Suppose the lost struts have m_1 active joints and m_2 independent constraints that restrict the motion of the struts, and the nominal system has n redundant constraints. If $m_1 + m_2 \geq n$, then that hard failure will not cause kinematic instability. A good way of designing a system to tolerate hard failures is adding redundant struts.

4. EXAMPLES

In order to experimentally verify these theories, they are implemented on the University of Wyoming's (UW) mutually orthogonal hexapod for pointing control. The mechanical parts of the hexapod are all custom machined, based on a NASA Jet Propulsion Laboratory design. Each strut has a maximum stroke of ± 0.025 inches. The geometry of the hexapod is well designed so that the system can be decoupled by the Jacobian matrix.[10]

The nominal system has 6 DOF's. But, only rotations around x and y axes are required for the precision pointing. So there is redundancy in the Cartesain space, and this redundancy provides the potential to tolerate up to 4 position failures. If the passive joint at the top of strut 4 stucks, then it enforce 2 redundant constraints, and the system becomes an over-constrained system. From the rederived Jacobian matrix, the system now has only 4 DOF's, and 4 independent prismatic joints. However, the rotations around x and y axes are still available, so this position failure can be tolerated. Figure 2 shows the velocity ellipsoid of $\dot{\theta}_x$ and $\dot{\theta}_y$ for the nominal system and the system with failed joints. From the figure, we can see that the manipulability ellipsoid for the system with position failure is not isotropic anymore, and it has smaller manipulability for both rotations compared with the nominal system.

The UW hexapod was used to track a large spiral under 3 cases: (1) nominal case with no failures; (2) the position failure case without correction; (3) the position failure case controlled using the corrected Jacobian matrix. Figure 3 shows the result for these 3 cases. It is clear that the uncorrected system displays a degradation in performance, and the performance of the corrected system is similar to the case with no failures.

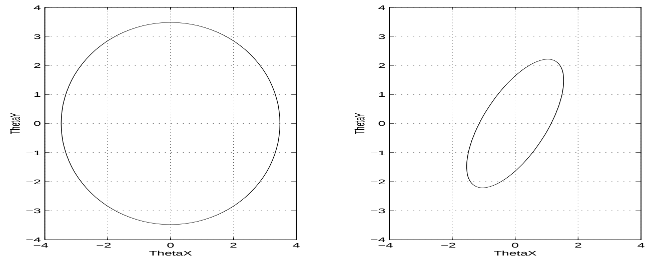

Figure 2. Manipulability ellipsoids for the nominal system (left) and the system with passive spherical joint at the top of the strut 4 stuck (right).

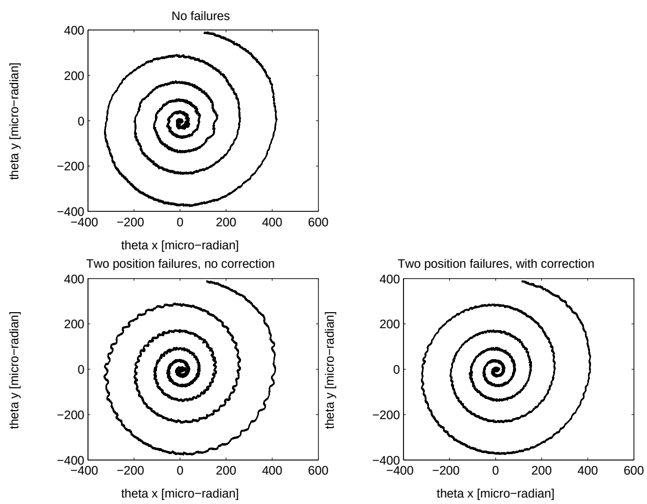

Figure 3: Tracking spirals with and without correction.

5. CONCLUSION

Over-constrained systems are especially useful for fault tolerance. Based on Ref. 8, this paper extends the kinematics and static force model of a general multibody system to over-constrained cases. The result shows that the extra constraints result in constrained active joints and redundant internal force. The ideas of how to generate $\dot{\theta}_a$ and τ that honors redundant constraints are introduced. Specifically for a Stewart platform, the relation between its forward Jacobian and its inverse Jacobian is found as $M\bar{J} = T$. General ideas of handling soft failures and hard failures are discussed.

ACKNOWLEDGEMENT

This work was supported by the Ballistic Missile Defense Organization and Army Research Office under grants DAAG55-98-1-0007 and DAAD19-00-1-0153.

REFERENCES

1. J. O'Brien and J. T. Wen, "Redundant actuation for improving kinematic manipulability," *IEEE Transactions on Robotics and Automation* **2**, pp. 45–53, March 1999.
2. J.-P. Merlet, "Redundant parallel manipulators," *Laboratory Robotics and Automation* **8**, pp. 17–24, Feb 1996.
3. M. A. Nahon and J. Angeles, "Force optimization in redundantly-actuated closed kinematic chains," in *IEEE Conference on Robotics and automation*, pp. 951–956, 1989.

4. J. E. McInroy, J. F. O'Brien, and G. W. Neat, "Precise, fault tolerant pointing using a Stewart platform," *IEEE/ASME Transactions on Mechatronics* **4**, pp. 91–95, March 1999.
5. X. Li, J. E. McInroy, and J. C. Hamann, "Optimal fault tolerant pointing of a flexure jointed hexapod," in *IEEE International Conference on Decision and Control*, pp. 3337–3338, (Sydney, Australia), December 2000.
6. Y. Chen, J. E. McInroy, and Y. Yi, "Optimal, fault tolerant velocity and static force mapping for prioritized manipulation," in *Submitted to IEEE conference on Robotics and Automation*, 2002.
7. J. Wen and L. Wilfinger, "Kinematic manipulability of general rigid multibody systems," in *IEEE Conference on Robotics and automation*, pp. 1032–1037, (Leuven, Belgium), May 1998.
8. J. Wen and L. Wilfinger, "Kinematic manipulability of general constrained rigid multibody systems," *IEEE Transactions on Robotics and Automation* **15**, pp. 558–567, June 1999.
9. K. H. Pittens and R. Podhorodeski, "A family of stewart platforms with optimal dexterity," *Journal of Robotic systems* **10**, pp. 464–479, April 1993.
10. J. E. McInroy and J. C. Hamann, "Design and control of flexure jointed hexapods," *IEEE Transactions on Robotics and Automation* **16**, pp. 372–381, August 2000.