Application of Computational Topology to the Design of Microelectromechanical Structures

Karl-Friedrich Böhringer
Department of Computer Science
Cornell University
Ithaca, NY 14853

Abstract

We are applying tools from computational topology and combinatorial geometry in our investigation of computational aspects of engineering design problems. We seek tractable algorithmic solutions by exploiting algebraic formulations of the fundamental computational-topological problems arising in the kinematic and dynamic analysis of microelectromechanical (MEM) structures. In particular, we use concepts from computational topology and geometry to determine the shape of microelectromechanical (MEM) structures, given specifications of their parameterized geometry and their functionality (parametric design).
MEM structures are particularly suited for automated parametric design because of (i) their highly specialized and automated production process, which allows only a very limited number of elementary structures, (ii) the tractable geometry and functionality of the devices, and (iii) the possibly large number of devices produced simultaneously in one manufacturing process. Kinematic interactions between mechanism components are reflected in their configuration space. Each pair of components in contact is responsible for a (hyper-) surface region (algebraic variety) in configuration space. We describe a class of mechanisms (micromechanical hinged structures) whose kinematic interactions can be encoded by low-degree algebraic curves in configuration space. We then compute the topology of the arrangement created by these curves. Two surface regions in the arrangement are adjacent if and only if there exists a physically feasible mechanical transition between the corresponding contact states of the mechanism. Adjacency relationships in the configuration space arrangement define a dual graph whose nodes represent contact states of the mechanism, and whose edges represent mechanical state transitions. This "configuration space graph" can be interpreted as a concise encoding of the relationship between configuration space topology and kinematic behavior of the mechanism.
The problem of parametric design can be solved by determining design parameter values such that a required mechanism behavior is achieved. The incorporation of design parameters into the mechanism geometry is reflected in configuration space by a particular family of surfaces parameterizations. The topology of the surface arrangement will therefore depend on the specific parameter values. We describe an algorithm that determines all possible arrangement topologies and their corresponding design parameter values for the class of micromechanical hinged structures. If the required mechanism behavior can be obtained by a certain configuration space topology then the algorithm solves the parametric design problem. Now, general mechanical system behavior is also constrained by dynamics, friction, and other constitutive differential inclusions. We intend to annotate the permissible configuration space topologies for the mechanism with the dynamically feasible transitions by lifting our analysis to the tangent bundle. Already, our kinematic analysis to design will significantly simplify the solution of the other subproblems.
The microscopic size of MEM structures, the possibly large number of devices employed simultaneously, and uncertainty in the manufacturing process suggest the use of randomizing assembly algorithms. We develop a connection between randomization techniques and our approach for design automation.

1 Introduction

This paper describes our current work in engineering geometric design. The goal of this subfield of industrial design is to find algorithmic, computational methods that determine the geometry of an artifact such that it satisfies or optimizes given functional requirements ("form from function").

The design process for an artifact can be viewed as a series of transformations through different levels of abstraction that characterize the artifact (Kannapan and Marshek 1991):

- requirements, specifying functionality and domain constraints;
- structure, describing the elementary components;
- parametrization, giving a parameterized descrip-

Proc. NSF Design and Manufacturing Systems Grantees Conference, Charlotte, NC, January 1993.

tion of the mechanism;
- geometry, fixing the exact dimensions of all components;
- manufacture, specifying the production sequence.

Design automation aims at algorithmic solutions for these transformations. Our work focuses on the transition from parameter space to geometry space (*parametric design*) in the domain of microelectromechanical (MEM) structures.

Our approach starts out with the configuration space (C-space) of the mechanism (Lozano-Pérez 1983; Brost 1989): If a mechanism has n degrees of freedom, then each configuration can be identified with an n-dimensional point. Each configuration in Euclidean space corresponds to a point in C-space, and vice-versa. This technique allows an elegant and convenient identification of forbidden and allowed configurations (C-space obstacles and free space). Forbidden configurations are e.g. configurations in which parts would overlap. Points on the surface of C-space obstacles correspond to configurations in which movable parts of the mechanism are in contact.

These surfaces can be divided into regions with uniform kinematic properties, similar to (Brooks and Lozano-Pérez 1982; Donald 1989; Joskowicz 1989; Joskowicz 1990; Brost 1991). E.g., all configurations where a vertex of one part of the mechanism slides along a surface of another part could form such a region. The shape of every region is governed by a unique set of conditions derived from the geometry of the mechanism. (Donald and Pai 1992) describe classes of mechanisms that generate low-degree algebraic equations for these conditions, and give an efficient algorithm to compute them.

We assume that the mechanism usually operates in configurations corresponding to C-space surfaces, and not in free space. This is a reasonable assumption, as engineered devices are used mostly in such configurations. This property may be enforced by gravity, or by superimposing other dynamic constraints, such as springs that push parts together.

Regions in C-space can be seen as different operation modes of the mechanism, as each of them is governed by different sets of geometric conditions. Together with their adjacency relationships, these regions define a graph that gives a simplified description of kinematic behavior, abstracting away from geometric details. Let us call this graph a C-space graph.

When doing parametric design, we consider a parameterized mechanism geometry. Each design parameter can be seen as an additional degree of freedom of the mechanism. If there are m design parameters, we get an $(n+m)$-dimensional generalized configuration space (Pai 1988; Donald 1989). The C-space graph of a parameterized mechanism may have nodes and edges that exist only for certain design parameter values. Thus for different parameter values we get different behaviors of the mechanism.

If the required behavior of the mechanism can be specified as (paths of) nodes in a C-space graph, then parametric design can be done by "unifying" these nodes with a subgraph of the mechanism's parameterized C-space graph. The parameter values can then be "collected" along this path.

In the following sections we briefly introduce MEM structures and snap fasteners as a domain for automated design. We develop a model for these structures that is appropriate for the approach mentioned above. We then give a sample hinged MEM structure to demonstrate our ideas.

In the domain of MEM structures the issue of easy assembly is of major importance. The small size and the possibly large number of mechanisms make conventional assembly techniques too expensive and too time-consuming. Ideally one would like to have structures that are "self-assembling." We investigate the application of randomized strategies (Erdmann 1989; Erdmann 1992) for assembly of MEM structures.

2 Sample Domains for Design Automation

2.1 Microelectromechanical (MEM) Structures

A wide variety of micromechanical structures (devices typically in the micrometer range) have been built recently by using processing techniques known from integrated circuit industry, including micromotors, gears, tweezers, diverse sensors, among others (see e.g. (Journal of MEMS 1992)).

As a consequence of the production process MEM structures are essentially two-dimensional. Microfabricated hinged structures (Pister et al. 1992) allow the construction of three-dimensional devices. The basic idea is to make a plate that can rotate about a single axis defined by a pair of hinges (Figure 1). When the plate rotates out of the wafer, it has to push a spring loaded lock out of the way (Figure 2). The lock is designed such that when the plate reaches a certain angle (typically $\pi/2$), the lock snaps down into a slot in the plate, resulting in a rigid three-dimensional structure. Two different spring loaded lock designs exist: Torsional spring locks and cantilevered beam spring locks. Torsional spring locks (Figure 1) have a rigid lock attached to a long flexible beam that allows the lock to deviate up to a certain angle from the wafer plate. Cantilevered beam locks have a long flexible lock that bends on its entire length if the plate is rotated.

The structures are rotated into place either by a probe station under the microscope (which is very labor intensive) or by hydraulic forces, i.e. shaking the wafer under water, such that the random motion of the water moves the parts into their final position (Burgett et al. 1992).

The problem then is how to design such micromechanical devices. This involves
- satisfying functional requirements for the assembled parts,
- satisfying geometric and engineering constraints for the manufacturing process,
- and dealing with physical phenomena like friction, material properties, and liquid flow.

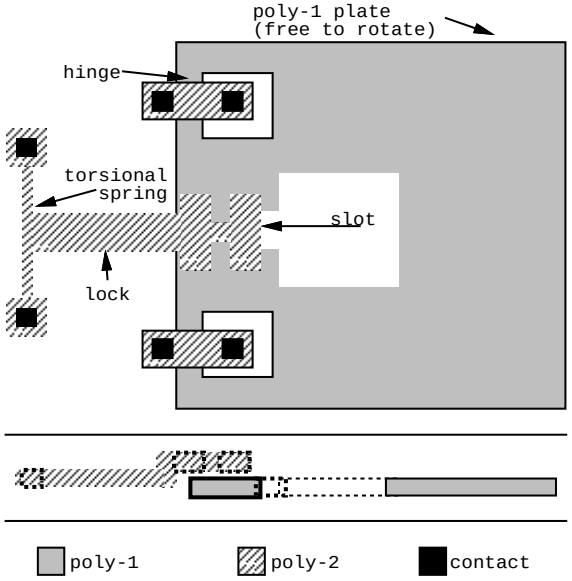

Figure 1: Micromechanical Hinged Plate (Top and Side View)

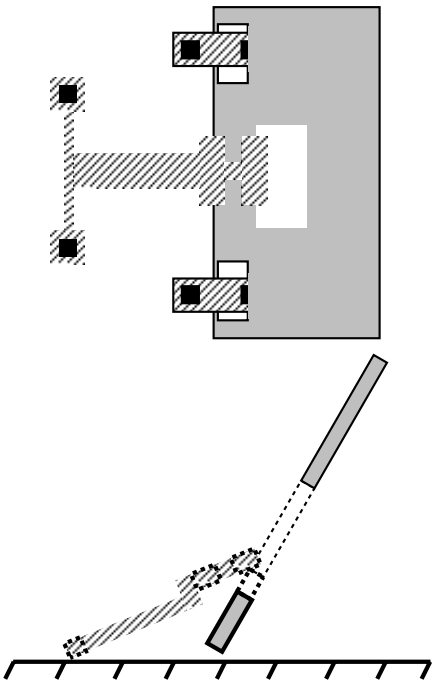

Figure 2: Micromechanical Hinged Mechanism Rotating out of the Wafer Plane (Top and Side View)

2.2 Snap Fasteners

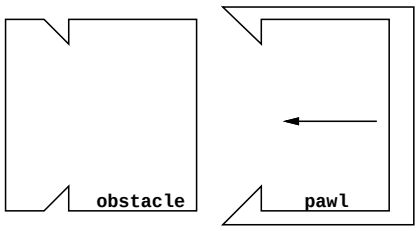

Figure 3: Snap Fastener

Screws and rivets are conventional fastening techniques for assembly processes. In recent years assemblies have been simplified significantly by the use of flexible parts (mainly made of plastic) that simply snap together: During assembly an external force deforms the parts until their complementary surfaces mate up (Figure 3). The force required to mate two parts is usually much less than the force required to take them apart. Snap fasteners are already widely used, and they are particularly interesting for MEM structures because of the simple assembly process where one part remains fixed, and the other moves on a straight line towards the first part.

(Pai and Donald 1989; Donald and Pai 1989; Donald and Pai 1992) show that under widely used assumptions (see e.g. (Whitney 1982)), the assembly process with snap fasteners can be simulated and analyzed with purely algebraic methods, avoiding numerical integration of velocities over time. We follow this approach of algebraic, symbolic simulation.

3 Modeling

A model for the mechanism to design should have the following key properties:

- Allow fast and robust analysis and simulation of the mechanism.
- Keep track of correspondence between geometric properties and functional properties.
- Abstract away from geometric and physical details by finding classes of "qualitatively equivalent" design.
- Make it easy to compare different designs.

A model for snap fasteners has been developed in (Pai and Donald 1989; Donald and Pai 1989; Donald and Pai 1992). It differs from conventional models in that analysis and simulation of snap fastener mechanisms

(i) are purely algebraic, and hence exact, (ii) are combinatorially precise, in that the computational complexity is exactly known, and (iii) require no (numerical) integration of motion. This model is the basis for a similar model for hinged structures which will tackle the above items. We will describe it in the following subsections.

3.1 Modeling Assumptions

To model a hinged structure consisting of one plate-lock pair we make the following assumptions:

3.1.1 Assumptions on Geometry

- The parts are a finite union of rectangular polyhedra. This means that all cross sections are rectangular polygons. Furthermore a finite (usually small) number of cross sections completely describes the parts.
- The hinged plate has one rotational degree of freedom.
- The spring loaded lock has one rotational degree of freedom. This is a good model for torsional spring locks, and a first approximation for cantilevered beam springs, as they bend over their entire length.
- The thickness of the parts (perpendicular to the wafer plane) is small compared to their total length.

It follows that the configuration space of a hinged structure with one plate-lock pair is $C = (S^1)^2$. In fact, since the parts cannot rotate into the wafer, the configuration space reduces to $C = [0, \pi]^2$.

3.1.2 Assumptions on Physics

We make the following assumptions on the physics of body motion and interaction (for the first three points see also (Whitney 1982; Pai 1988; Donald 1990; Donald and Pai 1992)).

- Quasi-static motion: no inertial and Coriolis forces.
- Instantaneous snap: if the lock snaps off an edge of the plate, the time to reach contact again is neglectable.
- Coulomb friction.
- The hydraulic force (when the plates are moved in liquid) is proportional to the dot product of surface and liquid velocity.
- The spring force in a spring loaded lock is proportional to its vertical deviation from the wafer plane.

3.2 Geometric Model

Above we stated that the geometry of a hinged mechanism is completely specified by a list of cross sections. Therefore it is sufficient if we develop a model for the cross section of one plate-lock pair (Figure 4).

The cross sections of lock $\mathcal{L}$ and plate $\mathcal{P}$ are modeled by sequences of points L_i, $i = 0 \ldots n_{\mathcal{L}}$, and P_j, $j = 0 \ldots n_{\mathcal{P}}$. Let us look at the plate $\mathcal{P}$. P_0 lies closest to the center of rotation, and $P_{n_{\mathcal{P}}}$ is the point furthest from the center of rotation. We denote the distance (center of rotation — P_i) by p_i. The points P_i define an alternating sequence of filled and empty rectangles of the cross section of the plate, e.g. the rectangle between P_0 and P_1 is filled. Note that $n_{\mathcal{P}}$ is always odd, and that $p_0 \leq p_1 \leq \ldots \leq p_{n_{\mathcal{P}}}$. Finally, the angle between wafer plane and plate $\mathcal{P}$ is called $\phi_{\mathcal{P}}$.

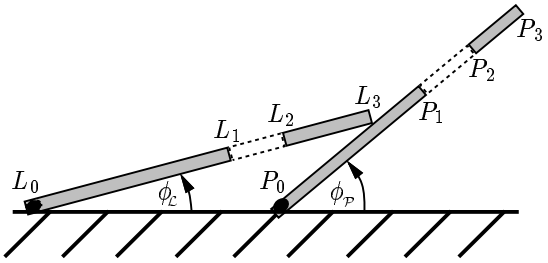

Figure 4: Cross Section Model

For the lock $\mathcal{L}$ we define analogous point sets L_i, distances l_i, where $i = 0 \ldots n_{\mathcal{L}}$, and angle $\phi_{\mathcal{L}}$. At last, we define Δ as the distance between the centers of rotation of $\mathcal{P}$ and $\mathcal{L}$.

Note that we reduce a three-dimensional mechanism to "$1\frac{1}{2}$D" extrusion structures. By looking at the cross section, we eliminate the direction perpendicular to the cross section. Finally, the assumption that the thickness of lock and plate is small compared to their lengths yields a one-dimensional model, such that each part is represented as an ordered list of line segments.

3.3 Dynamic Model

In this section we come up with a dynamic model for a hinged structure with one plate-lock pair. It will be used to deal with spring forces and forces due to liquid flow.

Let us define $\widehat{P}$ and $\widehat{L}$ as unit vectors in the direction of plate $\mathcal{P}$ and lock $\mathcal{L}$, respectively. Define $\widehat{P}_\perp$ and $\widehat{L}_\perp$ as vectors normal to $\widehat{P}$ and $\widehat{L}$ (rotated counterclockwise by $\pi/2$).

The plate $\mathcal{P}$ has one rotational degree of freedom $\phi_{\mathcal{P}}$. Due to liquid flow, $\mathcal{P}$ can exert a force $F_{\mathcal{P}}$. This force acts normal to $\mathcal{P}$, it is proportional to the dot product of the surface normal and the liquid velocity. More precisely:

$$
\begin{aligned}
|F_{\mathcal{P}}| = f_{\mathcal{P}} &= c_{\mathcal{P}} \langle A_{\mathcal{P}}, v \rangle \frac{\overline{p}}{p} \\
F_{\mathcal{P}} &= f_{\mathcal{P}} \widehat{P}_\perp
\end{aligned}
$$

where $c_{\mathcal{P}}$ is a positive constant, $A_{\mathcal{P}}$ is the normal vector to the plate such that $|A_{\mathcal{P}}|$ is the size of the plate surface, v is the velocity of the liquid, $\overline{p}$ is the distance between the rotational center of $\mathcal{P}$ and the force center, and p is the distance between the rotational center and the point where $F_{\mathcal{P}}$ is acting. $\langle \ldots, \ldots \rangle$ is the dot product.

Similarly, we model the lock $\mathcal{L}$ as a part with one rotational degree of freedom $\phi_{\mathcal{L}}$. Due to the spring the lock generates a force $F_{\mathcal{L}}$. This force is normal to the lock and proportional to the vertical deviation. More precisely:

$$\begin{aligned} |F_{\mathcal{L}}| = f_{\mathcal{L}} &= c_{\mathcal{L}} \sin\phi_{\mathcal{L}} \frac{1}{l} \\ F_{\mathcal{L}} &= f_{\mathcal{L}}(-\widehat{L}_{\perp}) \end{aligned}$$

where $c_{\mathcal{L}}$ is a positive constant, and l is the distance between the center of rotation of $\mathcal{L}$ and the point where $F_{\mathcal{L}}$ is acting.

3.3.1 Type-A Contact

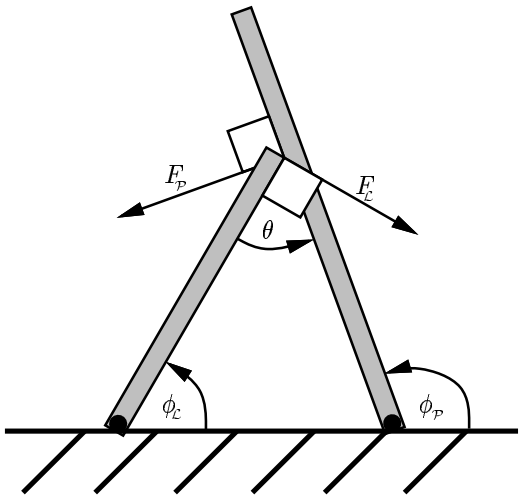

Figure 5: Forces at Type-A Contact

Assume that a vertex of the lock $\mathcal{L}$ is in contact with an edge of the plate $\mathcal{P}$ (type-A contact, see section 4.1) at angle $\theta = \phi_{\mathcal{P}} - \phi_{\mathcal{L}}$ (Figure 5) . To compute the forces and the conditions for sticking and sliding, we look at the Free Body Diagrams of plate and lock (Figure 6). Assume a force F acts on the plate in the contact point at an angle β to the plate. If the system is in equilibrium, then the component of F normal to the plate, $F_{\mathcal{P}_{\perp}}$, has to balance out the force due to liquid flow, $F_{\mathcal{P}}$. The component of F parallel to the plate, $F_{\mathcal{P}_{\parallel}}$, is a friction force.

A reaction force $-F$ has to act on the lock under an angle $\gamma = \pi - \theta - \beta$. For the system to be in equilibrium, its normal component, $F_{\mathcal{L}_{\perp}}$, has to balance the spring force $F_{\mathcal{L}}$.

We want to find the condition when lock and plate are in contact and sticking. For the parts to remain in contact, $F_{\mathcal{P}}$ has to be positive. The lock sticks when F lies in the friction cone:

$$\begin{aligned} f_{\mathcal{P}} &> 0 \\ \mu &\geq |\cot\beta| \end{aligned}$$

Now we compute the forces.

$$\begin{aligned} F_{\mathcal{P}_{\perp}} &= f\sin\beta(-\widehat{P}_{\perp}) \\ F_{\mathcal{P}_{\parallel}} &= -f\cos\beta\widehat{P} \\ F_{\mathcal{L}_{\perp}} &= f\sin\gamma\widehat{L}_{\perp} \\ &= f\sin(\theta+\beta)\widehat{L}_{\perp} \\ F_{\mathcal{L}_{\parallel}} &= f\cos\gamma(-\widehat{L}) \\ &= f\cos(\theta+\beta)\widehat{L} \end{aligned}$$

In equilibrium $F_{\mathcal{L}_{\perp}}$ is balanced by the spring force $F_{\mathcal{L}}$, $F_{\mathcal{P}_{\perp}}$ is balanced by the hydraulic force on the plate $F_{\mathcal{P}}$. $F_{\mathcal{L}_{\parallel}}$ and $F_{\mathcal{P}_{\parallel}}$ are balanced by the structure.

$$\begin{aligned} f_{\mathcal{P}} &= f\sin\beta \\ f_{\mathcal{L}} &= f\sin(\theta+\beta) \\ &= f\sin\theta\cos\beta + \cos\theta\sin\beta \\ \Rightarrow \frac{f_{\mathcal{L}}}{f_{\mathcal{P}}} &= \frac{\sin\theta\cos\beta + \cos\theta\sin\beta}{\sin\beta} \\ &= \sin\theta\cot\beta + \cos\theta \\ \Rightarrow \cot\beta &= \frac{f_{\mathcal{L}}/f_{\mathcal{P}} - \cos\theta}{\sin\theta} \end{aligned}$$

So finally we get as the condition for sticking:

$$f_{\mathcal{P}} > 0 \tag{1}$$

$$\mu \geq \left|\frac{f_{\mathcal{L}}/f_{\mathcal{P}} - \cos\theta}{\sin\theta}\right| \tag{2}$$

(2) tells us that sticking depends on the ratio $f_{\mathcal{L}}/f_{\mathcal{P}}$ and behaves somewhat similar to $\cot\theta$. There are several interesting cases:

$f_{\mathcal{P}} \to 0$: Then (2) $\to \infty$, so if $F_{\mathcal{P}}$ becomes small, then the area of sticking vanishes.

$f_{\mathcal{L}} \to 0$: Then (2) $\to \cot\theta$. This is intuitively clear, because we only have a force normal to the surface.

$\theta = \pi/2$: Then (2) $= f_{\mathcal{L}}/f_{\mathcal{P}}$, the ratio of parallel to normal force.

$\theta \to 0, \pi$: Then (2) $\to \infty$, so no sticking occurs if lock and plate are parallel unless $f_{\mathcal{P}} \approx f_{\mathcal{L}}$.

Plugging in the force magnitudes we get:

$$\langle A_{\mathcal{P}}, v\rangle > 0 \tag{3}$$

$$\mu \geq \left|\frac{\frac{c_{\mathcal{L}}\sin\phi_{\mathcal{L}}\,p}{c_{\mathcal{P}}\langle A_{\mathcal{P}}, v\rangle\overline{pl}} - \cos\theta}{\sin\theta}\right| \tag{4}$$

Note: We were assuming a contact between a vertex of the lock and an "inside" edge of the plate, i.e. and edge of the plate on the side of the lock. It is also possible to have a contact with an "outside" edge of the plate. Then the condition for contact (1) changes sign.

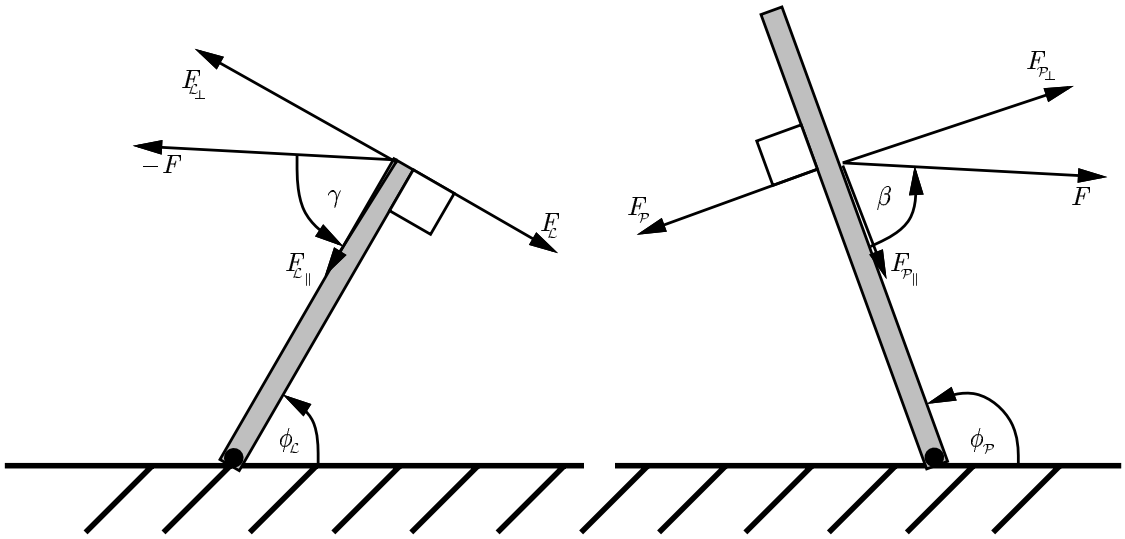

Figure 6: Free Body Diagrams

3.3.2 Type-B Contact

Assume that a vertex of the plate is in contact with an edge of the lock (Type-B contact) at angle $\theta = \phi_{\mathcal{P}} - \phi_{\mathcal{L}}$. By a similar argument as for Type-A contact, we get as the condition for sticking:

$$f_{\mathcal{L}} > 0 \tag{5}$$

$$\mu \geq \left| \frac{f_{\mathcal{P}}/f_{\mathcal{L}} - \cos\theta}{\sin\theta} \right| \tag{6}$$

(5) is always satisfied, because we are assuming a positive spring constant. So finally we get:

$$\mu \geq \left| \frac{\frac{c_{\mathcal{P}}\langle A_{\mathcal{P}}, v\rangle \overline{p} l}{c_{\mathcal{L}} \sin\phi_{\mathcal{L}}\, p} - \cos\theta}{\sin\theta} \right| \tag{7}$$

Note: The above condition is for "inside" edges of the lock. For an "outside" edge (5) has to be negated. The negated equation is never satisfied. Therefore sticking at Type-B contact can only occur with "inside" edges.

4 Approach

4.1 C-Space of Model

In the model for rigid bodies in compliant contact described in (Donald and Pai 1992), contacts of mechanism parts define curves in their two-dimensional configuration space. These curves are algebraic of low-degree. Simulation of snap fasteners then is reduced to computing the arrangement of curves in the C-space plane. This yields an efficient simulation algorithm and avoids numerical integration of motion.

Our hinged structure has two rotational degrees of freedom. This makes the curves that describe the contact constraints more complicated. A simplification to algebraic curves as in (Donald and Pai 1992) is possible, but leads to polynomial curves of high degree, so the approach is not applicable. On the other hand, we observe that in the geometry of the mechanism model there is a lot of structure that can be exploited to find a simple description of C-space curves. Consider a configuration where lock $\mathcal{L}$ and plate $\mathcal{P}$ are in contact, the contact points being at l and p distance units from the center of rotation, respectively. Then we get the following two conditions (see Figure 4 again):

$$\begin{aligned} l\sin\phi_{\mathcal{L}} &= p\sin\phi_{\mathcal{P}} \\ l\cos\phi_{\mathcal{L}} - p\cos\phi_{\mathcal{P}} &= \Delta \end{aligned}$$

Squaring, we get

$$\begin{aligned} l^2 - l^2\cos^2\phi_{\mathcal{L}} &= p^2 - p^2\cos^2\phi_{\mathcal{P}} \\ (l\cos\phi_{\mathcal{L}} - \Delta)^2 &= p^2\cos^2\phi_{\mathcal{P}} \end{aligned}$$

Eliminating $\cos\phi_{\mathcal{P}}$,

$$\begin{aligned} (l\cos\phi_{\mathcal{L}} - \Delta)^2 &= p^2 - l^2 + l^2\cos^2\phi_{\mathcal{L}} \\ \Rightarrow \cos\phi_{\mathcal{L}} &= \frac{l^2 - p^2 + \Delta^2}{2l\Delta} \end{aligned} \tag{8}$$

And similarly,

$$\cos\phi_{\mathcal{P}} = \frac{l^2 - p^2 - \Delta^2}{2p\Delta} \tag{9}$$

Note that $\phi \mapsto \cos\phi$ is a one-to-one map on $[0, \pi]$. The C-space of our mechanism maps one-to-one into the square $[-1, 1] \times [-1, 1]$. So by taking the cosines

of the angles $\phi_{\mathcal{P}}$ and $\phi_{\mathcal{L}}$ as arguments to the C-space curves, we obtain a purely algebraic representation without trigonometric functions. Also, no trigonometrical functions are necessary to e.g. sort configurations with respect to $\cos\phi_{\mathcal{P}}$ (and thus $\phi_{\mathcal{P}}$).

Let's consider the different possibilities of contact:

1. **Vertex-edge contact:** A vertex l_i of the lock is sliding along an edge (p_j, p_{j+1}) of the plate (type-A contact, following the convention of (Lozano-Pérez 1983; Canny 1986; Donald 1987), as in Figure 4).[1] Using equations 8 and 9 we get the following parametrization of the corresponding C-space curve:

$$(\cos\phi_{\mathcal{P}}, \cos\phi_{\mathcal{L}}) = (\frac{l_i^2 - p^2 - \Delta^2}{2p\Delta}, \frac{l_i^2 - p^2 + \Delta^2}{2l_i\Delta}),$$

$$\text{where } p \in [p_j, p_{j+1}]$$

2. **Edge-vertex contact:** A vertex p_j of the plate is sliding along an edge (l_i, l_{i+1}) of the lock (type-B contact):

$$(\cos\phi_{\mathcal{P}}, \cos\phi_{\mathcal{L}}) = (\frac{l^2 - p_j^2 - \Delta^2}{2p_j\Delta}, \frac{l^2 - p_j^2 + \Delta^2}{2l\Delta}),$$

$$\text{where } l \in [l_i, l_{i+1}]$$

3. **Edge-edge contact:** This happens only at angles

$$(\phi_{\mathcal{P}}, \phi_{\mathcal{L}}) = (0,0), (0,\pi), (\pi,0), (\pi,\pi)$$

or, equivalently

$$(\cos\phi_{\mathcal{P}}, \cos\phi_{\mathcal{L}}) = (1,1), (1,-1), (-1,1), (-1,-1)$$

4. **Vertex-vertex contact:** This is the intersection of a vertex-edge contact and an edge-vertex contact within a cross section. A vertex l_i touches a vertex p_j of the plate:

$$(\cos\phi_{\mathcal{P}}, \cos\phi_{\mathcal{L}}) = (\frac{l_i^2 - p_j^2 - \Delta^2}{2p_j\Delta}, \frac{l_i^2 - p_j^2 + \Delta^2}{2l_i\Delta})$$

5. **Vertex-edge and edge-vertex contact:** This is the intersection of a vertex-edge contact and an edge-vertex contact in different cross sections. A mechanism with more than one cross section may have configurations in which one cross section is in vertex-edge contact, and another cross section is in edge-vertex contact. The formula for this case is the same as for vertex-vertex contacts. We call this case "type-AB."

[1] Note that we refer to vertices and edges in the cross section model. A vertex in the model corresponds to an edge in the mechanism. So a type-A contact occurs when an edge of the lock touches a surface of the plate. Vertex-edge or vertex-plate contacts do not occur in the mechanism, assuming rectangular part geometry.

4.2 Example: A Hinged Structure

Consider the hinged structure in Figure 7. It consists of two cross sections which are shown separately in Figures 9 and 11. Their respective configuration spaces are shown in Figures 8, 10, and 12. The vertices and edges in the cross section model are numbered outward from the rotational center, from front to back. In Figures 13 to 19 all possible edge-edge, vertex-vertex, and type-AB configurations of the hinged mechanism are shown. All pictures were generated automatically (see section 4.6).

4.3 C-Space Graph

As we have seen in the previous section, the behavior of the system can be described with two kinds of states. We distinguish zero-dimensional *events* (edge-edge, vertex-vertex, type-AB contacts) and one-dimensional *transitions* (edge-vertex, vertex-edge contacts). Using events and transitions, we can map the C-space of the mechanism to a graph (C-space graph) whose nodes are states and transitions, and whose edges reflect the adjacency relationships of the nodes. This graph can be seen as a discrete map of the C-space, where curve segments map into transition nodes, curve endpoints map into event nodes, and adjacency relationships are preserved.

Note that the C-space graph is bipartite with respect to events and transitions. For a given hinged structure the number of nodes is $O(n_{\mathcal{P}} n_{\mathcal{L}})$ if the plate has $n_{\mathcal{P}}$ vertices, and the lock has $n_{\mathcal{L}}$ vertices. The graph is sparse, because the number of outgoing edges for each state is bounded by 2, one each for moving clockwise or counterclockwise on the C-space curve infinitesimally.

The C-space graph in our model for hinged structures is the equivalent to the arrangement of C-space curves in (Donald and Pai 1992). However, in the C-space graph only the intersections of the curves are computed explicitly. Each curve segment is described by a single transition node. This means no loss of information, because all points on this curve segment obey the same equations. Thus each node of the C-space graph can be seen as a class of equivalent behavior of the mechanism.

Each cross section of a mechanism generates a separate C-space, and thus a C-space graph. The free space of the entire mechanism is the "geometric AND" of the free space of all cross sections. Intersections of C-space curves specify configurations where parts in two cross sections are in contact. If the mechanism's configuration moves through such an intersection point, the contact changes from one cross section to another. In our model this happens only in type-AB nodes. Figures 21 and 22 show the C-space graphs of front and back part of the mechanism described earlier in section 4.2. Events are shown as rectangles, transitions as diamonds. The graphs look somewhat trivial, but in fact for a two-dimensional C-space only linear graphs and simple loops are possible. For a three-dimensional C-space all C-space graphs are planar.

The combined graph for both cross sections is shown

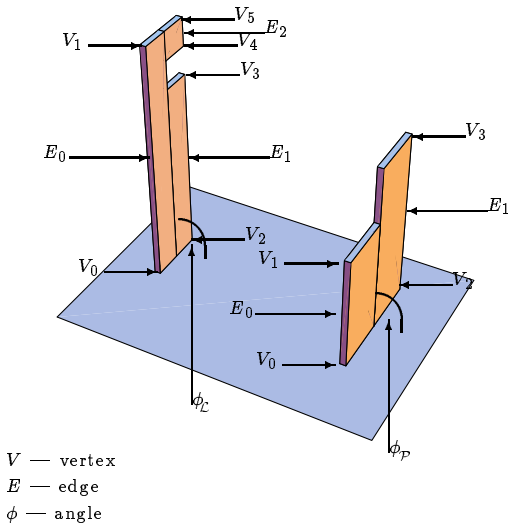

Figure 7: A Hinged Structure

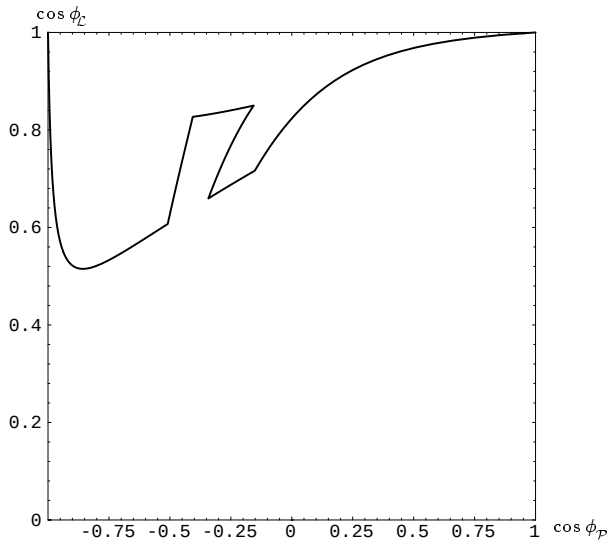

Figure 8: C-Space of Hinged Structure

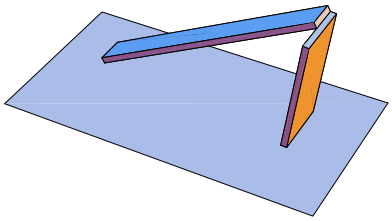

Figure 9: V_1-V_1 (Vertex-Vertex) Contact of Front Part

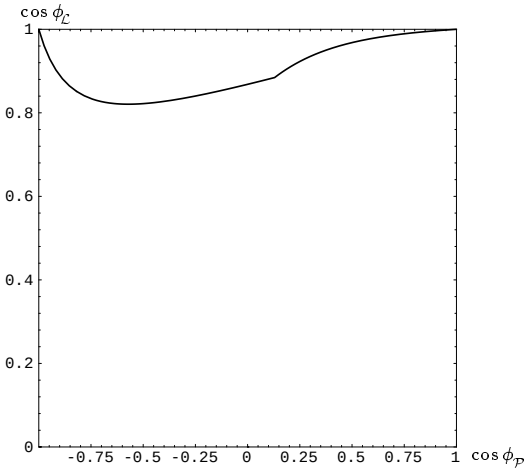

Figure 10: C-Space of Front Part

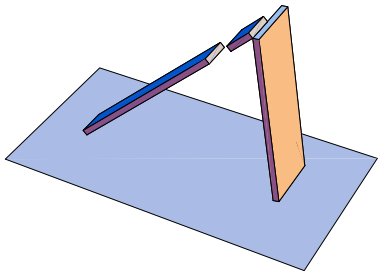

Figure 11: V_5-V_3 (Vertex-Vertex) Contact of Back Part

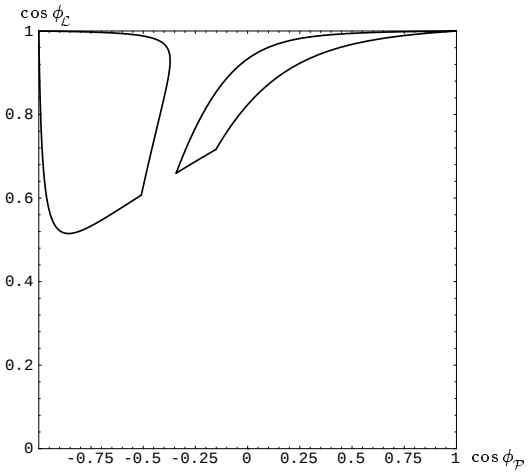

Figure 12: C-Space of Back Part

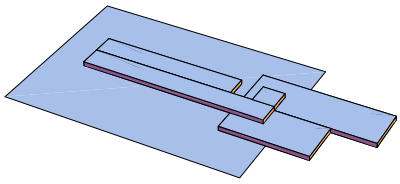

Figure 13: E_2-E_1 (Edge-Edge) Contact

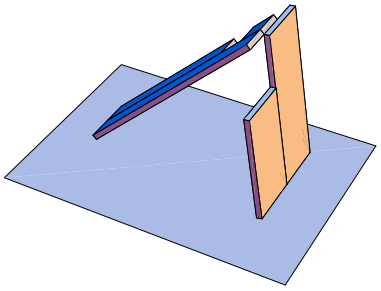

Figure 14: V_5-V_3 Contact

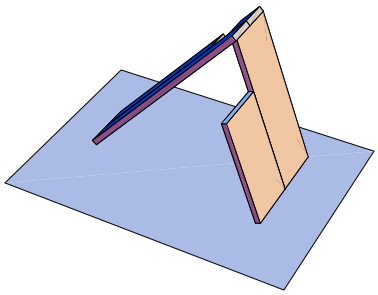

Figure 15: V_4-V_3 Contact

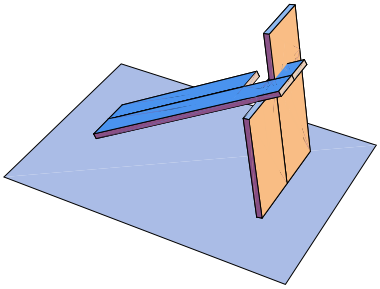

Figure 16: V_4-E_1/E_0-V_1 (Type-AB) Contact

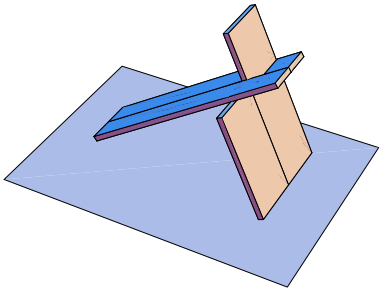

Figure 17: V_3-E_1/E_0-V_1 (Type-AB) Contact

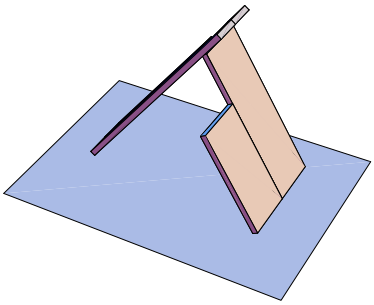

Figure 18: V_3-V_3 Contact

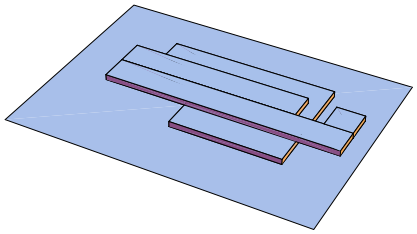

Figure 19: E_1-E_1 Contact

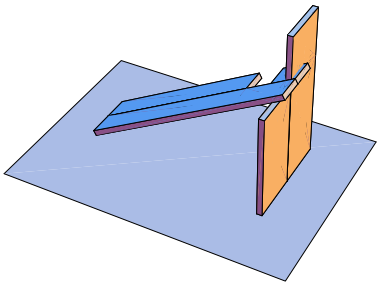

Figure 20: Illegal Configuration

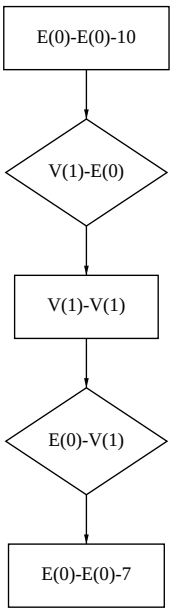

Figure 21: C-Space Graph of Front Part

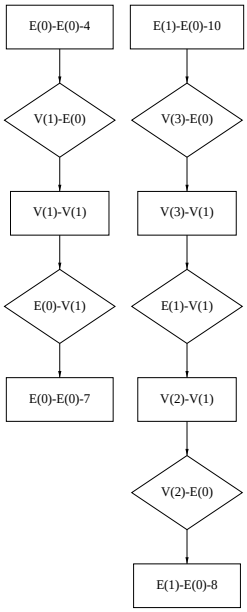

Figure 22: C-Space Graph of Back Part

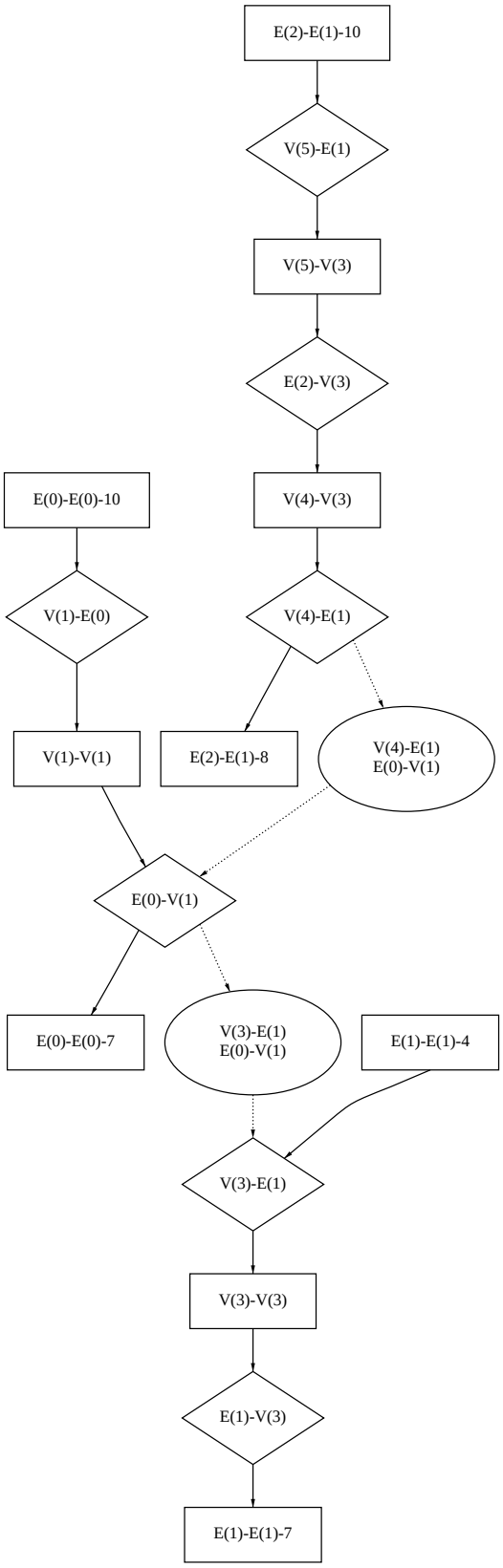

Figure 23: C-Space Graph of Hinged Mechanism

in Figure 23. Ellipses denote type-AB events, their edges to transitions (shown as dotted arrows) "dominate" edges within the C-space graph of the cross section, i.e. if the mechanism is in motion, it will follow the transition denoted by the dotted line, due to kinematic constraints between different cross sections. Recall that whenever the mechanism goes through such a transition, the contact region changes from one cross section to another.

Therefore some nodes in the combined graph become unreachable. Figure 20 shows a configuration where edge E_0 of the lock is in contact with vertex V_1 of the plate. This configuration would be possible for the front cross section, but is impossible because the parts in the back cross section would overlap. In Figure 23 we can see that when going counterclockwise (see below) from node "E_0-V_1" the edge to type-AB node "V_4-E_1/E_0-V_1" dominates the edge to node "V_1-V_1."

Some more notes on the C-space graphs in Figures 21 to 23.

- Edges in the graph are directed. Moving in direction of an arc corresponds moving clockwise around a C-space obstacle, i.e. with the obstacle to the right.
- For each edge-edge contact, we distinguish between different possible configurations. This is necessary because we neglect the vertical thickness of lock $\mathcal{L}$ and plate $\mathcal{P}$, so there may be a contact on either side of an edge, depending on whether $\mathcal{L}$ or $\mathcal{P}$ are at angles $(\phi_{\mathcal{P}}, \phi_{\mathcal{L}}) = (0, 0)$, $(0, \pi/2)$, or $(\pi/2, \pi/2)$, and whether $\mathcal{L}$ is above or below $\mathcal{P}$.
- Several extensions for this graph are possible. In particular, one could imagine a labeling of nodes and edges. These labels could include information about dynamics (see section 3.3) or probabilities of state changes (see section 5).

4.4 Parameterized C-Space Graphs

We are now ready to get back to the original problem of mechanism design. Consider a simple example. Let's assume that the mechanism in Figures 13 to 19 has one design parameter x that determines the length of the front part of the plate. I.e. x is the length of the plate (the right part) in Figure 9. The C-space and therefore also the C-space graph of the parameterized mechanism will depend on this design parameter. More specifically, many nodes and edges will only exist for certain ranges of values for x. Figure 24 shows the C-space graph for the parameterized mechanism. All nodes whose existence depends on x have their conditions on the design parameter x attached. For our model of hinged structures, all constraints are conjunctions of linear inequalities.

Alternatively the design parameters could be viewed as generating a generalized configuration space (Pai 1988; Donald 1989) with one additional dimension for each design parameter, and a family of C-space graphs. Figure 25 shows the generalized C-space for the parameterized mechanism mentioned above. The design parameter x corresponds to the vertical axis. Each

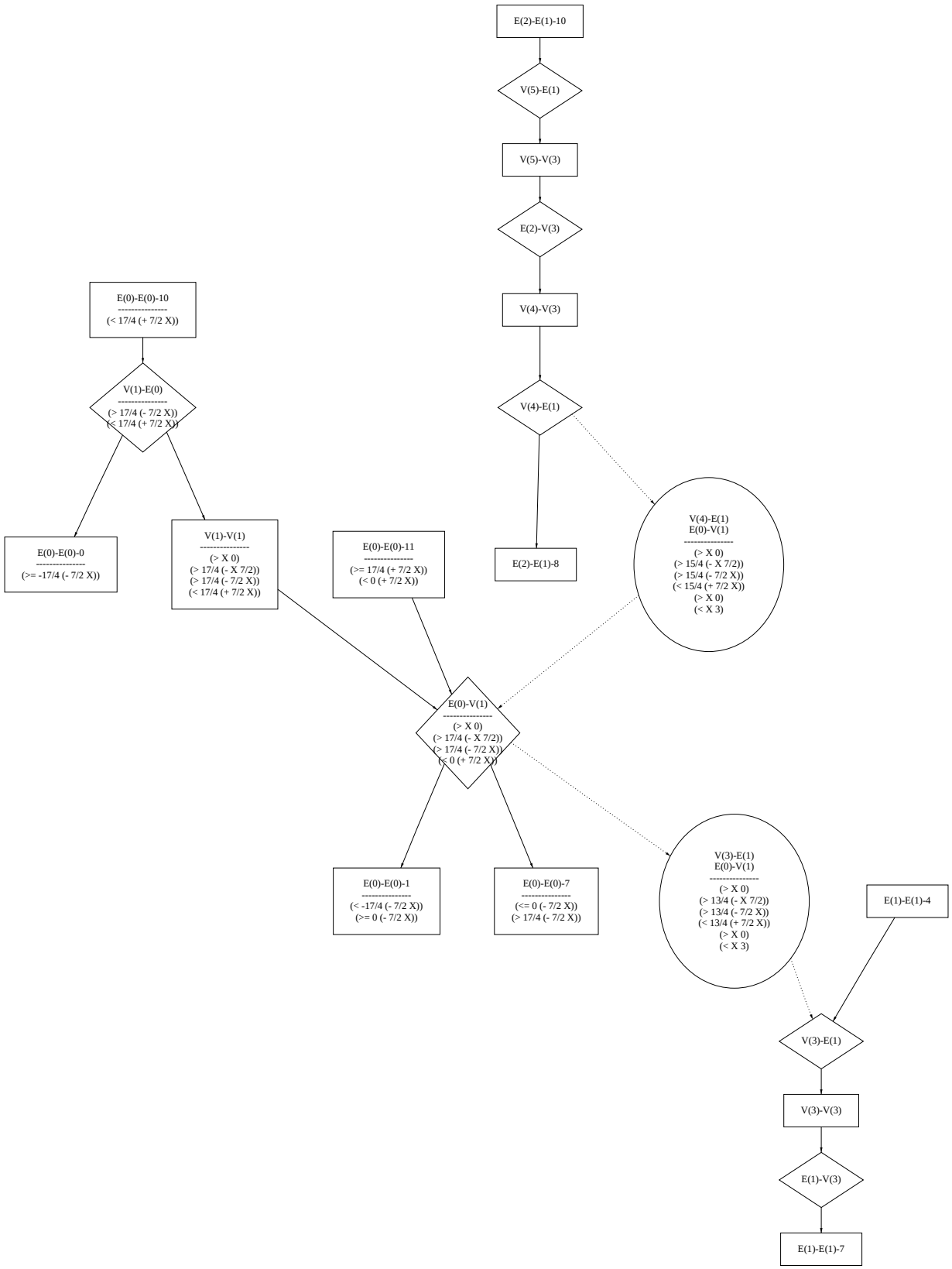

Figure 24: Parameterized C-Space Graph of Hinged Mechanism

slice in the generalized C-space gives the C-space for a particular parameter value, and each C-space graph represents the kinematic states of the graph for a particular range of design parameter values. However, in the following we prefer a single parameterized C-space graph with conditions attached to nodes and edges.

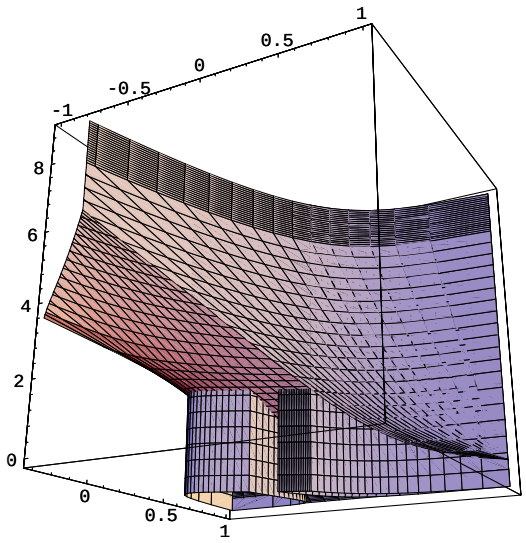

Figure 25: Generalized C-Space of Hinged Mechanism

Note that now the graph looks nondeterministic, in that nodes can have multiple outgoing and incoming edges, corresponding to multiple states as their clockwise or counterclockwise neighbors. Of course this is not really an indeterminacy, because by replacing x with any specific value we would get a deterministic graph. Depending on the value of x the mechanism has different states, and therefore exhibits different behaviors. The parameterized C-space graph in Figure 24 are a concise representation for the correlation between parameter values and mechanism behavior.

4.5 Design Algorithm Outline

Suppose that the required behavior of a parameterized mechanism can be specified by a set of desired states. Then an algorithm that "unifies" this set with a C-space subgraph of the parameterized mechanism can determine the constraints on the design parameters.

In the above example, a design objective would be a mechanism that has a configuration in which the motion of the plate is constrained such that it stands up approximately vertically. This configuration is achieved in the diamond "E_0-V_1" that is bounded by the two ellipses "V_4-E_1/E_0-V_1" and "V_3-E_1/E_0-V_1" in Figure 24 (see also Figures 16 and 17 again). As can be determined from the constraints in Figure 24, x is required to lie in $(\frac{1}{4}, 3)$. This makes sense, because if $x \geq 3$ then the front part of the plate is not shorter than the back part, and the plate cannot fit into the slot in the lock. If $x \leq \frac{1}{4}$ then the plate fits into the slot in the lock, but the plate's rotational degree of freedom is not constrained. It is exactly for parameter values of $x \in (\frac{1}{4}, 3)$ that we get back the graph of Figure 23.

4.6 Implementation

We have started a LISP implementation of the above ideas. The current program computes the C-space graph for a hinged mechanism. In particular, it generates input for the graph drawing program DAG (Gansner et al. 1988). It can also generate input for MATHEMATICA (Wolfram 1991) to draw the 2D configuration space of the mechanism, and 3D pictures of any configuration. MATHEMATICA is an interactive system for mathematical computation. Figures 7 to 24 were generated by the program.

For example, this is the LISP representation of the sample hinged structure from section 4.2:

```
(make-mechanism
  (make-part
  `(,(make-rectangle
      (make-interval 0 17/4)
      (make-interval 1/2 1))
    ,(make-rectangle
      (make-interval 0 13/4)
      (make-interval 1 3/2))
    ,(make-rectangle
      (make-interval 15/4 17/4)
      (make-interval 1 3/2))))
  (make-part
  `(,(make-rectangle
      (make-interval 0 (make-parameter x))
      (make-interval 0 1))
    ,(make-rectangle
      (make-interval 0 3)
      (make-interval 1 2))))
  7/2)
```

In general, a mechanism is defined by two parts (lock $\mathcal{L}$ and plate $\mathcal{P}$), and their relative distance Δ. Each part consists of a list of rectangles. The dimensions are specified either by numbers or by symbolic values (design parameters).

The algorithm to construct the C-space graph from the geometric description of a hinged structure is shown here:

input: Geometric description of hinged structure $(\mathcal{L}, \mathcal{P}, \Delta)$.
output: Its corresponding C-space graph.

for each l in the set of vertices and edges of lock $\mathcal{L}$ **do**
  for each p in the set of vertices and edges of plate $\mathcal{P}$ **do**
    if there is a configuration for which l and p are in contact
    then (1) create node(s) for pair (l, p)
      (2) compute the adjacencies for new node(s)
    end if
  end for
end for

For the complexity analysis we assume that the lock $\mathcal{L}$ has $n_{\mathcal{L}}$ vertices, and the plate $\mathcal{P}$ has $n_{\mathcal{P}}$ vertices. Let us first consider the case without parameters. In a single cross section of the mechanism, each node (events: vertex-vertex, edge-edge; transitions: type-A, type-B) has at most two neighbors. Events and transitions form a bipartite graph. Graphs of different cross sections are connected via type-AB nodes, which in turn have two neighbors. From this it follows that steps (1) and (2) in the algorithm above can be computed in $O(1)$ time. $O(n_{\mathcal{L}} n_{\mathcal{P}})$ is the bound for computing the C-space graph, and it is also the bound for the number of nodes and edges in the graph. Therefore the graph is sparse.

Now consider the case when the geometry is parameterized. Define $\widehat{n}_{\mathcal{L}}$ ($\widehat{n}_{\mathcal{P}}$) as the maximum number of lock (plate) vertices in any cross section. Then for each vertex pair (l, p) there may be up to $O(\widehat{n}_{\mathcal{L}} \widehat{n}_{\mathcal{P}})$ type-AB nodes, because there can be $O(\widehat{n}_{\mathcal{P}})$ plate edges that form a type-A contact with l, and $O(\widehat{n}_{\mathcal{L}})$ lock edges that form a type-B contact with p. Therefore steps (1) and (2) have $O(\widehat{n}_{\mathcal{L}} \widehat{n}_{\mathcal{P}})$ worst running time. This implies $O(n_{\mathcal{L}} \widehat{n}_{\mathcal{L}} n_{\mathcal{P}} \widehat{n}_{\mathcal{P}}) \leq O(n_{\mathcal{L}}^2 n_{\mathcal{P}}^2)$ as time bound for constructing the graph, and as space bound for the number of vertices and edges. Note that the introduction of design parameters does not affect the sparseness of the C-space graph.

5 Randomized Assembly

Randomized strategies for assembly have been proposed by many authors. They are used for example in vibratory bowl feeders (Boothroyd et al. 1977; Boothroyd et al. 1982) to align parts for an assembly. Parts that are not properly aligned after a randomized orientation phase are filtered out and oriented again. (Lozano-Pérez 1985) develops ideas for an automatic design of these filtering devices.

(Moncevicz et al. 1991) come up with the term of "shake-and-make assembly" as an ideal assembly process where devices are assembled by agitated random motion. For this process to work, the completed assembly has to have certain properties. E.g. it should have a particularly low entropy relative to all possible configurations, or the assembly action should be irreversible.

(Erdmann 1989; Erdmann 1992) gives a general approach for probabilistic strategies in robot tasks. The basic idea is to use randomized motions whenever the uncertainty in the sensor readings is too large to be of any use for motion planning. This is of particular interest for the assembly of microelectromechanical structures. In fact, (Burgett et al. 1992) use a randomized strategy to assemble the hinged structures described in section 2.1, by shaking the wafer under water such that the random motion of the water moves the plates into their final position.

(Erdmann 1989) investigates randomized strategies for both discrete and continuous state spaces. For the discrete case, he shows that the expected time of goal attainment in randomized tasks is low if the state space of the mechanism can be modeled by Markov chains. This creates a promising connection to C-space graphs. If we label C-space graph edges with probabilities of being traversed, then we can apply the theory developed in (Erdmann 1989) to predict the behavior of the mechanism. In particular, we can compute the probability that a certain desired state is reached. The crucial task now is to determine these probabilities. They will depend on both kinematic and dynamic properties of the mechanism.

6 Summary

We have outlined an approach for automation of parametric design. A model suited for analysis, simulation, and design of microelectromechanical hinged structures was developed. The model compresses the configuration space of a mechanism into a concise description of its kinematics, the C-space graph. Further processing, incorporating dynamics and randomization, will yield a precise description of the mechanism's functionality. The resulting graph will possess the following properties:

- It gives a concise description of the mechanism behavior.
- It keeps track of the correspondence between geometric features and functionality of the mechanism.
- It forms the basis for algorithms that automate parametric design, by making explicit the relationship between design parameter values and modes of mechanism operation.
- It provides a framework for analysis of random assembly strategies.

We believe that our approach, though still in an early stage, will advance design automation for useful subclasses of design problems. Models and techniques that allow fast, exact analysis and simulation of the mechanism to design are crucial for algorithms that search for good designs. Randomization techniques will extend the possibilities of algorithms for analysis, simulation, and design of mechanisms. Microelectromechanical structures offer a domain in which these techniques are extremely useful.

7 Future Goals

At the current stage of our work we want to focus on the following problems:

- Extend the model by incorporating mechanism dynamics, such as spring forces, and forces of liquid flow. This will complete the model of MEM hinged structures.
- Come up with a detailed algorithm (including complexity analysis) for "unifying" C-space subgraphs. This will form the crucial part of the proposed design algorithm for MEM hinged structures.
- This approach was developed for mechanisms with two-dimensional C-spaces. Investigate extensions for C-spaces with higher dimensions.
- Incorporate randomization techniques into the model. This will require analyzing the probabilistic behavior of the mechanisms. The result will be predictions on the probabilities for specific mechanism behaviors, and probabilities and expected times for reaching specific mechanism states.
- Extend the current implementation to include the items from above.
- Test the designed mechanisms by building MEM structures.

Acknowledgements

This paper describes research done in the Robotics and Vision Laboratory and the Computer Science Department at Cornell University. Support for our robotics research is provided in part by the National Science Foundation under grants No. IRI-8802390, IRI-9000532 and by a Presidential Young Investigator award to Bruce Donald, and in part by the Air Force Office of Sponsored Research, the Mathematical Sciences Institute, Intel Corporation, and AT&T Bell Laboratories.

Bruce Donald, Srikanth Kannapan (Xerox Design Research Institute), and Kris Pister (Berkeley) were of substantial help in writing this paper. Thanks also to Vivek Bhatt for his help with Free Body Diagrams.

References

Boothroyd, G.; Poli, C. R.; and Murch, L. E. 1977. *Handbook of Feeding and Orienting Techniques for Small Parts.* University of Massachusetts, Amherst, MA. updated July 1981.

Boothroyd, G.; Poli, C. R.; and Murch, L. E. 1982. *Automatic Assembly.* Marcel Dekker, Inc., New York.

Brooks, Rodney A. and Lozano-Pérez, Tomás 1982. A subdivision algorithm in configuration space for findpath with rotation. Technical Report A.I. Memo No. 684, Massachusetts Institute of Technology.

Brost, Randy C. 1989. Computing metric and topological properties of configuration space obstacles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Scottsdale, AZ.

Brost, Randy C. 1991. Computing the possible rest configurations of two interacting polygons. In *Proceedings of the IEEE International Conference on Robotics and Automation.*

Burgett, S.R.; Pister, K.S.J.; and Fearing, R.S. 1992. Three dimensional structures made with microfabricated hinges. In *Micromechanics.* ASME Winter Annual Meeting, Anaheim CA.

Canny, John 1986. Collision detection for moving polyhedra. *IEEE Transactions PAMI* 8(2).

Donald, Bruce R. and Pai, Dinesh K. 1989. On the motion of compliantly-connected rigid bodies in contact, part ii: A system for analyzing designs for assembly. Technical Report TR 89–1048, Cornell University, Department of Computer Science, Ithaca, NY 14853–7501.

Donald, Bruce R. and Pai, Dinesh 1992. Symbolic methods for the simulation of planar mechanical systems in design. In Donald, B.; Kapur, D.; and Mundy, J., editors 1992, *Symbolic and Numerical Computation for Artificial Intelligence.* Academic Press, Cambridge.

Donald, Bruce R. 1987. A search algorithm for motion planning with six degrees of freedom. *Artificial Intelligence* 31(3).

Donald, Bruce R. 1989. *Error Detection and Recovery in Robotics*, volume 336 of *Lecture Notes in Computer Science.* Springer Verlag, Berlin.

Donald, Bruce R. 1990. Planning multi-step error detection and recovery strategies. *International Journal of Robotics Research* 5(1):3–60.

Erdmann, M. A. 1989. *On Probabilistic Strategies for Robot Tasks.* Ph.D. Dissertation, Massachusetts Institute of Technology, Department of Computer Science.

Erdmann, M. A. 1992. Randomization in robot tasks. *IJRR.*

Gansner, E.R.; North, S. C.; and Vo, K. P. 1988. DAG: A program that draws directed graphs. *Software—Practice and Experience* 18(11):1047–1062.

Joskowicz, Leo 1989. Simplification and abstraction of kinematic behaviors. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, Detroit. 1337–1342.

Joskowicz, Leo 1990. Mechanism comparison and classification in design. *Research in Engineering Design* 1:149–166.

Journal of Microelectromechanical Systems. A Joint IEEE and ASME Publication on Microstructures, Microactuators, Microsensors, and Microsystems.

Kannapan, Srikanth M. and Marshek, Kurt M. 1991. Engineering design methodologies: A new perspective. Technical report, The University of Texas at Austin, Austin, Texas 78712–1063.

Lozano-Pérez, Tomás 1983. Spacial planning: A configuration space approach. *IEEE Transactions on Computers* C-32(2):108–120.

Lozano-Pérez, Tomás 1985. Towards the automatic design of orienting devices for vibratory part feeders. Working draft, Massachusetts Institute of Technology, Artificial Intelligence Laboratory.

Moncevicz, Paul H. Jr.; Jakiela, Mark J.; and Ulrich, Karl T. 1991. Orientation and insertion of randomly presented parts using vibratory agitation. In *Proceedings of the ASME Flexible Assembly Systems Conference*, Miami, Florida.

Pai, Dinesh K. and Donald, Bruce R. 1989. On the motion of compliantly-connected rigid bodies in contact, part i: The motion prediction problem. Technical Report TR 89–1047, Cornell University, Department of Computer Science, Ithaca, NY 14853–7501.

Pai, Dinesh K. 1988. *Singularity, Uncertainty and Compliance of Robot Manipulators.* Ph.D. Dissertation, Cornell University, Ithaca, NY.

Pister, K.S.J.; Judy, M.W.; Burgett, S.R.; and Fearing, R.S. 1992. Microfabricated hinges. *SAA* 33(3):249–256.

Whitney, D. E. 1982. Quasi-static assembly of compliantly supported rigid parts. *Journal of Dynamic Systems, Measurement, and Control* 104:65–77.

Wolfram, S. 1991. *Mathematica: A System for Doing Mathematics by Computer.* Addison-Wesley, second edition.