Gravitational Stability of Frictionless Assemblies

Raju Mattikalli, David Baraff, Pradeep Khosla, *Member, IEEE,* and Bruno Repetto

***Abstract*— The stability of an assemblage of contacting rigid bodies without friction is investigated. A method is presented for finding an orientation of the assembly so that the assembly remains motionless under gravity. If no stable orientation exists for an assembly, the method finds the "least" unstable orientation. The metric used to measure stability is based on the second time-rate of change of the gravitational potential energy, and the desired orientation for an assembly is expressed in terms of an optimization problem involving changes in potential energy. The problem of finding stable or maximally-stable orientations is formulated as a constrained maximin problem. The maximin problem is shown to be a variant of standard zero-sum matrix games, and can be solved using linear programming. The method is the first general method for automatically determining stable orientations. Example assemblies are presented.**

I. Introduction

In recent years there has been much interest in automated assembly, including both planning and execution. High-level assembly plans prescribe the order in which parts come together as well as the motions that bring them together; low-level plans include control and sensing strategies to ensure completion of assembly operations. Automated *disassembly* of computer-modeled assemblies is a good method for obtaining high-level assembly plans [1]–[3]. At each stage of assembly, the stability of subassemblies is an important concern.

Consider the two subassemblies S_1 and S_2 shown in Fig. 1(a) which need to be assembled as in Fig. 1(b). The two subassemblies are manipulated by gripping components C_1 and C_4. If all contact surfaces between the components are frictionless and assembly is attempted in the orientation shown in Fig. 1(b), the assembly will be unsuccessful. Since the direction of gravity (denoted as g) is straight down in Fig. 1(b), subassembly S_2 is unstable—components C_2 and C_3 will slide free of C_1. It can be seen that the only orientation in which assembly can be performed without causing objects within S_1 and S_2 to move is the orientation shown in Fig. 1(c). In this orientation, each of the subassemblies is stable.

In this paper, we define an *assembly* as a collection of frictionless contacting rigid objects, one or more of which is assumed to be fixed in place (for example a floor, a supporting surface such as a table, or an object held by a gripper). We will assume that all objects are initially at rest. Given external forces acting on the objects, an assembly is said to be *stable* if all objects remain motionless under the influence of the external forces. Otherwise, the assembly is *unstable*. Our focus is on the stability of assemblies that are acted upon by an external uniform gravity field. We address the following problems:

Problem 1: (Stability) Given an assembly in a given orientation, is the assembly stable?

Problem 2: (Stable Orientation) Given an assembly, determine an orientation (if one exists) that makes the assembly stable under the gravity field.

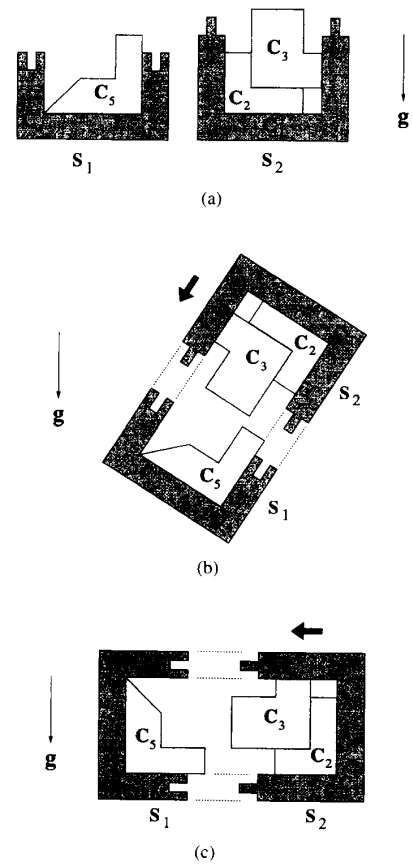


Fig. 1. (a) Subassemblies S_1 and S_2 with objects C_4 and C_1 held fixed within grippers. (b) Assembly cannot be performed in this orientation because C_2 and C_3 will move under the influence of gravity. (c) Assembly can be performed in this orientation since S_1 and S_2 are stable.

Manuscript received August 6, 1993; revised October 6, 1994. This research was supported in part by the Engineering Design Research Center, an NSF Engineering Research Center at Carnegie Mellon University.

R. Mattikalli is with the Engineering Design Research Center, Carnegie Mellon University, Pittsburgh, PA 15213–3891 USA.

D. Baraff is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213–3891 USA.

P. K. Khosla is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213–3891 USA.

B. Repetto is with Management Science Associates, Progress Automation Solutions and Services (MSA PASS), Pyromet Division.

IEEE Log Number 9410110.

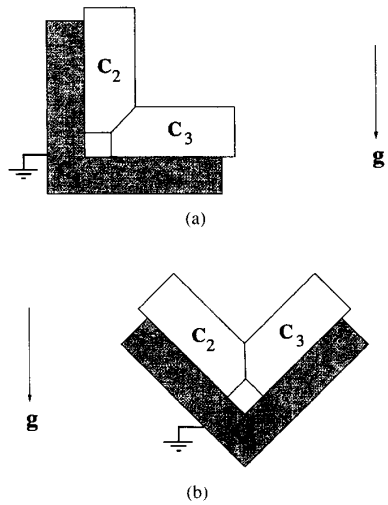


Fig. 2. (a) An assembly in a gravitationally unstable orientation. (b) The same assembly in a gravitationally stable orientation.

Problem 3: (Least Unstable Orientation) Given an assembly that is unstable in all orientations, find an orientation that makes the assembly as stable as possible.

As an example of the first two problems, consider the assembly shown in Fig. 2. Components C_2 and C_3 are identical and are placed within the right angled component C_1. Component C_1 is fixed in place as indicated in the figure by the ground symbol. The vector **g** indicates the direction of the uniform gravity field. In the orientation shown in Fig. 2(a), the assembly is unstable. Gravitational forces produce a downward translational motion on component C_2 which causes component C_3 to translate toward the right. However, the assembly can be made stable by changing its orientation (Fig. 2(b)). In this orientation, the gravitational force on components C_2 and C_3 is exactly balanced by the contact forces between the three objects, resulting in zero net force on both C_2 and C_3.

As an example of the third problem, consider the assembly in Fig. 3. Clearly, there is no orientation for which this assembly is stable. However, the orientation shown in Fig. 3(b) is clearly "less" unstable than the orientation shown in Fig. 3(a). We will defer a precise definition of how we measure the stability of an assembly until Section V, but note that using this definition Fig. 3(b) does in fact show the "least" unstable orientation for the assembly under gravity. This orientation could be used during assembly planning. In the above example, by making a change in the grasping strategy to include the unstable object C_5, the entire assembly can be made stable. This is the first method that provides a systematic way of dealing with unstable assemblies. It can be used as an aid to modify designs or to produce execution plans that stabilize the assembly.

In this paper, we will give solutions to the three problems listed above. The first problem, determining stability for a given orientation, has previously been addressed. In particular, Blum, Griffith and Neumann [4] describe a linear programming-based solution which models contact forces between contacting objects and yields a "yes/no" answer as to the stability of an assembly. The more general quadratic programming-based methods for contact forces used by the simulation systems described below can be used to determine the stability of an assembly, and the initial acceleration of parts if the assembly is unstable. However, we have not encountered any methods for determining how an assembly that is unstable can be reoriented so that it is stable.

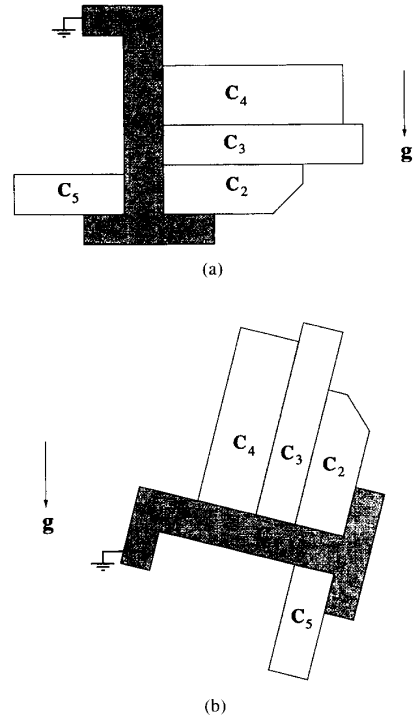


Fig. 3. (a) An assembly in a gravitationally unstable orientation. (b) Although unstable, this orientation is "less" unstable.

In this paper, we present a simple extension to Blum, Griffith and Neumann's work that enables us to find a stable orientation for an assembly (if one exists). As in Blum, Griffith and Neumann's work, this extension uses linear programming to return a "yes/no" result as to whether a stable orientation exists, and if so, what that orientation is. Unfortunately, no useful information is returned if an assembly cannot be made stable (other than the fact that no stable orientation exists). Since it is desirable to be able to minimize the instability of an assembly (even if it cannot be made completely stable), we present a potential-energy based measure for the stability of an assembly in a given orientation, and show how an orientation that is minimally unstable can be found. Remarkably, even though the formulation for finding minimally-unstable orientations is much more involved than for simply finding a stable orientation, minimally-unstable orientations can also be found using linear programming. Our method for finding such orientations finds a stable orientation if it exists and therefore

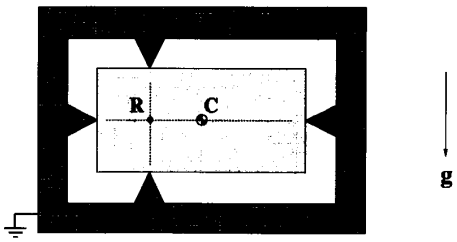


Fig. 4. By considering surface normals, the inside block possess an allowable virtual displacement about point R. However, it cannot really move.

subsumes our extension to Blum, Griffith and Neumann's work.

A. Previous Work

Our notion of stability corresponds to what Palmer [5] defined as *infinitesimal stability*. A frictionless structure is infinitesimally stable if no allowable virtual displacement of objects (that is, a displacement that is consistent with the contact constraints) decreases the potential energy of the system. Infinitesimally stability is actually a conservative approximation to actual stability. It is possible that a system may possess an allowable virtual displacement, without being able to move at all (Fig. 4). In practice, we believe that this situation is uncommon enough that characterizing stability in terms of virtual displacements is a reasonable approach.

Palmer [5] obtained computational complexity results for stability problems in which assemblies consist of rigid planar polygons. Palmer's work was mostly concerned with structures in which some surface normals were ill-defined: for example, an isolated vertex-to-vertex contact between two polyhedra. This sort of interaction adds great complexity to stability problems, and most stability results become *NP*-hard; however, when ill-defined normals do not occur, most of the problems Palmer discusses become polynomial-time problems. In particular, all of the frictionless problems and the problem of *potential* stability with friction are solvable in polynomial-time. In the absence of ill-defined normals, Palmer's result that guaranteed stability with friction is *NP*-hard does not necessarily hold; currently, the complexity of determining guaranteed stability with friction and well-defined normals is unknown. In our work, we assume that surface normals are well-defined.

One approach to determine stability is to model the interaction between parts in an assembly in terms of unknown forces. Potential stability for systems with friction can be determined by attempting to find a set of forces that produces equilibrium. Blum, Griffith and Neumann [4] implemented a potential stability test based on linear programming that searches for such a solution. The analysis begins by assuming that the assembly in question is stable. For an assembly in equilibrium, one can generate a set of force balance equations in terms of unknown interaction forces between parts and known gravitational forces. If forces that simultaneously satisfy all the force balance equations cannot be found, then the assembly is declared unstable. However, no information is obtained about the impending motion of unstable parts.

Trinkle [6] analyzes the stability of a single object, grasped by multiple frictionless fingers. He characterizes the stability of the object using both a velocity formulation and a force formulation. The velocity formulation characterizes stability by saying that the object is stable if no legal velocity of the system causes the potential energy U of the system to instantaneously decrease. The force formulation, which is the dual of the velocity formulation, is similar in spirit to Blum, Griffith, and Neumann, stating that an object is stable if there exists compressive contact forces that exactly balance gravity. Trinkle also gives a stronger criterion for stability, which he terms *first order stability*; roughly, a system is first-order stable if every nonzero legal virtual displacement of the system causes a strictly positive virtual change in the potential energy. The test for first-order stability involves examining the associated wrench matrix of a stable system to see if it has full rank. In this paper, rather than characterizing different types of stability or searching for part motions, our emphasis is on determining orientations that are stable or as nearly stable as possible. We do not attempt to find orientations which necessarily yield first-order stability (as opposed to mere stability) in this paper.

Boneschanscher *et al.* [7] tried to determine which parts are unstable (but not their impending motion) by solving for the stability of each part individually. In their work, the assembly is assumed to be sitting on a table with external forces being applied to it. Conditions on the forces between each individual part (or subassembly) and the table are written as systems of inequalities. By transforming the contact graph, a larger system of inequalities is obtained. The emerging system is solved using linear programming. If any subassembly during this process is found to be unstable, the entire assembly is declared unstable. However, as Boneschanscher *et al.* point out, their technique will not work if there are loops in the contact graph. This is a significant shortcoming since loops occur quite frequently in assemblies.

Ingleton [8] describes a formulation that yields the acceleration of frictionless rigid bodies in resting contact. At each point of contact, an unknown repulsive normal force exists. The normal forces must be strong enough to prevent interpenetration while at the same time being workless. This implies that the normal force must be zero wherever contact is broken. The normal forces can be found by solving a quadratic program. Once the normal forces have been computed, the ensuing acceleration is given by Newton's first law. Descriptions of simulation systems that use this formulation to simulate the motion of frictionless assemblies include Lötstedt [9], Featherstone [10], and Baraff [11]. Thus, both the stability of and the impending motion of any part in an assembly can be determined by examining the accelerations of the parts in these simulation systems. This method differs from Boneschanscher *et al.* in that it can be used to solve the infinitesimal instability problem for frictionless systems with loops.

B. Approach Outline

Our method to solve the stability problems listed above follows similar lines to Baraff [11]. However, instead of

taking a vectorial approach by solving for forces, we adopt a Lagrangian approach and solve for part motions. Our method is centered on a formulation for the virtual change in the gravitational potential energy U of an assembly, subject to a given virtual displacement. In particular, we consider the virtual change δU in energy corresponding to a *legal* virtual displacement of the assembly, (that is, a displacement which does not violate the contact constraints). Let $\delta \mathbf{r}_i$ denote a virtual translation of the ith part and let δU denote the virtual change in U caused by this displacement. If the ith part has mass M_i and $\mathbf{g}$ denotes the gravity vector, then

$$\delta U = -\sum_{i=1}^{n} M_i \mathbf{g} \cdot \delta \mathbf{r}_i \tag{1}$$

for an assembly with n parts.

If the assembly is initially at rest, the kinetic energy T of the assembly is zero. Since we are not considering friction forces, our system is conservative so that the total energy $U + T$ of the system remains constant. If the system moves, then T must increase, which means that U must *decrease*. Thus, if there is no legal displacement for which δU is negative (meaning that U decreases), the system cannot undergo any motion and must be stable. Conversely, if there exists a virtual displacement which yields a negative value for δU, then the system is unstable and is guaranteed to begin moving. In particular then, the stability of a system can be characterized in terms of the legal virtual displacements that cause the largest decrease in U, or equivalently, that minimize the corresponding energy change δU. This is essentially the same as Trinkle's [6] velocity formulation (with virtual displacement in place of velocity).

This method of analysis offers no advances in determining stability compared with previous methods; unlike previous methods, however, we are not limited to simply determining the stability of an assembly in a particular orientation. An assembly that is unstable in one orientation might be made stable by a change of orientation; that is, by a rotation of all the objects about a common point. However, rather than viewing a change in orientation as a rotation of the objects in our frame of reference (but with no change in the gravity direction), it will be simpler to consider changing only the gravity direction $\mathbf{g}$ instead while leaving the objects alone. Thus, when we say we are searching for a stable orientation for an assembly in some particular configuration, we mean that we are trying to find a gravity direction $\mathbf{g}$ for which the assembly is stable in that same configuration. With this viewpoint, we can extend our potential energy analysis to consider $\mathbf{g}$ as a variable rather than a constant. We know that if every motion that does not cause interpenetration between objects is "uphill" (that is, causes U to *increase*), then the configuration is stable. Accordingly, we can use our potential energy formulation to search for a value for $\mathbf{g}$ that makes all allowable motions "uphill," resulting in a stable orientation. Failing that, we can try to choose a value for $\mathbf{g}$ that minimizes the steepness of the most "downhill" allowable motion.

In earlier work [12], a measure of the stability of an assembly in a given orientation was defined in terms of the most negative δU possible over a set of bounded legal displacements. The optimal direction for $\mathbf{g}$ was then found by solving a constrained maximin problem. The maximin problem was solved by numerical iteration. Each step of the iteration involved solving a linear program. In this paper, we will show that a solution to this maximin problem can be found by solving a single linear program, eliminating the need for an iterative solution method. The insight into this reduction lies in viewing the maximin problem as a variant of two-person zero-sum matrix games. Two-person zero-sum matrix games were first shown to be equivalent to linear programs by Dantzig. Additionally, we refine our measure of stability so that it is both better physically motivated and coordinate-system invariant.

The outline of the paper is as follows. In Section II, we describe the contact constraints for the system and characterize the set of legal displacements. Section III details two linear programs that can be used to determine the stability of an assembly in a given orientation. The first linear program is based on potential energy considerations, while the second linear program follows the method used by Blum, Griffith and Neumann [4]. In Section IV, we show how this second linear program is easily modified to find a stable orientation (if it exists). Section V describes the metric used to measure stability and relates this metric to the second rate of change $\ddot{U}$ of the potential energy. Based on this metric, the potential-energy stability formulation is extended to characterize the "least" unstable orientation in terms of a linearly constrained maximin problem. The solution of this maximin problem using a single linear program is presented in Section VI. Results and discussions are presented in Section VII.

II. Motion Constraints

We will represent possible motions of a system of rigid bodies in terms of virtual displacements of each body. Let $\delta \mathbf{p}_i = (\delta \mathbf{r}_i, \delta \boldsymbol{\theta}_i)$ represent a displacement of the ith body in the system, with $\delta \mathbf{r}_i$ and $\delta \boldsymbol{\theta}_i$ vectors in $\boldsymbol{R}^3$. The vector $\delta \mathbf{r}_i$ denotes a translational displacement of the ith body, while $\delta \boldsymbol{\theta}_i$ denotes a rotation of magnitude $||\delta \boldsymbol{\theta}_i||$ of the body around its center of mass. The axis of the rotation is along the $\delta \boldsymbol{\theta}_i$ direction.

Contact between bodies generates constraints on the allowable displacements. Consider Fig. 5 where bodies A and B contact. If body A undergoes a displacement $\delta \mathbf{p}_a = (\delta \mathbf{r}_a, \delta \boldsymbol{\theta}_a)$, then point $\mathbf{d}$, as attached to body A, undergoes a particular displacement $\delta \mathbf{d}_a$. Similarly, a displacement $\delta \mathbf{p}_b$ of body B causes a displacement $\delta \mathbf{d}_b$ of point $\mathbf{d}$, as attached to body B. To prevent interpenetration from occurring, the relative displacement $\delta \mathbf{d}_a - \delta \mathbf{d}_b$ cannot have any component opposite the unit normal direction $\hat{\mathbf{n}}$. We can express this as the constraint

$$\hat{\mathbf{n}} \cdot (\delta \mathbf{d}_a - \delta \mathbf{d}_b) \geq 0. \tag{2}$$

Similarly, we also need to prevent interpenetration from occurring at point $\mathbf{d}'$ by requiring that $\hat{\mathbf{n}} \cdot (\delta \mathbf{d}'_a - \delta \mathbf{d}'_b) \geq 0$. If body B was fixed, the motion constraint at $\mathbf{d}$ would simply be

$$\hat{\mathbf{n}} \cdot \delta \mathbf{d}_a \geq 0 \tag{3}$$

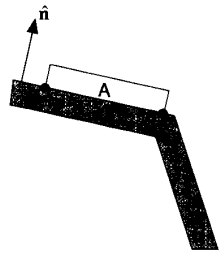


Fig. 5. Contact between bodies A and B. Motion constraints are formulated in terms of the relative motion of the bodies at points $\mathbf{d}$ and $\mathbf{d}'$.

and similarly for $\mathbf{d}'$. To simplify bookkeeping, we do not count fixed objects as bodies in our system; rather, we simply note when regular movable objects are in contact with fixed objects, and generate the appropriate motion constraint, such as (3).

We will assume that the motion constraints can be expressed by a finite number of constraint inequalities in the form of (2) or (3), all of which must be satisfied. (Palmer [5] and Baraff [13] contain further discussion on this issue.) That is, we consider systems whose motion constraints are expressed in terms of m contact points between the bodies.

The virtual displacement of any point on any object is a linear function of the appropriate $\delta\mathbf{r}$ and $\delta\boldsymbol{\theta}$ variables; thus each contact constraint can be written as a linear inequality involving some of the $\delta\mathbf{r}$ and $\delta\boldsymbol{\theta}$ variables. We can express the simultaneous satisfaction of all the contact constraints as one large linear system. If the vector $\delta\mathbf{p}$ denotes the virtual displacements of the n bodies by writing

$$\delta\mathbf{p} = \begin{pmatrix} \delta\mathbf{r}_1 \\ \delta\boldsymbol{\theta}_1 \\ \vdots \\ \delta\mathbf{r}_n \\ \delta\boldsymbol{\theta}_n \end{pmatrix}, \tag{4}$$

all m motion constraints can be written in the form

$$\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \tag{5}$$

where $\mathbf{J}$ is an $m \times 6n$ matrix, and $\mathbf{0}$ is an m-vector of zeroes. (We will denote row vectors, column vectors, and matrices whose entries are all zero simply by $\mathbf{0}$ throughout this paper. The dimension of $\mathbf{0}$ should be clear from the context in which it occurs.)

Using this notation, we can say that a legal motion for an assembly is a displacement $\delta\mathbf{p}$ that satisfies $\mathbf{J}\delta\mathbf{p} \geq \mathbf{0}$. Note that the displacement $\delta\mathbf{p} = \mathbf{0}$ always yields a legal motion (the null-motion). Also, if $\delta\mathbf{p} \neq \mathbf{0}$ is a legal motion and α is a nonnegative scalar, then $\alpha\delta\mathbf{p}$ is also a legal motion.

III. Determining Stability

Determining if an assembly is stable under a particular direction of gravity is fairly straightforward. In this subsection, we describe two different methods for determining if an assembly is stable. The first method is based on potential energy considerations, while the second method considers the contact forces that arise at contact points. Both methods involve linear programming. The latter method was used in work by Blum, Griffith and Neumann [4], and is not limited to frictionless assemblies (although in assemblies with friction, the method determines only potential stability, not guaranteed stability). In Section IV and V, we show how both of these methods for determining stability can be modified to find a stable orientation for the assembly (if it exists).

A. Potential Energy

Suppose that an assembly undergoes a virtual displacement $\delta\mathbf{p}$. The change in potential energy δU corresponding to the displacements $\delta\mathbf{p}_i = (\delta\mathbf{r}_i, \delta\boldsymbol{\theta}_i)$ of the n bodies is

$$\delta U = -\sum_{i=1}^{n} M_i \mathbf{g} \cdot \delta\mathbf{r}_i \tag{6}$$

where M_i is the mass of the ith body. If we define the matrix $\mathbf{M}$ as the following $3 \times 6n$ matrix,

$$\begin{pmatrix} M_1 & 0 & 0 & 0 & 0 & 0 & \dots & M_n & 0 & 0 & 0 & 0 & 0 \\ 0 & M_1 & 0 & 0 & 0 & 0 & \dots & 0 & M_n & 0 & 0 & 0 & 0 \\ 0 & 0 & M_1 & 0 & 0 & 0 & \dots & 0 & 0 & M_n & 0 & 0 & 0 \end{pmatrix} \tag{7}$$

then we can write

$$\sum_{i=1}^{n} M_i \delta\mathbf{r}_i = \mathbf{M}\delta\mathbf{p} \tag{8}$$

and thus

$$\delta U = -\mathbf{g} \cdot \mathbf{M}\delta\mathbf{p} = -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}. \tag{9}$$

As stated in the introduction, if for a given gravity direction $\mathbf{g}$ all legal motions yield $\delta U \geq 0$, then the assembly is stable. We are therefore interested in knowing the minimum value that δU can assume over all legal motions. If we let $\bar{z}$ denote this minimum, by writing

$$\bar{z} = \min_{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p} \tag{10}$$

then the structure is stable if $\bar{z} \geq 0$. Note however that the null-motion $\delta\mathbf{p} = \mathbf{0}$ is always legal, and yields $\delta U = 0$. Thus, $\bar{z}$ is bounded above by zero, and we can say simply that the structure is stable if $\bar{z} = 0$.

The value $\bar{z}$ can be determined by linear programming. However, as it stands, if $\bar{z} \neq 0$, then there must exist a legal $\delta\mathbf{p}$ for which $\delta U < 0$. In this case, the minimum value of δU is $-\infty$, since a displacement of $\alpha\delta\mathbf{p}$ is legal and yields an energy change of $\alpha\delta U < 0$ for any $\alpha > 0$. This is a consequence of the constraint $\mathbf{J}\delta\mathbf{p} \geq \mathbf{0}$, which constrains the motion direction $\delta\mathbf{p}$, but not its magnitude.

Anticipating future development, it is useful to bound the magnitude of the displacements $\delta\mathbf{p}$ considered in (10). Since we would like to be able to use linear programming techniques, we would like to bound $\delta\mathbf{p}$'s magnitude with linear constraints. We can do this straightforwardly by redefining $\bar{z}$ as

$$\bar{z} = \min_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\delta\mathbf{p}\|_\infty \leq 1}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}. \tag{11}$$

The infinity norm $\|\mathbf{v}\|_\infty$ of a vector $\mathbf{v}$ is the maximum absolute value over all the components of $\mathbf{v}$. The condition $\|\delta\mathbf{p}\|_\infty \leq 1$ constrains all components of $\delta\mathbf{p}$ to a have magnitude of

at most one. Equation (11) can then be solved by linear programming. If the solution $\bar{z}$ is zero, then the assembly is stable. Otherwise, $\bar{z}$ is a (finite) negative value, and the assembly is unstable. (Note however, that the displacement $\delta\mathbf{p}$ which yields the minimal $\bar{z} = \delta U$ does not indicate the impending motion direction of the assembly, although in simple cases it may be a good approximation. To exactly determine the impending motion direction it is necessary to solve a quadratic programming problem [8].) The application of the above linear program to some test assemblies is shown in Figs. 6–7 in Section VII.

B. Contact Forces

Instead of taking an analytical approach and trying to solve for motion directions which decrease potential energy, we can use the vectorial approach. Let us consider the contact force that arises at each of the m contact points of the assembly. Since we are dealing with frictionless contacts, we know that the contact forces will act along normals to the contact surfaces. Thus, at the ith contact point, we consider a contact force $f_i\hat{\mathbf{n}}_i$ that acts on body A of the contact, and a contact force $-f_i\hat{\mathbf{n}}_i$ that acts on body B of the contact, with f_i the unknown scalar magnitude of the force. Since $\hat{\mathbf{n}}_i$ is directed from B toward A, and since contact forces must be repulsive, the magnitude f_i must be nonnegative; that is, $f_i \geq 0$.

Let the vector of contact force magnitudes f_i be denoted by $\boldsymbol{f}$. The net force $\mathbf{F}_j \in \mathbf{R}^3$ acting on the jth body of the assembly can be written as

$$\mathbf{F}_j = \sum_{i=1}^{m} s_{ji} f_i \hat{\mathbf{n}}_i + M_j \mathbf{g} \tag{12}$$

where s_{ji} is either 1, -1, or zero. If the jth body is not involved in the ith contact, then s_{ji} is zero. If the contact force exerted on the jth body from the ith contact point is $f_i\hat{\mathbf{n}}_i$, then s_{ji} is 1. Otherwise, the contact force acting on the jth body is $-f_i\hat{\mathbf{n}}_i$, and s_{ji} is -1.

The net torque $\boldsymbol{\tau}_j \in \mathbf{R}^3$ acting on the jth body of the assembly about its center of mass is similarly written as

$$\boldsymbol{\tau}_j = \sum_{i=1}^{m} s_{ji}(\mathbf{d}_i - \mathbf{c}_j) \times f_i \hat{\mathbf{n}}_i \tag{13}$$

where $\mathbf{d}_i$ is the location of the ith contact point, and $\mathbf{c}_j$ is the location of the center of mass of the jth body. The scalars s_{ji} are the same as in the previous equation. The $\boldsymbol{\tau}_j$ are independent of $\mathbf{g}$ since a uniform gravity field does not exert a torque. If we define the $6n$-vectors $\mathbf{Q}$ and $\mathbf{G}$ as the collections

$$\mathbf{Q} = \begin{pmatrix} \mathbf{F}_1 \\ \boldsymbol{\tau}_1 \\ \vdots \\ \mathbf{F}_n \\ \boldsymbol{\tau}_n \end{pmatrix} \quad \text{and} \quad \mathbf{G} = \begin{pmatrix} M_1\mathbf{g} \\ \mathbf{0} \\ \vdots \\ M_n\mathbf{g} \\ \mathbf{0} \end{pmatrix}, \tag{14}$$

we can write

$$\mathbf{Q} = \mathbf{A}\boldsymbol{f} + \mathbf{G} \tag{15}$$

where $\mathbf{A}$ is a $6n \times m$ matrix whose coefficients are given by (12, 13).

Because the assembly is frictionless, its impending motion is completely determined [8]. If there exist repulsive contact forces such that the net force and torque on every body is zero, then such forces will arise at the contact points, and the assembly is stable and will not move. Thus, we can determine stability by simply checking whether there exists $\boldsymbol{f}$ such that

$$\mathbf{Q} = \mathbf{A}\boldsymbol{f} + \mathbf{G} = \mathbf{0} \quad \text{and} \quad \boldsymbol{f} \geq \mathbf{0}. \tag{16}$$

The existence of a suitable $\boldsymbol{f}$ can be determined by linear programming.

IV. Finding a Stable Orientation

Suppose that for an assembly and a given gravity direction we find that the structure is unstable by using either of the two methods described in the previous section. Can we find a new orientation which will make the structure stable? In this section, we show how the contact-force formulation to determine stability can be trivially modified to find a stable orientation if one exists. However, if no such orientation exists, the contact-force formulation in this section yields no information other than the fact that the assembly cannot be made stable. This is in contrast to the potential-energy method described in Section V which finds either a stable orientation or the least unstable orientation possible. Although the potential energy-method gives a more comprehensive result than the contact-force formulation, the latter method is worth developing because of its extremely simple presentation.

In Section III-B, we were searching for a vector $\boldsymbol{f}$ so that

$$\mathbf{Q} = \mathbf{A}\boldsymbol{f} + \mathbf{G} = \mathbf{0} \quad \text{and} \quad \boldsymbol{f} \geq \mathbf{0} \tag{17}$$

where $\mathbf{G}$ depended upon the known value of $\mathbf{g}$. If however we treat $\mathbf{g}$ as an additional unknown, all we need to do is simply check if there exist values for $\boldsymbol{f}$ *and* $\mathbf{g}$ that satisfy $\mathbf{Q} = \mathbf{0}$, indicating that the assembly is stable. That is, if we can find vectors $\mathbf{g}$ and $\boldsymbol{f}$ such that

$$\mathbf{Q} = \mathbf{A}\boldsymbol{f} + \mathbf{G} = \mathbf{0}, \quad \boldsymbol{f} \geq \mathbf{0} \quad \text{and} \quad \|\mathbf{g}\|_2 = 1 \tag{18}$$

(where $\mathbf{G}$ is defined in terms of $\mathbf{g}$ by (14)) then the assembly is stable in orientation $\mathbf{g}$. If (18) has a solution, we can find it by linear programming, although a slight modification is required.

The constraint that $\mathbf{g}$ be a unit vector (that is, $\|\mathbf{g}\|_2 = 1$) cannot be enforced in a linear program. However, it is not necessary to search among only unit vectors; it is merely necessary to make sure that we search among all possible directions. We can do this by considering vectors $\mathbf{g}$ such that $\|\mathbf{g}\|_1 = 1$. (For a vector $\mathbf{v}$, $\|\mathbf{v}\|_1 = \sum_i |v_i|$.) Geometrically, we are changing the search domain of $\mathbf{g}$ from the surface of a unit sphere (that is, $\|\mathbf{g}\|_2 = 1$) to the surface of a "unit diamond" ($\|\mathbf{g}\|_1 = 1$) consisting of eight planar facets, each of which is a convex linear domain. We can search for a stable orientation by considering each planar facet in turn.

Let the set S_1 be defined as the set of vectors $\mathbf{g} = (g_x, g_y, g_z)^T$ satisfying

$$g_x, g_y, g_z \geq 0 \quad \text{and} \quad g_x + g_y + g_z = 1. \tag{19}$$

Similarly, define S_2 to consist of vectors satisfying

$$-g_x, g_y, g_z \geq 0 \quad \text{and} \quad -g_x + g_y + g_z = 1 \tag{20}$$

and so on through all the eight sign permutations of g_x, g_y and g_z. Using this notation, we can look for a $\mathbf{g}$ that makes the assembly stable by determining if any of the eight linear programs

$$\mathbf{Q} = \mathbf{A}\boldsymbol{f} + \mathbf{G} = \mathbf{0}, \quad \boldsymbol{f} \geq \mathbf{0}, \qquad \text{and} \qquad \mathbf{g} \in S_i (1 \leq i \leq 8) \tag{21}$$

have a solution $\boldsymbol{f}$ and $\mathbf{g}$.[1] If the assembly can be made stable, we will find a value of $\mathbf{g}$ which makes it stable. When no such $\mathbf{g}$ exists, the only information furnished by this formulation is the fact that all of the linear programs are unsatisfiable.

V. Finding the Least Unstable Orientation

Modifying the contact-force formulation to find a stable orientation (if it exists) was straightforward. In working with the potential energy formulation though, we are not limited to simply finding a stable orientation, or reporting that the assembly is unstable. Instead, we can modify the potential energy formulation so that we can find either a stable orientation, or, for unstable assemblies, the least unstable orientation possible. In assembly planning, an unstable assembly can be made stable by placing external support elements such as fixture components or robot fingers at certain locations on unstable objects. Thus in situations where no stable orientation for an assembly can be found, it is desirable to find an orientation wherein the assembly is closest to being stable.

A. Measuring Stability

In an earlier paper [12], we measured stability as follows. Given a fixed direction for $\mathbf{g}$, we characterized stability by the quantity

$$\min_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\delta\mathbf{p}\|_2 \leq 1}} \delta U = \min_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\delta\mathbf{p}\|_2 \leq 1}} -\mathbf{g}^T \mathbf{M} \delta\mathbf{p} \tag{22}$$

(although we used the linearized constraint $\|\delta\mathbf{p}\|_\infty \leq 1$ in actual computations.) The intuition behind this characterization is that physical systems tend to move "downhill" as quickly as possible. The quicker an assembly can lose potential energy, the more unstable the assembly is. By examining all unit legal displacements for a fixed gravity direction $\mathbf{g}$, our idea was to characterize the stability of a system in terms of the most negative potential energy change δU. For stable systems, we would obtain $\min \delta U = 0$, while for unstable systems we would find $\min \delta U < 0$.

Although the intuition behind this measure is valid, there is a subtle flaw in its definition as given by (22). The problem is that (22) is not coordinate-system invariant. To see this, consider measuring the displacement of a planar rigid body, with degrees of freedom x, y, and θ. Imagine that there are two observers, and that each observer measures θ in radians, but one observer measures x and y in meters while the other measures x and y in centimeters. As a result, the displacement sets $\{(x, y, \theta) \mid x^2 + y^2 + \theta^2 \leq 1\}$ considered by the two observers are different, and the observers may not be able to agree which displacement it is that yields the minimum δU, nor what the minimum δU is.

The problem is that the constraint $\|\delta\mathbf{p}\|_2 \leq 1$is not physically motivated. Baraff and Mattikalli [15] show that the stability of a system can be measured in a coordinate-system invariant way as follows. Let $\mathbf{N}$ be the symmetric positive definite $6n \times 6n$ matrix defined as

$$\mathbf{N} = \begin{pmatrix} M_1\mathbf{1} & \mathbf{0} & \cdots & & \\ \mathbf{0} & \mathbf{I}_1 & & & \\ \vdots & & \ddots & & \vdots \\ & & & M_n\mathbf{1} & \mathbf{0} \\ & & \cdots & \mathbf{0} & \mathbf{I}_n \end{pmatrix} \tag{23}$$

where $\mathbf{1}$ denotes the 3×3 identity matrix, and M_i and $\mathbf{I}_i \in \boldsymbol{R}^{3x3}$ are the mass and inertia tensor of the ith body. For a vector $\mathbf{v} \in \boldsymbol{R}^{6n}$, the "$\|\cdot\|_\mathbf{N}$-norm" of a vector $\mathbf{v}$ is defined as[2]

$$\|\mathbf{v}\|_\mathbf{N} = \sqrt{\mathbf{v}^T \mathbf{N} \mathbf{v}}. \tag{24}$$

Using the $\|\cdot\|_\mathbf{N}$-norm in place of the $\|\cdot\|_2$-norm in (22) produces a sensible, coordinate-invariant characterization of the stability of an assembly. In Baraff and Mattikalli [15], the maximum "steepness" S of potential-energy descent for an (initially) motionless system with contact constraints is defined as

$$S = \min_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\delta\mathbf{p}\|_\mathbf{N} \leq 1}} -\mathbf{g}^T \mathbf{M} \delta\mathbf{p}. \tag{25}$$

It is then shown that if an assembly is unstable, the displacement $\delta\mathbf{p}$ which minimizes (25) is unique, and gives the direction (though not the magnitude) of the initial acceleration of all the objects in the assembly. In addition, the maximum steepness S is related to the second time-derivative of potential energy U in a simple way: for any motionless assembly, appendix A shows that $\ddot{U} = -S^2$. (Note that these characterizations are only to be applied to an assembly that is initially motionless, and do not necessarily apply once the assembly has begun to move.) Thus, (25) yields a physically-sensible (and thus coordinate-system invariant) measure of stability.

It could be argued that, for assembly planning, an orientation which minimizes the number of unstable parts may be a better metric than the one we propose (i.e. minimizing $-\ddot{U}$, the second time-derivative of change in potential energy). Although this may be true in certain situations, it is not always the case. Consider Fig. 5 which shows an assembly in two different orientations, in both of which the object C_1 is fixed. The orientation in Fig. 5(a) minimizes the number of unstable parts, while that in Fig. 5(b) minimizes $-\ddot{U}$. A single contact finger placed at any point along the smaller edge of object C_6 would suffice to stabilize the assembly in Fig. 5(b). However, to stabilize the orientation in Fig. 5(a)

[1] We could also partition gravity by considering the unit cube of directions $\|\mathbf{g}\|_\infty = 1$. The natural division here would be to use sets S_1 through S_6, with S_1 defined by $g_x = 1$ and $-1 \leq g_y, g_z \leq 1$ and S_2 by $g_x = -1$ and $-1 \leq g_y, g_z \leq 1$, and similarly for S_3 through S_6. This would work just as well as considering the set $\|\mathbf{g}\|_1 = 1$. The only reason for using the metric $\|\mathbf{g}\|_1 = 1$ in this section is that we are forced to use this metric in subsequent sections.

[2] Using this notation, if $\mathbf{v}$ is the velocity of the system, the kinetic energy T of the system is $1/2\|\mathbf{v}\|_\mathbf{N}^2$. Essentially, we are placing a metric on our displacement space that is defined in terms of kinetic energy.

two fingers are required—one each on the left end of the objects C_2 and C_3. A good metric for instability that would be useful for assembly planning would be one which minimizes the "effort" for grasping/fixturing. This "effort" would depend on a number of factors such as accessibility of object surfaces, number of grasping/fixturing points, force magnitudes at grasping/fixturing points, relative location of the points, design of the grasping/fixturing agent among other things. By minimizing $-\bar{U}$, our metric is minimizing the force required to keep the unstable parts in place.

Considering the above, we will base our metric of stability on (25). For practical use though, we will need to linearize the constraint $\|\delta\mathbf{p}\|_{\mathbf{N}} \leq 1$. (Although this will reintroduce some variance with respect to the coordinate system, we will be left with a computational method that is still much better than using our original constraint $\|\delta\mathbf{p}\|_\infty \leq 1$.) Let $\mathbf{N}$ be expressed in the form

$$\mathbf{N} = \mathbf{K}^T\mathbf{K} \tag{26}$$

where $\mathbf{K}$ is a $6n \times 6n$ matrix. (The matrix $\mathbf{K}$ is easily computed using Cholesky decomposition, the more so since one need only decompose the 3×3 symmetric inertia tensors $\mathbf{I}_i$ that form $\mathbf{N}$ to compute $\mathbf{K}$.) From the definition of the $\|\cdot\|_{\mathbf{N}}$-norm, for any vector $\mathbf{v}$ we have

$$\begin{aligned}\|\mathbf{v}\|_{\mathbf{N}} &= \sqrt{\mathbf{v}^T\mathbf{N}\mathbf{v}}\\ &= \sqrt{\mathbf{v}^T\mathbf{K}^T\mathbf{K}\mathbf{v}}\\ &= \sqrt{(\mathbf{K}\mathbf{v})^T(\mathbf{K}\mathbf{v})}\\ &= \|\mathbf{K}\mathbf{v}\|_2.\end{aligned} \tag{27}$$

We then linearize by replacing the bound

$$\|\delta\mathbf{p}\|_{\mathbf{N}} = \|\mathbf{K}\delta\mathbf{p}\|_2 \leq 1 \tag{28}$$

with

$$\|\mathbf{K}\delta\mathbf{p}\|_\infty \leq 1. \tag{29}$$

B. The Maximin Formulation

Let us recast $\bar{z}$ as a function of $\mathbf{g}$ by writing

$$\bar{z}(\mathbf{g}) = \min_{\substack{\mathbf{J}\delta\mathbf{p}\geq\mathbf{0}\\ \|\mathbf{K}\delta\mathbf{p}\|_\infty\leq 1}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}. \tag{30}$$

We measure the stability of an assembly with gravity direction $\mathbf{g}$ using the function $\bar{z}(\mathbf{g})$. If $\mathbf{g}$ is an orientation for which the assembly is stable, then $\bar{z}(\mathbf{g}) = 0$. Otherwise, the assembly is unstable, and $\bar{z}(\mathbf{g}) < 0$. In trying to find the "least" unstable orientation, we seek to maximize $\bar{z}(\mathbf{g})$. In the remainder of this paper, we will restrict $\mathbf{g}$ to lie in S_1, by writing $\mathbf{g} \geq \mathbf{0}$ and $\|\mathbf{g}\|_1 = \sum_{i=1}^3 g_i = 1$.[3] In searching for the least unstable assembly, we will have to perform eight different searches; one for each partition S_i. All statements and methods made hereafter involving $\mathbf{g} \in S_1$ can be applied to the other seven cases of $\mathbf{g} \in S_i$.

To find the least unstable orientation, we are trying to solve a *maximin* problem. That is, we are trying to solve

$$\max_{\substack{\mathbf{g}\geq\mathbf{0}\\ \sum g_i=1}} \bar{z}(\mathbf{g}) = \max_{\substack{\mathbf{g}\geq\mathbf{0}\\ \sum g_i=1}} \left(\min_{\substack{\mathbf{J}\delta\mathbf{p}\geq\mathbf{0}\\ \|\mathbf{K}\delta\mathbf{p}\|_\infty\leq 1}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right). \tag{31}$$

Constrained maximin problems are in general hard to solve. However, (31) has a form similar to a maximin problem that is solvable by linear programming. Given an $m \times n$ matrix $\mathbf{A}$, the minimax theorem of matrix games, first proved by Von Neumann, states that

$$\max_{\substack{\mathbf{x}\geq\mathbf{0}\\ \sum x_i=1}} \left(\min_{\substack{\mathbf{y}\geq\mathbf{0}\\ \sum y_i=1}} \mathbf{y}^T\mathbf{A}\mathbf{x}\right) = \min_{\substack{\mathbf{y}\geq\mathbf{0}\\ \sum y_i=1}} \left(\max_{\substack{\mathbf{x}\geq\mathbf{0}\\ \sum x_i=1}} \mathbf{y}^T\mathbf{A}\mathbf{x}\right). \tag{32}$$

where $\mathbf{x}$ and $\mathbf{y}$ are vectors in $\boldsymbol{R}^n$ and $\boldsymbol{R}^m$, respectively. Furthermore, the value of $\mathbf{x}$ for which the maximum on the left is attained can be found by solving a linear program; the solution of the dual linear program gives the value of $\mathbf{y}$ for which the minimum on the right is attained.

The maximin problem (31) is similar to (32) in that the constraints on $\mathbf{g}$ are $\mathbf{g} \geq \mathbf{0}$ and $\sum g_i = 1$. However, the constraints on the inner variables $\delta\mathbf{p}$ have quite a different form. It turns out that a variant of the linear program used to solve the maximization in (32) for $\mathbf{x}$ can be used to solve (31) for $\mathbf{g}$. In Section VI we will exhibit a pair of dual linear programs which find a solution $\mathbf{g}$ to (31) and a solution $\delta\mathbf{p}$ to the dual problem

$$\min_{\substack{\mathbf{J}\delta\mathbf{p}\geq\mathbf{0}\\ \|\mathbf{K}\delta\mathbf{p}\|_\infty\leq 1}} \left(\max_{\substack{\mathbf{g}\geq\mathbf{0}\\ \sum g_i=1}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right). \tag{33}$$

It will also turn out that

$$\begin{aligned}\max_{\substack{\mathbf{g}\geq\mathbf{0}\\ \sum g_i=1}} \bar{z}(\mathbf{g}) &= \max_{\substack{\mathbf{g}\geq\mathbf{0}\\ \sum g_i=1}} \left(\min_{\substack{\mathbf{J}\delta\mathbf{p}\geq\mathbf{0}\\ \|\mathbf{K}\delta\mathbf{p}\|_\infty\leq 1}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right)\\ &= \min_{\substack{\mathbf{J}\delta\mathbf{p}\geq\mathbf{0}\\ \|\mathbf{K}\delta\mathbf{p}\|_\infty\leq 1}} \left(\max_{\substack{\mathbf{g}\geq\mathbf{0}\\ \sum g_i=1}} -\mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right).\end{aligned} \tag{34}$$

For now however, it will be more instructive to simply hope that (34) holds, and use physical intuition to formulate a linear program that solves (33) for $\delta\mathbf{p}$. Assuming that (34) holds, the solution of the dual of this linear program will yield a vector $\mathbf{g}$ which maximizes (31). The intuition which allows us to directly formulate the necessary linear program lies in viewing (33) as a competition, or game (just as (32) is viewed as what is known as a "two-person zero-sum game").

[3] Note that the linearization of $\|\mathbf{g}\|_2 = 1$ to $\|\mathbf{g}\|_1 = 1$ further distorts the original stability measure of (25). We feel that the distortion caused by the linearizations of (25) is at most of minor concern in typical manufacturing and design applications. Note that if the least unstable orientation (as measured without the linearizations) yields $\bar{z}(\mathbf{g}) = 0$, then even with the linearizations, $\bar{z}(\mathbf{g})$ is still zero. Thus, if a completely stable orientation exists, it will not be missed because of the linearizations.

C. A Particle versus Gravity

For illustrative purposes, let us greatly simplify the problem. Our assembly, for the moment, consists of a single particle in $\mathbf{R}^2$, with degrees of freedom δr_x and δr_y and unit mass. Our gravity vector likewise has two components g_x and g_y. To begin with, we will assume that there are no constraints on δr_x or δr_y, except for the bounds $|\delta r_x| \leq 1$ and $|\delta r_y| \leq 1$. We will search for the least unstable orientation by finding the solution of

$$\min_{|\delta r_x|,|\delta r_y| \leq 1} \left(\max_{\substack{g_x, g_y \geq 0 \\ g_x + g_y = 1}} -L \right) \tag{35}$$

where

$$L = (g_x g_y) \begin{pmatrix} \delta r_x \\ \delta r_y \end{pmatrix} = g_x \delta r_x + g_y \delta r_y. \tag{36}$$

Let us rewrite this by removing the minus sign; this swaps the "min" and "max" functions, yielding

$$\max_{|\delta r_x|,|\delta r_y| \leq 1} \left(\min_{\substack{g_x, g_y \geq 0 \\ g_x + g_y = 1}} L \right) \tag{37}$$

The quantities associated with the "min" are the components of a force, while those associated with the "max" are those of motion. Let us consider (37) from a game-theoretic point of view: the "gravity" player wants to prevent the "particle" player from moving (which minimizes L), while the particle player wants to move in the direction of gravity (thus maximizing L) as much as possible. Since moving against gravity (i.e. uphill) results in a negative value for L, while standing still results in L being zero, the particle player would rather stay still than move uphill even slightly. Conversely, the gravity player wishes to pick a gravity direction so that the particle will move as little as possible.

Suppose that the game is played without any constraints on the particle's motion. By not moving at all (that is, by choosing $\delta r_x = \delta r_y = 0$) a value of $L = 0$ is attained. However, since neither g_x nor g_y can be negative, the particle player can obtain a larger value for L by choosing $\delta r_x = \delta r_y = 1$: no matter what direction is chosen for gravity, $L = 1$, and the particle player wins the game.

If however the particle is in contact with an obstacle and must satisfy the motion constraint

$$\delta r_x + \delta r_y \leq 0 \tag{38}$$

along with the regular bounds $|\delta r_x| \leq 1$ and $|\delta r_y| \leq 1$ (as in Fig. 7), the game is quite different.

In this case, the gravity player chooses $g_x = g_y = 1/2$, since that is exactly the direction the particle *cannot* move in. This guarantees that $L \leq 0$, regardless of what the particle player does. The particle player reasons that choosing $\delta r_x < \delta r_y$ would let the gravity player choose $g_x = 1$, resulting in a negative value for L. Likewise, choosing $\delta r_y < \delta r_x$ also lets the gravity player achieve $L < 0$. Thus, the particle player's best strategy is to simply choose $\delta r_x = \delta r_y = 0$ and not move at all. Both player's strategies can be cast in terms of linear programming. Since the gravity player focuses attention on the smaller of δr_x or δr_y, to achieve a value of L equal to the smaller of the two, the particle player's strategy is simply to choose δr_x and δr_y such that the *smaller* of the two is as *large* as possible (given the constraint $\delta r_x + \delta r_y \leq 0$).

Let us apply this reasoning to (34), but with the minus sign removed:

$$\max_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\mathbf{K}\delta\mathbf{p}\|_\infty \leq 1}} \left(\min_{\substack{\mathbf{g} \geq \mathbf{0} \\ \sum g_i = 1}} \mathbf{g}^T \mathbf{M} \delta\mathbf{p} \right). \tag{39}$$

Consider the vector $\mathbf{M}\delta\mathbf{p} \in \boldsymbol{R}^3$ (assuming we are working with a three-dimensional assembly). Given a vector $\delta\mathbf{p}$, the $\mathbf{g}$ that minimizes $\mathbf{g}^T\mathbf{M}\delta\mathbf{p}$ will be such that $g_i = 1$ where $(\mathbf{M}\delta\mathbf{p})_i$ is the smallest element of the vector $\mathbf{M}\delta\mathbf{p}$. Clearly then, the maximum of (39) occurs when $\delta\mathbf{p}$ is chosen so that the *minimum* component of $\mathbf{M}\delta\mathbf{p}$ is *maximized*. Such a $\delta\mathbf{p}$ can be found by linear programming. Then, assuming (34) is true, the solution to the dual of this linear program will yield the choice of $\mathbf{g}$ which maximizes $\bar{z}(\mathbf{g})$. In the next section, we explicitly describe a dual pair of linear programs that enable us to maximize $\bar{z}(\mathbf{g})$ and prove that (34) holds.

VI. Linear Programming Solutions of the Maximin Problem

In this section, we exhibit dual linear programs to solve (31) and (33), but with the minus sign removed. That is, we will find a solution vector $\mathbf{g}$ to the problem

$$\min_{\substack{\mathbf{g} \geq \mathbf{0} \\ \sum g_i = 1}} \left(\max_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\mathbf{K}\delta\mathbf{p}\|_\infty \leq 1}} L \right) \tag{40}$$

and a solution vector $\delta\mathbf{p}$ to the problem

$$\max_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\mathbf{K}\delta\mathbf{p}\|_\infty \leq 1}} \left(\min_{\substack{\mathbf{g} \geq \mathbf{0} \\ \sum g_i = 1}} L \right) \tag{41}$$

where $L = \mathbf{g}^T\mathbf{M}\delta\mathbf{p}$. The goal is to make the optimal solutions to the dual linear programs satisfy a condition called a "saddle-point condition." A pair of vectors $\mathbf{g}^*$ and $\delta\mathbf{p}^*$ satisfying the constraints on $\mathbf{g}$ and $\delta\mathbf{p}$ in problems (40) and (41) is called a saddle-point if for all $\mathbf{g}$ and $\delta\mathbf{p}$ which also satisfy the constraints of the problems, the relation

$$\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^* \tag{42}$$

holds. In terms of the game of the previous section, a saddle-point indicates a pair of strategies such that neither player is inclined to change their strategy, providing the other player holds constant as well. We will show that the solutions to the dual linear programs satisfy the saddle-point condition, and then prove that any vectors that satisfy the saddle-point condition solve problems (40) and (41).

Let us define the vector $\mathbf{b}$ by $\mathbf{b} = (1,1,1)^T$. In what follows, a *feasible* vector $\mathbf{g}$ is a vector satisfying $\mathbf{g} \geq \mathbf{0}$ and

$\sum g_i = \mathbf{b}^T\mathbf{g} = 1$. To express feasibility for a vector $\delta\mathbf{p}$, let $\mathbf{I}$ denote the $6n \times 6n$ identity matrix, and let $\mathbf{e}$ be a vector of length $6n$, with every element equal to one. If we define the vector $\mathbf{d}$ of length $m+12n$ and the $(m+12n)\times 6n$ matrix $\mathbf{B}$ by

$$\mathbf{d} = \begin{pmatrix} \mathbf{o} \\ \boldsymbol{e} \\ \boldsymbol{e} \end{pmatrix} \quad \text{and} \quad \mathbf{B} = \begin{pmatrix} -\boldsymbol{J} \\ \boldsymbol{K} \\ -\boldsymbol{K} \end{pmatrix} \tag{43}$$

then $\mathbf{B}\delta\mathbf{p} \leq \mathbf{d}$ implies

$$\mathbf{B}\delta\mathbf{p} = \begin{pmatrix} -\boldsymbol{J}\delta\boldsymbol{P} \\ \boldsymbol{K}\delta\boldsymbol{P} \\ -\boldsymbol{K}\delta\boldsymbol{P} \end{pmatrix} \leq \begin{pmatrix} \mathbf{o} \\ \boldsymbol{e} \\ \boldsymbol{e} \end{pmatrix}. \tag{44}$$

This in turn implies $\mathbf{J}\delta\mathbf{p} \geq \mathbf{0}$ and $-\mathbf{e} \leq \mathbf{K}\delta\mathbf{p} \leq \mathbf{e}$ which is equivalent to $\|\mathbf{K}\delta\mathbf{p}\|_\infty \leq 1$. Thus, we will say that a vector $\delta\mathbf{p}$ is feasible if $\mathbf{B}\delta\mathbf{p} \leq \mathbf{d}$.

A. Primal and Dual Linear Programs

Let v be a scalar and consider the (primal) linear program

$$\max v \text{ subject to} \begin{pmatrix} \mathbf{b} & -\mathbf{M} \\ \mathbf{0} & \mathbf{B} \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{\delta p} \end{pmatrix} \leq \begin{pmatrix} \mathbf{o} \\ \boldsymbol{d} \end{pmatrix}. \tag{45}$$

If we view the constraint $\mathbf{b}v - \mathbf{M}\delta\mathbf{p} \leq \mathbf{0}$ as $\mathbf{M}\delta\mathbf{p} \geq \mathbf{b}v$, we see that to maximize v we need to make $\mathbf{M}\delta\mathbf{p}$ as large as possible. In particular, v is bounded by the maximum that the smallest element of $\mathbf{M}\delta\mathbf{p}$ can attain, given the constraints on $\delta\mathbf{p}$. Thus, this linear program exactly captures the strategy articulated at the end of Section V-C to choose $\delta\mathbf{p}$. Since setting $v = 0$ and $\delta\mathbf{p} = \mathbf{0}$ satisfies the conditions in (45), it is clear that a solution to (45) exists. Note that for any pair $(v, \delta\mathbf{p})$ which satisfies the conditions in (45), $\mathbf{B}\delta\mathbf{p} \leq \mathbf{d}$ which implies that $\delta\mathbf{p}$ is feasible, as defined above.

The dual linear program to (45) is

$$\min(\mathbf{0}, \mathbf{d}^T)\begin{pmatrix} \boldsymbol{g} \\ \boldsymbol{s} \end{pmatrix} \text{subject to } (\mathbf{g}^T, \mathbf{s}^T)\begin{pmatrix} \mathbf{b} & -\mathbf{M} \\ \mathbf{0} & \mathbf{B} \end{pmatrix} = (1, \mathbf{0}), \tag{46}$$

where $\mathbf{s} \in \boldsymbol{R}^{m+12n}$. Note that any $\mathbf{g}$ satisfying the conditions in (46) satisfies $\mathbf{g} \geq \mathbf{0}$ and $\mathbf{g}^T\mathbf{b} = 1$, and is thus feasible.

B. Saddle-Point Condition

Let $\mathbf{g}^*$ and $\mathbf{s}^*$ be optimal solutions to (46) and let v^* and $\delta\mathbf{p}^*$ be optimal solutions to (45). The duality theory of linear programming [16] states that the optimal values of the two linear programs are equal: that is,

$$v^* = (0, \mathbf{d}^T)\begin{pmatrix} \boldsymbol{g*} \\ \boldsymbol{s*} \end{pmatrix} = \mathbf{d}^T\mathbf{s}^*. \tag{47}$$

The duality theory also states that

$$(\mathbf{g}^{*T}, \mathbf{s}^{*T})\begin{pmatrix} \boldsymbol{M\delta p*} - \boldsymbol{b\nu*} \\ \boldsymbol{d} - \boldsymbol{B\delta p*} \end{pmatrix} = 0. \tag{48}$$

The fact that both vectors in this equation are nonnegative (due to the conditions in the linear programs) means that

$$\mathbf{g}^{*T}(\mathbf{M}\delta\mathbf{p}^* - \mathbf{b}v^*) = 0 \tag{49}$$

or

$$\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* = \mathbf{g}^{*T}\mathbf{b}v^* = v^* \tag{50}$$

since $\mathbf{g}^{*T}\mathbf{b} = 1$.

Since $\mathbf{g}^*$ and $\mathbf{s}^*$ satisfy the conditions in (46), $-\mathbf{g}^{*T}\mathbf{M} + \mathbf{s}^{*T}\mathbf{B} = \mathbf{0}$ or equivalently, $\mathbf{g}^{*T}\mathbf{M} = \mathbf{s}^{*T}\mathbf{B}$. For any feasible $\delta\mathbf{p}$ then,

$$(\mathbf{g}^{*T}\mathbf{M})\delta\mathbf{p} = (\mathbf{s}^{*T}\mathbf{B})\delta\mathbf{p} = \mathbf{s}^{*T}(\mathbf{B}\delta\mathbf{p}) \leq \mathbf{s}^{*T}\mathbf{d} \tag{51}$$

since $\mathbf{B}\delta\mathbf{p} \leq \mathbf{d}$ and $\mathbf{s}^*$ is nonnegative. Using the fact that $\mathbf{s}^{*T}\mathbf{d} = v^* = \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^*$, we obtain

$$\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \tag{52}$$

for all feasible $\delta\mathbf{p}$.

To obtain the other half of the saddle-point condition, observe that $\mathbf{M}\delta\mathbf{p}^* \geq \mathbf{b}v^*$. Since $\mathbf{g} \geq \mathbf{0}$ and $\mathbf{g}^T\mathbf{b} = 1$ for any feasible $\mathbf{g}$,

$$\mathbf{g}^T(\mathbf{M}\delta\mathbf{p}^*) \geq \mathbf{g}^T(\mathbf{b}v^*) = (\mathbf{g}^T\mathbf{b})v^* = v^* = \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^*. \tag{53}$$

Combining the previous two inequalities gives the saddle-point condition

$$\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^*. \tag{54}$$

for any feasible $\mathbf{g}$ and $\delta\mathbf{p}$.

C. Maximin Result

We claim that (40) attains its minimum value (and thus $\bar{z}(\mathbf{g})$ attains its maximum value) for $\mathbf{g} = \mathbf{g}^*$, and that (41) attains its maximum value for $\delta\mathbf{p} = \delta\mathbf{p}^*$. Furthermore, these values are equal. In what follows, we restrict our attention to feasible vectors $\delta\mathbf{p}$ and $\mathbf{g}$ (that is, we will not write out the feasibility conditions).

To show that $\mathbf{g}^*$ minimizes (40) we need to show that

$$\max_{\delta\mathbf{p}} \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}. \tag{55}$$

for all feasible $\mathbf{g}$. From the saddle-point condition, $\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^*$ for all feasible $\delta\mathbf{p}$, so $\max_{\delta\mathbf{p}} \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}$ is bounded from above by $\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^*$. Since $\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^*$ for any feasible $\mathbf{g}$, we have

$$\max_{\delta\mathbf{p}} \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^* \leq \max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p} \tag{56}$$

for any feasible $\mathbf{g}$.[4] Thus,

$$\min_{\mathbf{g}} \left(\max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right) = \max_{\delta\mathbf{p}} \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^*. \tag{57}$$

Similarly, the saddle-point condition yields

$$\min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^* \tag{58}$$

[4]The inequality $\mathbf{g}^T\mathbf{M}\delta\mathbf{p}^* \leq \max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}$ follows from the fact that $\max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}$ is greater than or equal to $\mathbf{g}^T\mathbf{M}\delta\mathbf{p}$ for *any* choice of $\delta\mathbf{p}$. In particular then if we choose $\delta\mathbf{p} = \delta\mathbf{p}^*$, $\max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p} \geq \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^*$.

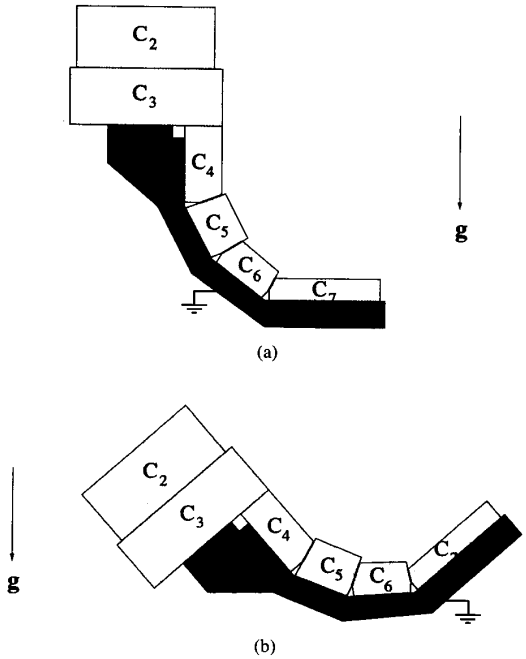


Fig. 6. Optimal orientations for an assembly based on two different criteria — (a) Minimizing the number of unstable objects, (b) Minimizing $-\ddot{U}$.

for all feasible $\delta\mathbf{p}$, so $\delta\mathbf{p}^*$ maximizes 41 and

$$\mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^* = \max_{\delta\mathbf{p}} \left(\min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right). \quad (59)$$

This proves that $\mathbf{g} = \mathbf{g}^*$ and $\delta\mathbf{p} = \delta\mathbf{p}^*$ are optimal for problems (40) and (41), respectively, and

$$\min_{\mathbf{g}} \left(\max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right) \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \max_{\delta\mathbf{p}} \left(\min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right). \quad (60)$$

Thus, we can find a value of $\mathbf{g}$ which maximizes $\bar{z}(\mathbf{g})$ by solving the linear program of (46). The motion $\delta\mathbf{p} = \delta\mathbf{p}^*$ which minimizes δU for the orientation $\mathbf{g}^*$ according to the constraints $\|\mathbf{K}\delta\mathbf{p}\|_\infty \leq 1$ and $\mathbf{J}\delta\mathbf{p} \geq \mathbf{0}$ can be found by solving the linear program (45).

Finally, we can use the optimal vectors $\mathbf{g}^*$ and $\delta\mathbf{p}^*$ to obtain

$$\begin{aligned} &\max_{\delta\mathbf{p}} \left(\min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right) \\ &\quad = \min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}^* \leq \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* \leq \max_{\delta\mathbf{p}} \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p} \\ &\quad = \min_{\mathbf{g}} \left(\max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right) \end{aligned} \quad (61)$$

which yields

$$\max_{\delta\mathbf{p}} \left(\min_{\mathbf{g}} L\right) \leq \min_{\mathbf{g}} \left(\max_{\delta\mathbf{p}} L\right). \quad (62)$$

Combining the inequalities in (60) and(62) proves that

$$\min_{\mathbf{g}} \left(\max_{\delta\mathbf{p}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right) = \mathbf{g}^{*T}\mathbf{M}\delta\mathbf{p}^* = \max_{\delta\mathbf{p}} \left(\min_{\mathbf{g}} \mathbf{g}^T\mathbf{M}\delta\mathbf{p}\right). \quad (63)$$

Hence (34) holds.

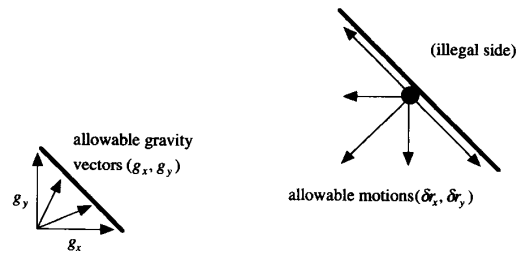


Fig. 7. The motion of the particle is constrained because it is in contact with a wall. The optimal strategies are to choose $g_x = g_y = 1/2$, and $\delta r_x = \delta r_y = 0$.

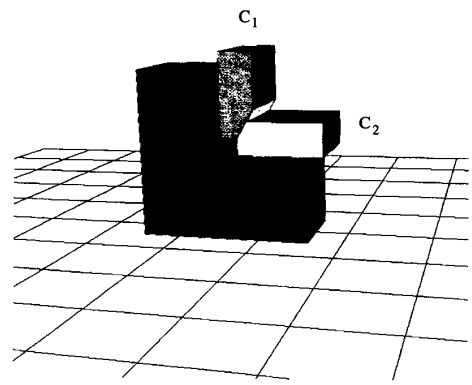


Fig. 8. Determining stability. The L-shaped part is grounded and the two blocks are chamfered at 30° and 60°. The quantity $\bar{z}$ is minimized for a displacement in which block C_1 moves downward by one unit and block C_2 moves to the right by 0.577 units.

VII. Results and Discussion

We have tested our method for finding stable and optimally-stable orientations on a number of sample assemblies. Geometric models were created using the nonmanifold geometric modeler Noodles [17]. The geometry of contact required to construct the linear programs was derived directly from the models. Figs. 8 and 9 show assemblies that were tested for stability using the potential-energy formulation.

Fig. 6 shows an assembly of 3 parts where the large L-shaped part is grounded. Gravity acts in the direction perpendicular to the bottom face of the grounded part. The two smaller blocks are chamfered at angles of 30° and 60°. The part motions that minimized $\bar{z}$ for this assembly indicated a downward translation for of magnitude 1.0 for the 30° block and a sideward translation of 0.577 ($1/\tan 60°$) for the 60° block. (In this case, because of the lack of rotational movement, the part motions that minimize $\bar{z}$ exactly indicate the direction of impending motion for the assembly.)

The assembly in Fig. 7 consists of four identical blocks chamfered at two edges at an angle of 45°. They are placed within a grounded square frame as shown. This assembly is a good example of the interdependence of part motions because of the presence of loops within the contact graph. This assembly is stable because although the parts can move in a cyclic fashion, this motion does not result in a net decrease in potential energy. The potential-energy formulation returns the result that this structure is stable.

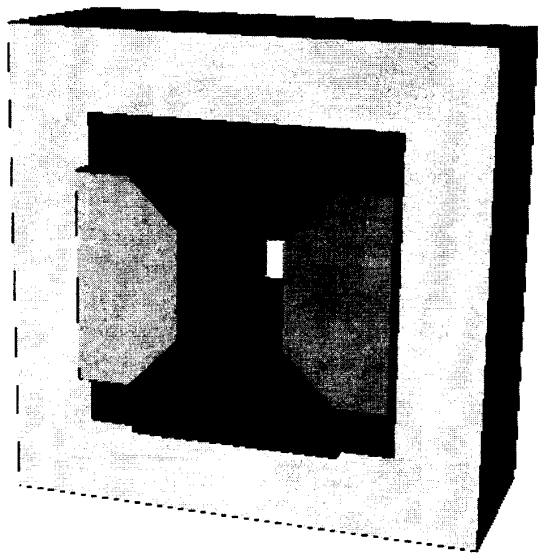
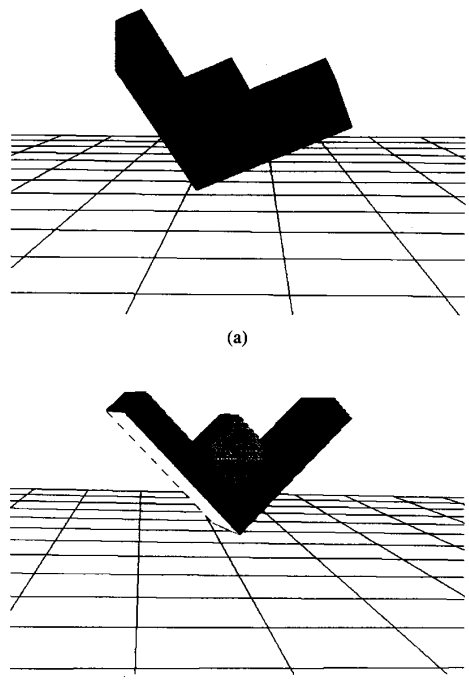


Fig. 9. An assembly with 4 identical blocks within a grounded frame. Gravity point straight down. The assembly is stable.



(a)

(b)

Fig. 10. Finding a stable orientation. The L-shaped part is grounded. (a) Initial unstable orientation. (b) Computed stable orientation.

Figs. 10, 11 and 12 illustrate the problem of finding a stable orientation. In all three figures, the gravity vector points straight down (that is, perpendicular to the reference plane shown). In the first two figures, the L-shaped parts are grounded while in the third the U-shaped part is grounded. On the left, the assemblies are shown in unstable orientations. On right, a computed stable orientation is shown. All the solutions are obtained using the dual linear program in (46). For each of the assemblies, the size of the linear programs is presented. Since the linear programs are relatively small in size, it takes less than 0.2 cpu seconds to solve each of them on a DEC5000 workstation.

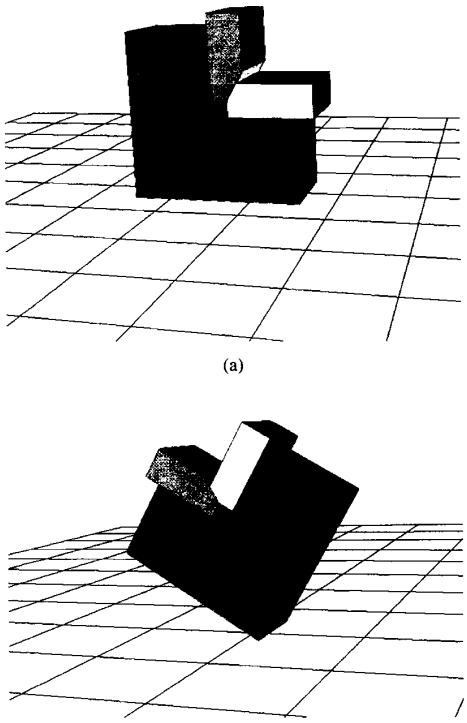
The linear program used to find a stable orientation for the assembly in Fig. 8 had 23 variables and 7 constraints. Note that the assembly is stable over a range of orientations. The method that we have proposed does not distinguish between these orientations, and may choose any one of them as a stable orientation. We plan to extend our method to produce the entire range of stable orientations.

Fig. 10 shows an example of instability caused by the rotation of a part. In the unstable orientation on the left, the L-shaped part with a large bulb at its end pivots about the wedge on the grounded part, causing the two smaller parts to move as well. The computed orientation on the right however is stable. This example has 67 variables and 19 constraints.

Fig. 11 is an example of the least-unstable orientation problem. It shows the assembly of Fig. 3 which is unstable in all orientations. The least unstable orientation obtained ensures that three of the blocks are motionless while the fourth moves downwards. This orientation can be selected as one suitable for assembly if the unstable block can be stabilized in some way. One way of doing this in situations where grippers are being used is by including the block for consideration during grasp planning so as to ground the block along with the T-shaped part. Note that in the orientation on the right, the three stable blocks are not oriented vertically. The reason is that an orientation which made these three blocks stand exactly upright would actually be *less* stable than the orientation shown. In the orientation shown on the right, the three top blocks are stable, and in addition, the bottom block does not fall downwards freely, but slides along its contact surface. As a result the bottom block accelerates more slowly in this orientation that it would in an orientation in which the top three blocks were exactly upright. (This sort of analysis is of course much more easily performed *after* one has seen the orientation of Fig. 11 than before.) This example has 87 variables and 25 constraints.

VIII. Summary and Conclusions

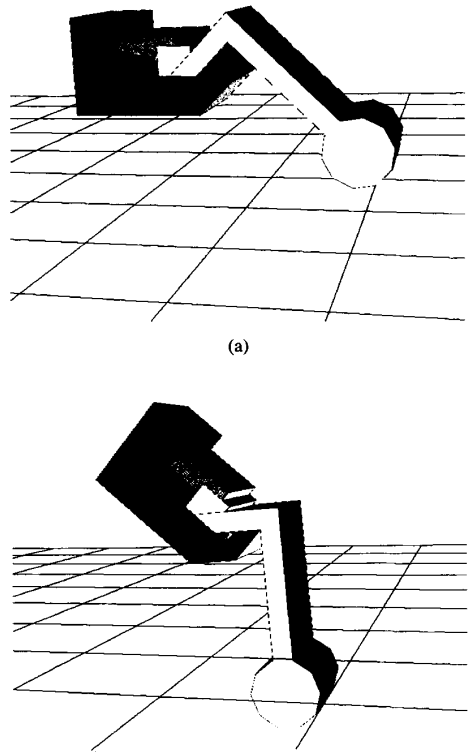
Geometric models of parts and assemblies are increasingly being used to plan manufacturing processes, including assembly. Analysis of the physical behavior of objects and assemblies is being automatically derived for process planning. In this paper, the problem of the gravitational stability of assemblies is addressed. It is assumed that assemblies consist of rigid objects with frictionless contacts, and that one (or more) of the objects is grounded. The second assumption, in general, is a statement of constraint on the absolute motion of some objects. Solutions to two problems have been proposed. We have considered the problem of finding a stable orientation for an assembly, and shown how such an orientation (if it exists) can be formulated using linear programming. We have also considered assemblies that are unstable under all orientations. We have proposed a physically-motivated



(a)

(b)

Fig. 11. Finding a stable orientation. The L-shaped part is grounded. (a) Initial unstable orientation. (b) Computed stable orientation.



(a)

(b)

Fig. 12. Finding a stable orientation. The U-shaped part is grounded. (a) Initial unstable orientation. (b) Computed stable orientation.

measure of stability based on potential energy considerations. The maximization of this stability measure was expressed as a maximin problem in terms of the orientation vector and the motions of the bodies. An interpretation of this maximin problem as a two-person zero-sum game led to a linear programming formulation of the maximin problem. Linear programming-based procedures to solve the two problems have been implemented and preliminary results have been presented. The solution to these problems is the first general method that automatically determines stable orientations and optimally-stable orientations of assemblies. This is a useful tool in automatic assembly planning.

In future work, the stability analysis needs to be extended to include frictional forces. The stability result that is predicted by a frictionless model is too conservative. By considering friction, the range of stable orientations found can be extended. Additionally, we would like to extend our method so that it can report the range of stable orientations, rather than picking a single stable orientation arbitrarily. This is useful during assembly planning when an assembly needs to be reoriented.

The proposed method can be applied to study the stability of assemblies in the presence of external forces other than gravitational forces. When two subassemblies come together during the process of assembly, objects within each of the subassemblies are subjected to forces when they interact with the other subassembly. Knowledge about the stability of subassemblies in such situations can be obtained by including the work done by the forces in the formulation. Another application of this method is in verifying grasp planning. In the example in Fig. 11, grasping only the T-shaped object does not produce a stable assembly, no matter what the orientation. One way of obtaining a stable assembly in such situations is by including the unstable objects within the grasp. A grasp plan applies forces to objects in an assembly subject to constraints arising from the capabilities of the grasping agent. Thus again, the proposed method can analyze the stability of an assembly in the presence of external grasp and gravitational forces to verify grasp plans.

APPENDIX A

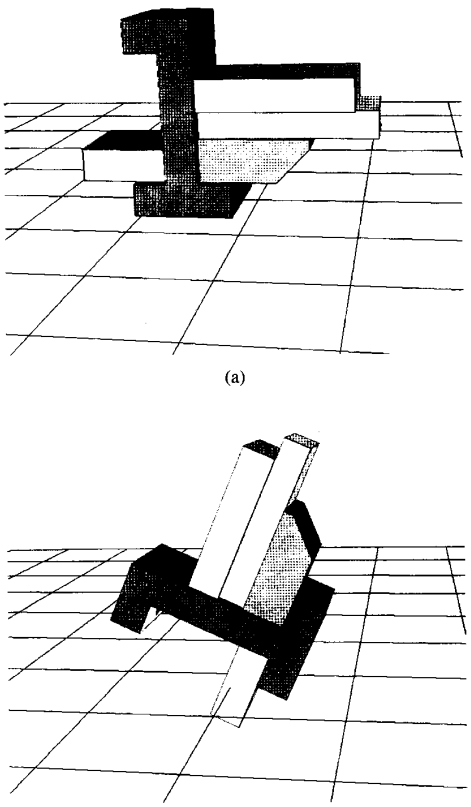
In this Appendix, we will show the relation between the stability measure S defined by

$$S = \min_{\substack{\mathbf{J}\delta\mathbf{p} \geq \mathbf{0} \\ \|\delta\mathbf{p}\|_{\mathbf{N}} \leq 1}} -\mathbf{g}^T \mathbf{M} \delta\mathbf{p}. \tag{64}$$

and $\ddot{U}$ for a fixed gravity vector $\mathbf{g}$.

Let the acceleration of the parts in an assembly be denoted by the vector $\mathbf{a} \in \boldsymbol{R}^{6n}$, where

$$\mathbf{a} = \begin{pmatrix} \dot{\mathbf{v}}_1 \\ \dot{\boldsymbol{\omega}}_1 \\ \vdots \\ \dot{\mathbf{v}}_n \\ \dot{\boldsymbol{\omega}}_n \end{pmatrix} \tag{65}$$



(a)

(b)

Fig. 13. Finding the least unstable orientation. The T-shaped part is grounded. (a) An assembly that is unstable in all orientations. (b) The least unstable orientation.

and $\mathbf{v}_i$ and $\boldsymbol{\omega}_i$ denote the linear and angular velocity of the ith part. In Baraff and Mattikalli [15], it is shown that for an initially motionless system with motion constraints $\mathbf{J}\delta\mathbf{p} \geq \mathbf{0}$ and a uniform gravity field $\mathbf{g}$, the acceleration $\mathbf{a}$ is a nonnegative scalar multiple of the solution $\delta\mathbf{p}$ to (64).

Furthermore, $\mathbf{a}$ can be written in the form

$$\mathbf{a} = \mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{N}^{-1}\mathbf{Q}_g \tag{66}$$

where $\mathbf{Q}_g$ and $\boldsymbol{\lambda}$ are vectors in R^{6n}. The vector $\mathbf{Q}_g$ represents the generalized force and is given by $\mathbf{Q}_g = \mathbf{M}^T\mathbf{g}$; note that since U is the gravitational potential energy of the system,

$$\nabla U = -\mathbf{M}^T\mathbf{g} = -\mathbf{Q}_g. \tag{67}$$

The vector $\boldsymbol{\lambda}$ gives the contact force magnitudes that arise between objects at contact points, and satisfies the relation

$$\boldsymbol{\lambda}^T(\mathbf{J}\mathbf{a}) = \boldsymbol{\lambda}^T(\mathbf{J}\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{J}\mathbf{N}^{-1}\mathbf{Q}_g) = 0 \tag{68}$$

which implies that

$$\boldsymbol{\lambda}^T\mathbf{J}\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} = -\boldsymbol{\lambda}^T\mathbf{J}\mathbf{N}^{-1}\mathbf{Q}_g = -\mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda}. \tag{69}$$

If the assembly is stable with orientation $\mathbf{g}$, then the stability measure S, the acceleration $\mathbf{a}$, and the second derivative $\ddot{U}$ of energy loss are all zero. Suppose however that the assembly is not stable, so that $S < 0$ and $\mathbf{a}$ is nonzero. In this case, it is clear that the minimum value of S is achieved by a vector $\delta\mathbf{p}$ satisfying $\|\delta\mathbf{p}\|_{\mathbf{N}} = 1$. Knowing that $\delta\mathbf{p}$ is parallel to $\mathbf{a}$ and that $\|\delta\mathbf{p}\|_{\mathbf{N}} = 1$, it must be that the solution $\delta\mathbf{p}$ to 64 is given by

$$\delta\mathbf{p} = \frac{\mathbf{a}}{\|\mathbf{a}\|_{\mathbf{N}}} \tag{70}$$

which means that

$$S = -\mathbf{g}^T\mathbf{M}\delta\mathbf{p} = -\mathbf{Q}_g^T\delta\mathbf{p} = \frac{-\mathbf{Q}_g^T\mathbf{a}}{\|\mathbf{a}\|_{\mathbf{N}}}. \tag{71}$$

However, since the gradient of U is $-\mathbf{Q}_g$, and the system is initially motionless, we have

$$\ddot{U} = (\nabla U)^T\mathbf{a} = -\mathbf{Q}_g^T\mathbf{a}. \tag{72}$$

Let us compare S and $\ddot{U}$. Expanding the product $\mathbf{Q}_g^T\mathbf{a}$ using (66) yields

$$\begin{aligned}\mathbf{Q}_g^T\mathbf{a} &= \mathbf{Q}_g^T(\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{N}^{-1}\mathbf{Q}_g)\\ &= \mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{Q}_g.\end{aligned} \tag{73}$$

However, examining $\|\mathbf{a}\|_{\mathbf{N}}^2$, we find that

$$\begin{aligned}\|\mathbf{a}\|_{\mathbf{N}}^2 &= \mathbf{a}^T\mathbf{N}\mathbf{a}\\ &= (\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{N}^{-1}\mathbf{Q}_g)^T\mathbf{N}(\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{N}^{-1}\mathbf{Q}_g)\\ &= \boldsymbol{\lambda}^T\mathbf{J}\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{Q}_g + 2\mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda}.\end{aligned} \tag{74}$$

Using (69), we replace $\boldsymbol{\lambda}^T\mathbf{J}\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda}$ with $-\mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda}$ to obtain

$$\begin{aligned}\|\mathbf{a}\|_{\mathbf{N}}^2 &= -\mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda} + \mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{Q}_g + 2\mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda}\\ &= \mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{Q}_g + \mathbf{Q}_g^T\mathbf{N}^{-1}\mathbf{J}^T\boldsymbol{\lambda}.\end{aligned} \tag{75}$$

But from (73), this yields simply

$$\|\mathbf{a}\|_{\mathbf{N}}^2 = \mathbf{Q}_g^T\mathbf{a} \tag{76}$$

or equivalently,

$$\|\mathbf{a}\|_{\mathbf{N}} = \sqrt{\mathbf{Q}_g^T\mathbf{a}}. \tag{77}$$

This gives as a final result

$$S = \frac{-\mathbf{Q}_g^T\mathbf{a}}{\|\mathbf{a}\|_{\mathbf{N}}} = \frac{-\mathbf{Q}_g^T\mathbf{a}}{\sqrt{\mathbf{Q}_g^T\mathbf{a}}} = -\sqrt{\mathbf{Q}_g^T\mathbf{a}} = -\sqrt{-\ddot{U}} \tag{78}$$

or simply

$$\ddot{U} = -S^2. \tag{79}$$

Acknowledgment

The authors would like to thank Matt Mason for very useful input during early discussions. Also the help of Levent Gürsöz and Jim Hemmerle for their insight into the mechanics of bodies, and Jagadisan Viswanathan's contribution during our discussions in solving linear programming problems is greatly appreciated.

REFERENCES

[1] L. S. H. de Mello, "Task sequence planning for robotic assembly," Ph.D. dissertation, Carnegie Mellon University, May 1989.

[2] R. S. Mattikalli, P. Khosla, and Y. Xu, "Subassembly identification and motion generation for assembly: A geometrical approach," in *Proc. IEEE Conf. Syst. Eng.*, 1990, pp. 399–403.

[3] R. H. Wilson, "On Geometric Assembly Planning," Ph.D. dissertation, Department of Computer Science, Stanford University, Mar. 1992.

[4] M. Blum, A. Griffith, and B. Neumann, "A stability test for configurations of blocks," *Tech. Rep. AI Memo no. 188*, Massachusetts Institute of Technology, Feb. 1970.

[5] R. Palmer, "Computational complexity of motion and stability of polygons," Ph.D. dissertation, Dept. of Computer Science, Cornell University, May 1989.

[6] J. C. Trinkle, "On the stability and instantaneous velocity of grasped frictionless objects," *IEEE Trans. Robot. and Automat.*, vol. 8, pp. 560–572, Oct. 1992.

[7] N. Boneschanscher, H. van der Drift, S. Buckley, and R. Taylor, "Subassembly stability," in *Proc. Nat. Conf. Artificial Intell.*, 1988, pp. 780–785.

[8] W. Ingleton, "A problem in linear inequalities," in *Proc. London Math. Soc.*, vol. 16, no. 3, pp. 519–536, 1966.

[9] P. Lötstedt, "Mechanical systems of rigid bodies subject to unilateral constraints," *SIAM J. Appl. Math.*, vol. 42, pp. 281–296, Apr. 1982.

[10] R. Featherstone, *Robot Dynamics Algorithms.* Norwell, MA: Kluwer, 1987.

[11] D. Baraff, "Analytical methods for dynamic simulation of nonpenetrating rigid bodies," *Comput. Graph.*, vol. 24, pp. 223–232, Aug. 1989.

[12] R. Mattikalli, P. Khosla, B. Repetto, and D. Baraff, "Stability of assemblies," in *Proc., IEEE/RSJ Int. Conf. on Intell. Robots and Syst.*, 1993.

[13] D. Baraff and A. Witkin, "Dynamic simulation of nonpenetrating flexible bodies," *Comput. Graph. (Proc. SIGGRAPH)*, vol. 26, pp. 303–308, July 1992.

[14] R. S. Mattikalli and P. K. Khosla, "Motion constraints from contact geometry: Representation and analysis," in *Proc. IEEE Int. Conf. Robot. and Automat.*, 1992, vol. 3, pp. 2178–2185.

[15] D. Baraff and R. Mattikalli, "Inpending motion direction of contacting rigid bodies," Tech. *Rep. CMU-RI-TR-93-15*, Carnegie Mellon University, 1993.

[16] G. Strang, *Introduction to Applied Mathematics.* Wellesley-Cambridge Press, 1986.

[17] L. Gürsöz, Y. Choi, and F. Prinz, "Vertex-based representation of non-manifold boundaries," in *Geometric Modeling for Product Engineering.* New York: North-Holland, 1988, pp. 107–130.

Raju Mattikalli received the B.Tech. in mechanical engineering from the Indian Institute of Technology, Bombay, in 1987 and the M.Eng. and Ph.D. in mechanical engineering from Carnegie Mellon University in 1989 and 1994, respectively.

His main research interests lie in the areas of computer-aided design and manufacture, which include topics in design for manufacture, geometry-based planning, mathematical simulation of manufacturing processes, and geometric modeling. His dissertation work was on mechanics-based assembly planning using geometric models of parts. Currently, he is a Post-Doctoral Research Associate at the Engineering Design Research Center at Carnegie Mellon University.

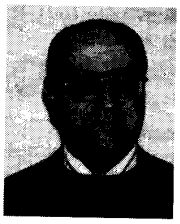
David Baraff received the BS.E. in computer science from the University of Pennsylvania in 1987. He received the Ph.D. from Cornell University in 1992, where he was a graduate student in the Department of Computer Science and Program of Computer Graphics.

Before and during his undergraduate studies he worked in AT&T Bell Laboratories Computer Technology Research Laboratory doing computer graphics research. After receiving the Ph.D., he joined the faculty of Carnegie Mellon University in 1992 as an Assistant Professor in the Robotics Institute, and School of Computer Science. His research interests include physical stimulation and modeling for computer graphics, robotics, and animation.

Dr. Baraff was named an ONR Young Investigator in 1995.

Pradeep Khosla received the M.S. and Ph.D. degrees from Carnegie Mellon University, Pittsburgh, PA.

He is currently a Professor in the Department of Electrical and Computer Engineering at Carnegie Mellon University. He is also a member of the Robotics Institute and Director of the Advanced Manipulators Laboratory. Prior to joining Carnegie Mellon University, he worked with Tata Consulting Engineers and Siemens in the area of real-time control. His research interests are in the area of real-time sensor-based manipulation, architectures for real-time control, integrated design-assembly systems, and robotic applications in space, field, and manufacturing environments. His research has resulted in more than 100 journal articles, conference papers, and book contributions. He also serves as a consultant to several industries in the U.S.



Bruno Repetto holds a Ph.D. in industrial administration operations research from the Graduate School of Industrial Administration, Carnegie Mellon University.

His interests are in the areas of discrete and combinatorial optimization and mathematical programming. He is currently researching new upper and lower bounding procedures for large-scale Traveling Salesman problems (asymmetric version). He is Senior Operations Research Scientist at Management Science Associates, Process Automation Solutions and Services (MSA PASS), Pyromet Division, where he is developing various mathematical models to be applied in the steel industry.