1991

Mechanical Parts Orienting: The Case of a Polyhedron on a Table

Michael Erdmann

Matthew T. Mason

George Vaněček

Report Number:

91-004

MECHANICAL PARTS ORIENTING: THE CASE OF A POLYHEDRON ON A TABLE

Michael Erdmann
Matthew T. Mason
George Vanecek, Jr.

CSD-TR-91-004
January 1991

Mechanical Parts Orienting:

The Case of a Polyhedron on a Table [1]

January 22, 1991

Michael Erdmann[2] Matthew T. Mason[3] George Vaněček, Jr.[4]

Abstract

The positioning and orienting of parts is a standard problem in manufacturing. Orienting parts is often a prelude to the assembly of parts at tight tolerances. This paper considers the problem of orienting a part resting on a table, by tilting the table. The initial orientation of the part is assumed to be completely unknown. The objective is to tilt the table in a manner that reduces the uncertainty in the part's orientation. The paper focuses on three-dimensional polyhedral parts, with infinite friction between the parts and the table. The paper proposes a planner that determines a sequence of tilting operations designed to minimize the uncertainty in the part's orientation. The planner runs in time $O(n^4)$, where n is the number of faces of the polyhedron. The planner produces a sequence of $O(n)$ distinct tilts.

[1] This research was supported, through the Leonardo Fibonacci Institute, by the Istituto Trentino di Cultura, Trento, Italy.

[2] School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15217. Additional support for Erdmann was provided by Carnegie Mellon University.

[3] School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15217.

[4] Department of Computer Science, Purdue University, West Lafayette, IN 47907. Additional support for Vaněček was provided by NSF grant CCR-86-19817 to Purdue University.

1 Introduction

The positioning and orienting of parts is a key problem in manufacturing and assembly. The localization of parts is often a prelude to more complicated assembly operations in which several parts must be mated at precise tolerances. There are two basic approaches for determining the position and orientation of a part. One is to sense this information, as with a vision system. Another approach is to use mechanical means for reorienting a part. The purpose is to act on the part in a manner that reduces uncertainty. Uncertainty is reduced whenever an action decreases the set of possible configurations of the part. In general, a manufacturing system will include both sensing and mechanical modules.

The mechanical approach towards parts localization underlies the use of feeder mechanisms, such as bowl feeders and conveyor belts. In these systems parts are forced to slide past a series of gates. Each gate either reorients a part or filters it out if the part arrives in an undesirable orientation. A coin sorter is a good example.

The mechanical approach has been most effective for localizing large quantities of parts in parallel. In part this is due to the long time required to sense precisely the location of a part using a vision system, then to apply some conditional manipulation operation that properly reorients the part. Additionally, with large numbers of parts, it is often not necessary to ensure that any particular part is properly oriented, merely that a large fraction of the parts entering the system are localized efficiently. Those that are not successfully reoriented are merely ejected from the current part of the orienting system, and fed back into the system at the beginning.

Humans perform much of the design of feeder mechanisms, relying on years of experience and intuition. One goal of an automated design and manufacturing system should be to automate as well the design of the hardware and software used to assemble the parts.

One type of orienting system in industrial use is a palletizing tray that contains depressions shaped approximately like the parts to be oriented. Parts are dropped onto the tray, the tray is made to shake for a while, and eventually the parts fall into their nests in desired orientations. One important question is how to design the parts and the nests so that the parts do in fact wind up in the desired orientations. Another important question is to decide how the tray should be moved in order to facilitate the orienting process.

With such a palletizing system as motivation, we consider in this paper the problem of orienting a three-dimensional polyhedron resting on a tiltable table. Friction is assumed to be infinite. An understanding of this problem is seen as a first step in solving the general nest-orienting problem. We derive an algorithm for obtaining a sequence of tilting operations of the table that minimizes the possible resting configurations of the polyhedron. We show that any such strategy contains at most $O(n)$ steps, where n is the number of faces of the polyhedron.

2 Previous Work

Previous work (Erdmann and Mason 1987) considered the problem of orienting planar parts resting in a planar tray. The tray could be tilted, causing the part to slide into walls and corners, thereby reorienting the part. The work implemented a planning and execution system based on Newtonian mechanics and Coulomb friction that would determine a sequence of tray-tilting operations guaranteed to unambiguously orient a part in the tray, whenever such a plan existed.

Other work has focused on obtaining low-polynomial-time algorithms for planning tray-tilting operations and for designing bowl feeders (Natarajan 1986). (Natarajan 1988) also obtained PSPACE-hardness results for the general sensorless motion planning problem in the presence of uncertainty. (Canny 1988) has derived NDEXPTIME-hardness results for this problem as well.

A very important mechanical operation for orienting parts is grasping. Another is pushing. See (Mason 1982, Mani and Wilson 1985, Mason 1986, Peshkin 1986, Brost 1988, Goldberg and Mason 1990) for work in this area.

(Boothroyd et. al 1972) determined the stable resting configurations of parts dropped onto a horizontal table. They also derived probabilities for transitions between these states as a function of the initial potential energy of the part.

(Grossman and Blasgen 1975) considered the problem of localizing a part by dropping the part into a dihedral corner, shaking the corner, then using probing operations to ascertain the configuration of the part.

(Taylor, Mason, and Goldberg 1987) re-introduced sensing into the tray-tilter. They proposed a general planning algorithm and suggested a means of comparing sensing and mechanical operations in terms of their respective information contents.

3 Definitions

We are given a convex polyhedron resting on a horizontal table. Friction between the table and the polyhedron is assumed to be infinite. The center of mass of the polyhedron is assumed to lie in the interior, or possibly on the boundary, of the polyhedron. We are given the position of the center of mass. We will use this to decide whether a given configuration of the polyhedron on the table is stable in the presence of gravity. Throughout the paper we will assume that all motions are quasi-static, that is, that inertial and impact forces may be ignored.

The table has two degrees of motion freedom. One degree of freedom corresponds to tilting the table from the horizontal, the other degree of freedom is the direction of the horizontal axis about which the table is tilted. The typical action that one may perform is a *wobble*. This action entails tilting the table from the horizontal up to some specified angle θ. The direction of the axis of rotation is chosen randomly. The reason for a random choice is to ensure that there is a non-zero probability that any particular side of the polyhedron will be pointing downhill. We will refer to the angle θ as the *tilt angle* of the action. We will restrict θ to the range $[0, \pi/2]$. At $\theta = 0$ the table is horizontal, while at $\theta = \pi/2$ the table is vertical.

We assume that a single tilt of the table from its horizontal position to a tilt angle of θ occurs nearly instantaneously, in the sense that it occurs faster than any dynamic motions of the polyhedron. The purpose of this assumption is again to ensure that there is a non-zero probability that any particular side of the polyhedron will be pointing downhill during a wobble action. Without this assumption some objects, such as conical cross sections, could roll in such a way as to prevent certain faces from ever pointing downhill.

Often we will perform a given wobble action repeatedly, until the polyhedron has settled into steady state. We will generally not distinguish between performing a single wobble and performing a series of wobbles for a sufficiently long time.

The basic state of the polyhedron is a face contact with the table. The orientation of the polyhedron about the normal to the table is in general unknown, and we will thus not include it in the definition of state. When the table is tilted, the polyhedron can rotate from one face to one or more adjacent faces, by tilting across bounding edges. The conditions under which such rotations are possible will be derived later. In general, for a given starting face, the polyhedron may be able to rotate to more than simply one adjacent face under a wobble action of the table. Thus the transitions between faces are non-deterministic. We can think of these non-deterministic transitions either as adversarial or as probabilistic. We will assume throughout this paper that the transitions are probabilistic. By this we mean that there is some non-zero probability that a possible non-deterministic face-to-face transition will actually be taken. Thus, over sufficiently many trials, we will see occurrences of all possible transitions out of a given face for a given tilt angle. The random choice of rotation axis is one approach for satisfying this assumption.

We will not worry about the values of the transition probabilities. For the purposes of this paper the probability values do not matter. In principle they could be obtained from experimental observation. Once known, one could determine the expected time that a wobble needs to be performed repeatedly in order for the system to settle into a steady state.

Here we are tacitly assuming that all face-face transitions may be viewed as rotations across edges. The more general case in which rotations occur across vertices may be handled in a similar manner as in this paper, although the complexity of operations increases considerably.

The main purpose in assuming probabilistic transitions is to ensure that possible transitions between contact states have non-zero probabilities of occurring. Consequently, if one performs a particular wobble action repeatedly then the system will eventually transit out of a given contact state if that is possible. The analogy to keep in mind is that of a Markov chain. The states of the Markov chain correspond to the face-table contacts. The Markov transitions correspond to the rotations from one face to another. For a given series of wobbles the system will settle into one or more possible equilibrium states. These are similar to the recurrent classes of the Markov chain.

There are two types of recurrent classes, those that contain a single state and those that contain several. The single-state classes correspond to stable resting configurations of the polyhedron on the (tilted) table. A recurrent class containing several states corresponds to a series of contact states between which the polyhedron rolls for the given tilt angle. For example, such a recurrent class might correspond to a tumbling motion of the polyhedron for a sufficiently steep tilt angle. If one now stops the tumbling motion, by returning the table to horizontal, then one's knowledge of the system's state is the entire recurrent class. In other words, the system could be in any of the face-table contacts comprising the recurrent class.

4 Problem Statement and Results

The basic problem is to orient the polyhedron. In other words, given initial uncertainty as to the resting configuration of the polyhedron, one would like to execute a series of wobbles at different tilt angles that minimizes the number of possible face-table contacts of the polyhedron. Ideally, the result is a single stable face-table contact.

The more general phrasing of the problem is as follows. We are given a polyhedron. The polyhedron need not be convex. However, since the only possible contacts between a polyhedron and a table are those on the convex hull of the polyhedron, we can assume without loss of generality that the polyhedron is convex.

The polyhedron has faces $\mathcal{F} = \{f_1, \ldots, f_n\}$. We will represent a face-table contact by the face f_i that is in contact with the table. We are given a set of possible initial contacts, $\mathcal{F}_I \subseteq \mathcal{F}$, all of which are stable resting configurations of the polyhedron on the table, and we are given a set of final contacts, $\mathcal{F}_F \subseteq \mathcal{F}$. The problem is to find a sequence of tilt angles $\theta_1, \theta_2, \ldots, \theta_\ell$ that are guaranteed to reorient the polyhedron from the initial set of possible contacts, $\mathcal{F}_I$, into some subset of the final contacts, $\mathcal{F}_F$.

In this paper we demonstrate an algorithm for determining the existence of such a strategy. The algorithm takes $O(n^4)$ time, and produces $\ell = O(n)$ number of actions.

5 Example

A useful example to keep in mind is the polyhedral version of an unfair coin. See Figure 1. Specifically, we will assume that the coin consists of two large parallel faces f_0 and f_{n+1}, and a set of small identical facets $f_1, \ldots, f_n$ that approximate the circumference of the coin. (We use the term 'facet' merely to distinguish the faces on the circumference of the coin from the large side faces of the coin.) The center of mass of the coin is symmetric with respect to these facets, but it is offset slightly closer to the face f_{n+1} than to the face f_0. The point is that for sufficiently small tilt angles, the coin, if upright initially, will roll on the small facets, but not fall over onto either of the flat faces. For slightly greater tilt angles, the coin can both roll and fall stably onto face f_{n+1}. For yet larger angles the coin can both roll and fall stably onto either face f_0 or face f_{n+1}. As one increases the tilt angle further the coin begins to

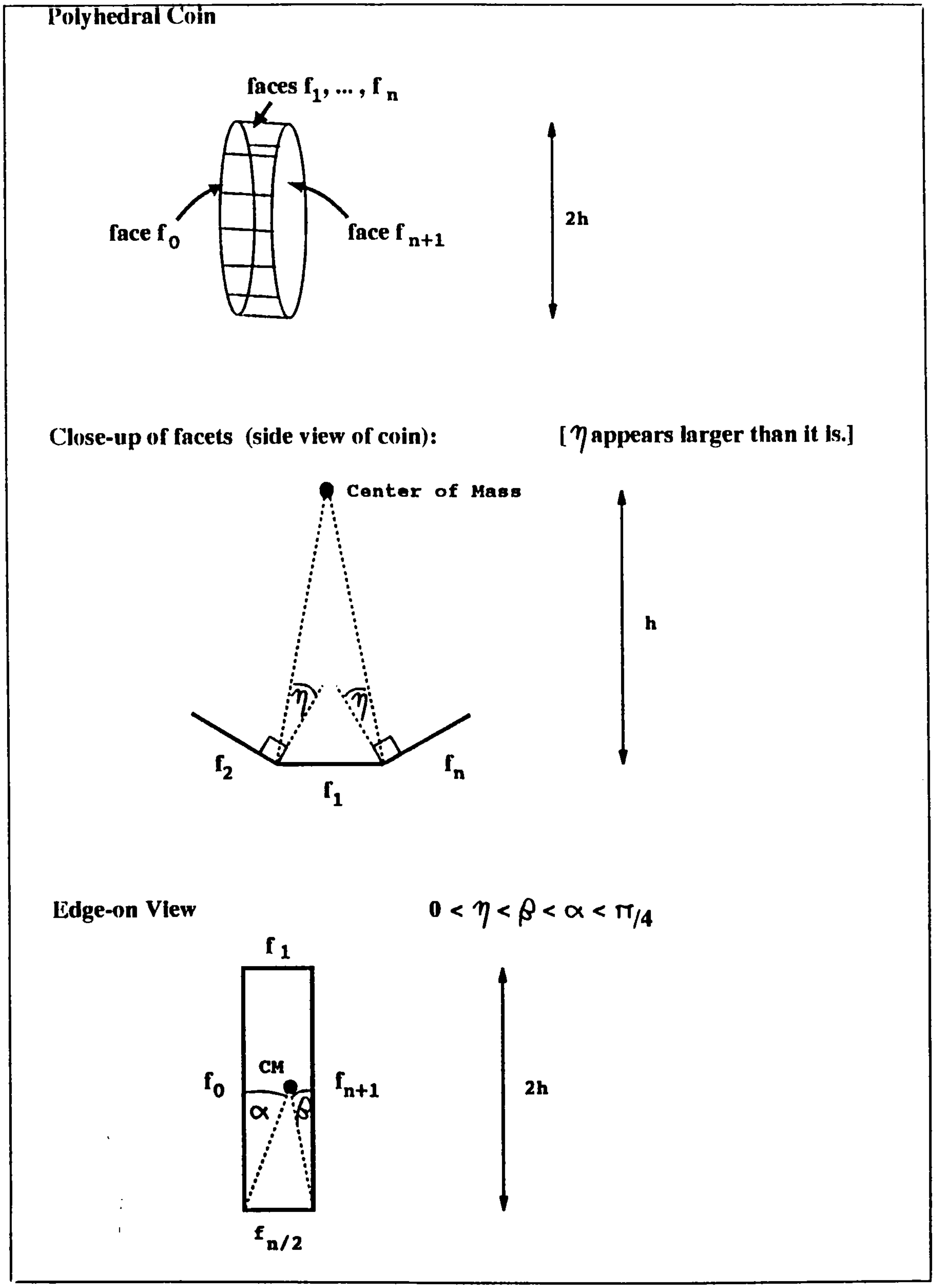

Figure 1: A thin polyhedral coin. The center of mass is biased to favor one of the large flat faces. It is symmetric with respect to the facets that comprise the circumference of the coin.

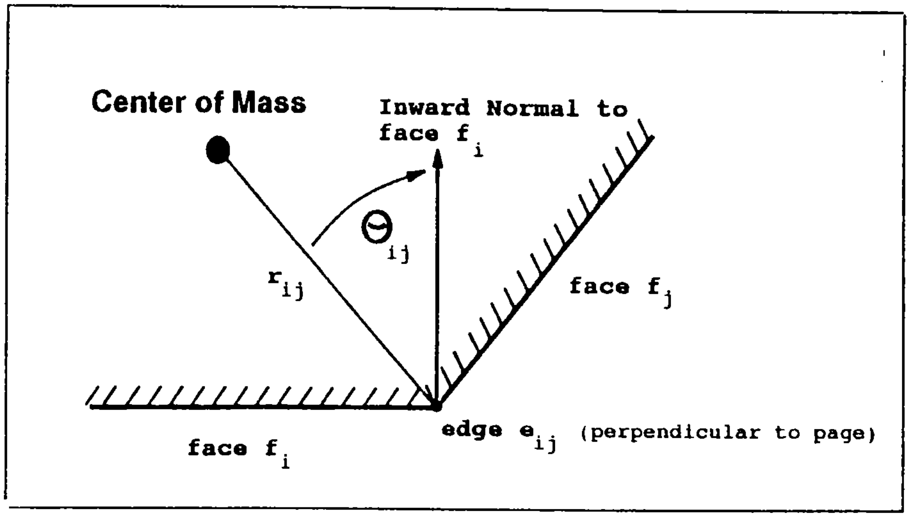

Figure 2: Description of the transition angle θ_{ij} between two faces of a polyhedron.

tumble. First it tumbles between face f_0 and the small facets, but eventually settles into stable equilibrium on face f_{n+1}. As the tilt angle is increased further yet, the coin tumbles between all faces. The details of this behavior will be derived in the remainder of the paper.

6 Transition Angles

Given two adjacent faces f_i and f_j on the polyhedron, we define θ_{ij} as the tilt angle that causes the polyhedron to rotate from face f_i to face f_j. Specifically, if face f_i of the polyhedron is in contact with the table, and the table is wobbled at an angle greater than θ_{ij}, the polyhedron may rotate away from face f_i, and one of the possible resulting face contacts is face f_j. Note that we are not saying anything about the stability of the face contact f_i before the rotation or of the face contact f_j after the rotation. We are saying merely that a rotation from f_i to f_j is possible if the tilt angle is greater than θ_{ij}, and indeed that it has non-zero probability of occurring. It is quite possible that contact f_i is unstable before the rotation, and it is quite possible that the polyhedron will continue to rotate after having made contact between the table and face f_j.

Let edge e_{ij} be the common edge between the two faces f_i and f_j. Consider rotating the polyhedron about the edge e_{ij}. Then θ_{ij} is the angle of rotation that moves the center of mass from above face f_i to just above face f_j. Here "above" is measured in terms of the direction of gravity. See Figure 2.

There are essentially three cases:

- First, if two faces f_i and f_j are not adjacent, we take $\theta_{ij} = +\infty$ since no direct rotation from f_i to f_j is possible without first rotating through some other face-table contact.

- Second, consider the polyhedron resting with face f_i on the table in its horizontal orientation. Let $\mathbf{r}_{ij}$ be the vector perpendicular to the edge e_{ij} that points from e_{ij} to the center of mass of the polyhedron. Define the *normal cylinder* above f_i to be the semi-infinite region of $\Re^3$ as the region resulting by sweeping the face along the direction of the normal. If the center of mass of the polyhedron lies in the normal cylinder above face f_i then the polyhedron will not rotate across edge e_{ij}. Said differently, if faces f_i and f_j were semi-infinite planes, then the face-table contact of the polyhedron on face f_i would be stable. In this case, the angle θ_{ij} is well-defined. It is simply the angle between the vector $\mathbf{r}_{ij}$ and the inward normal to face f_i. Clearly $0 \leq \theta_{ij} \leq \pi/2$.

- Third, if the center of mass does not lie in the normal cylinder to the face f_i then the contact is unstable, and thus must rotate. In this case we take θ_{ij} to be $-\infty$.

For the example of Figure 1, we obtain the following transition angles. For transitions between any two facets on the circumference of the coin, the tilt angle is η, where $\eta = \pi/n$. It is the same angle for all the facets since the center of mass is symmetrically located with respect to the facets. For rotation from a facet to the face f_0, the tilt angle is α, whereas from a facet to the face f_{n+1} the tilt angle is β. Since the center of mass is closer to f_{n+1} than to f_0, α is larger than β. In order to rotate in the opposite direction, that is, from face f_{n+1} to the facets on the circumference of the coin, the tilt angle must be at least $\pi/2 - \beta$. Similarly, in order to rotate from f_0 to the facets, the tilt angle must be at least $\pi/2 - \alpha$. Observe, of course, that in general the coin will tumble rather than remain stably poised on its circumference if the tilt angle is this large.

In short, we have

$$\begin{aligned}
\theta_{i,0} &= \alpha, && \text{for } 1 \leq i \leq n, \\
\theta_{i,j} &= \eta, && \text{for } 1 \leq i,j \leq n \text{ and } |i-j| = 1, \\
\theta_{i,n+1} &= \beta, && \text{for } 1 \leq i \leq n, \\
\theta_{0,j} &= \pi/2 - \alpha, && \text{for } 1 \leq j \leq n, \\
\theta_{n+1,j} &= \pi/2 - \beta, && \text{for } 1 \leq j \leq n,
\end{aligned}$$

with $0 < \eta < \beta < \alpha < \pi/4$ and thus $\pi/2 - \alpha < \pi/2 - \beta$.

7 Transition Graphs

For any tilt angle θ, we define a directed graph, G, that represents the possible transitions due to repeated wobbling at angle θ. The vertices of the digraph are labeled with the faces of $\mathcal{F}$. There is an edge between the vertices labeled with f_i

and f_j if the transition angle θ_{ij} is less than θ. In other words, if the tilt angle is greater than the transition angle then a rotation from f_i to f_j is possible. The case $\theta = \theta_{ij}$ is meta-stable. We will not worry about it here, since it will not arise in the remainder of the paper. We observe that the digraph is planar since it is a subset of the face-adjacency graph of a convex polyhedron.

Let us now sort the angles $\{\theta_{ij}\}$, yielding the following distinct set of angles:

$$-\infty = \theta_0 < \theta_1 < \cdots < \theta_m < \theta_{m+1} = +\infty,$$

with $\theta_i \in [0, \pi/2]$ for $i = 1, \ldots, m$, and $m \leq n$.

We now define action A_0 to be the null action which holds the tray in its horizontal position. Corresponding to A_0 one can construct a very simple digraph G_0, in the manner outlined above. The only transitions in this digraph are those from unstable face-table contacts to other face-table contacts.

Furthermore, for each θ_i, $i = 1, \ldots, m$, we define A_i to be the action consisting of repeated wobbling of the table at tilt angle $\theta_i + \epsilon$, where $0 < \epsilon < \theta_{i+1} - \theta_i$. This action involves tilting the table at an angle slightly greater than θ_i, but not as great as θ_{i+1}. Corresponding to A_i we obtain an appropriate digraph G_i. We observe for the graphs $G_0, \ldots, G_m$, that for $i < j$ the graph G_i is a subgraph of G_j.

The graphs $G_0, \ldots, G_5$ for the coin example of Figure 1 are displayed in Figures 3 and 4, and correspond to the following actions:

A_0	–	table is horizontal.
A_1	–	wobble table with tilt angle $\eta + \epsilon$.
A_2	–	wobble table with tilt angle $\beta + \epsilon$.
A_3	–	wobble table with tilt angle $\alpha + \epsilon$.
A_4	–	wobble table with tilt angle $\pi/2 - \alpha + \epsilon$.
A_5	–	wobble table with tilt angle $\pi/2 - \beta + \epsilon$.

Given the polyhedron as a planar face-adjacency graph with e directed edges labeled by the transition angles, we observe that each graph G_i may be computed in time O(e). If we are given the polyhedron simply as a set of oriented hyperplanes, then construction of the face-adjacency graph can be done in time $O(n^3)$.

8 Recurrent Classes

For each digraph G_i we can construct its recurrent classes. For our purposes a recurrent class of a digraph is a maximal set of vertices in the digraph in which any vertex is reachable from any other vertex by some sequence of edge transitions. In particular, a recurrent class has no edges that lead out of the class. Vertices that are not members of some recurrent class are called *transient states*. If a system starts inside of a recurrent class, then it will remain there. If a system starts in a transient state, then the assumption of probabilistic transitions ensures that eventually the

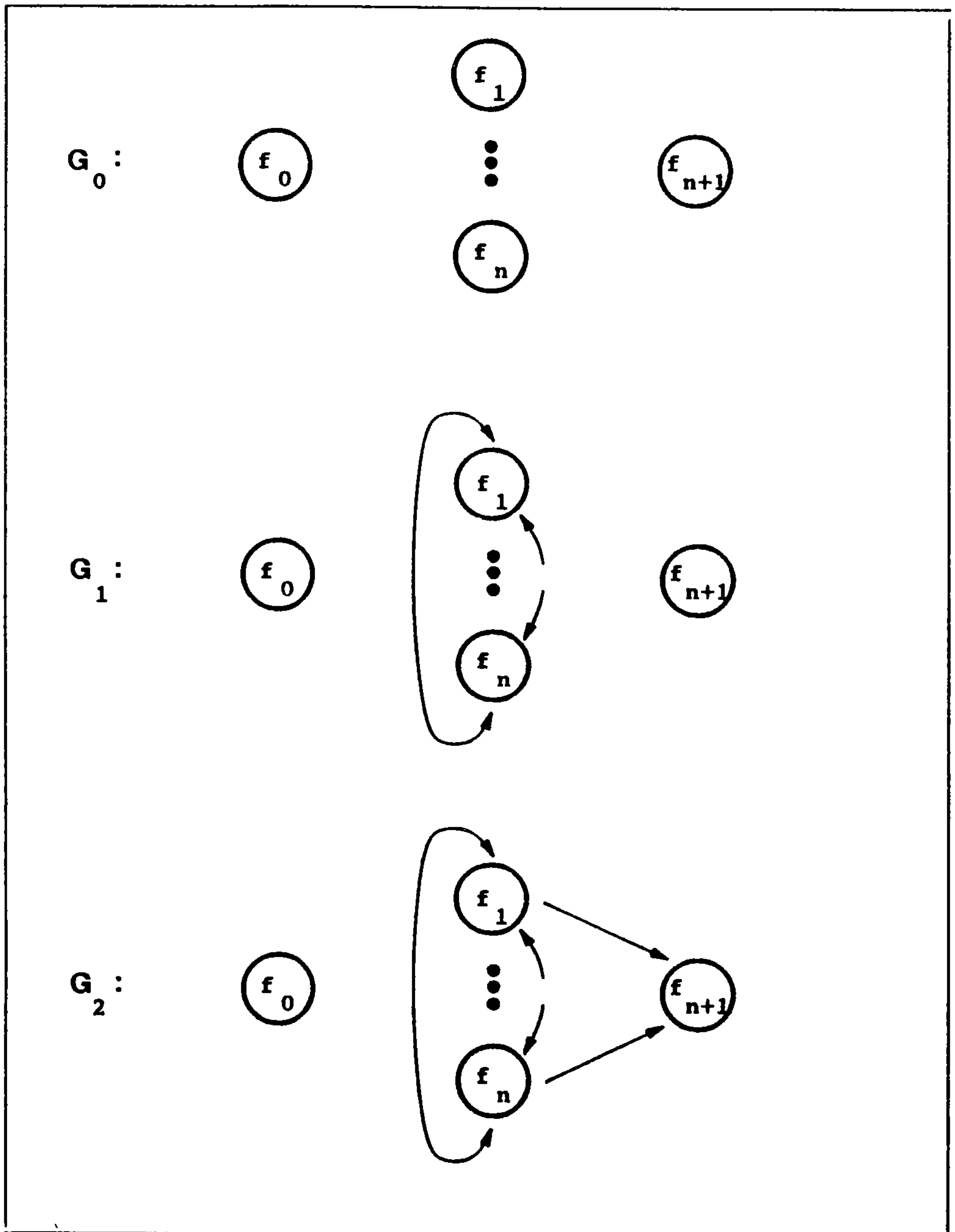

Figure 3: Three of the six transition graphs for the example of Figure 1. Graph G_i represents the transitions possible when the table is wobbled at a tilt angle slightly greater than θ_i. See also Figure 4.

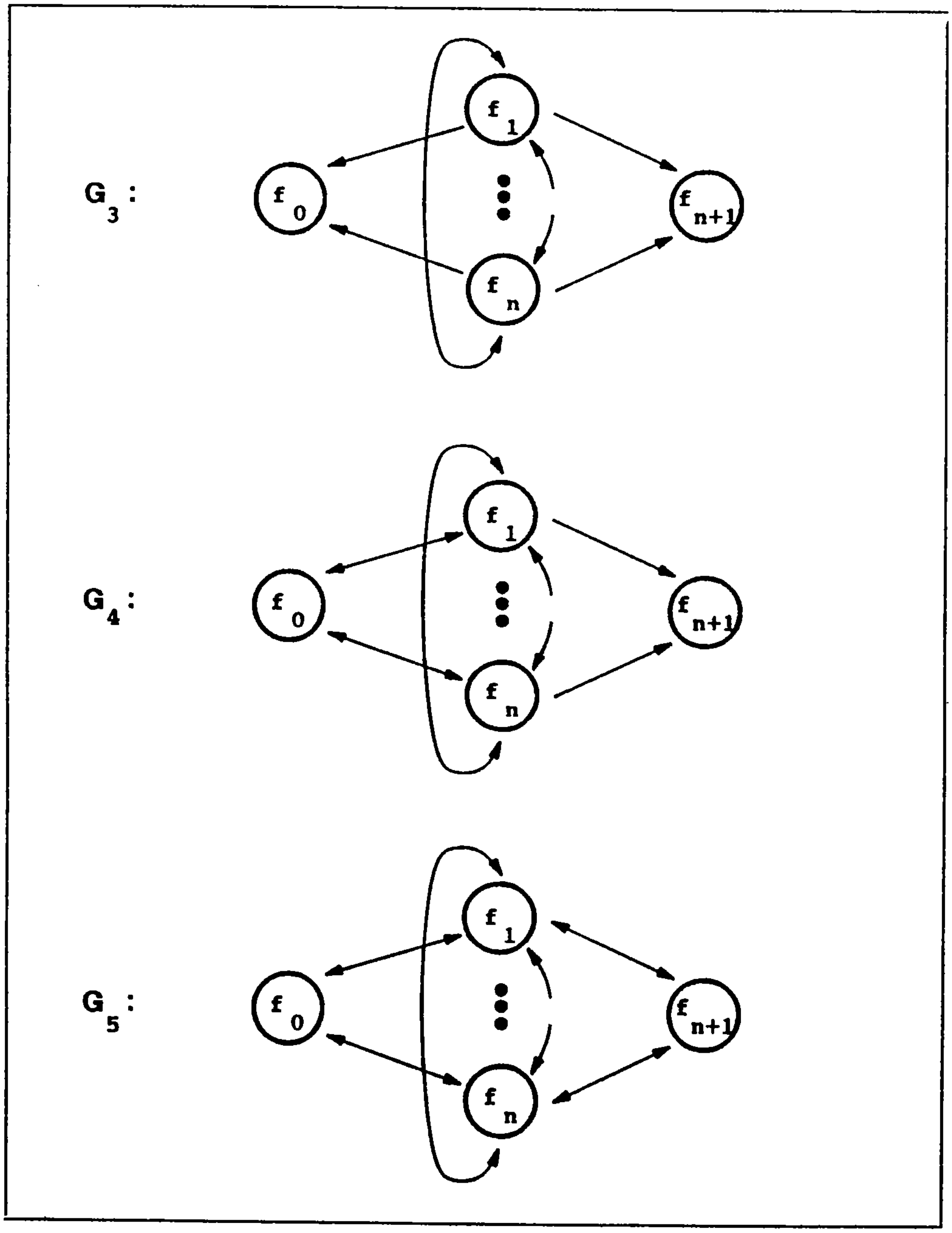

Figure 4: The remaining three transition graphs for the example of Figure 1. Graph G_i represents the transitions possible when the table is wobbled at a tilt angle slightly greater than θ_i. See also Figure 3.

system will move into some recurrent class, and remain within that class. Once in a given recurrent class, the state of the system is known to be one of the vertices comprising the class. In general, after application of action A_i, the state of the system is known simply to lie within the union of some subcollection of the recurrent classes of the graph G_i. The objective of a certainty-maximizing strategy is to make this union as small as possible by executing an appropriate sequence of actions.

Let us denote the collection of recurrent classes of G_i by $\{C_{i\alpha}\}$, where α runs over some index set containing at most n elements.

We observe that the set of recurrent classes of a graph G_i can be computed in time $O(n^3)$ using a transitive closure algorithm.

Consider the recurrent classes for the graphs in Figures 3 and 4. If we leave the table horizontal, then the coin can rest on any face, and thus the recurrent classes of G_0 are all the singleton face sets. There are n recurrent classes. If we tilt the table slightly more than η then the coin can roll on its circumference. Thus G_1 contains three recurrent classes, two of which are the large faces, and the third of which is the cycle of facets along the circumference of the coin. Once the table is wobbled at an angle slightly greater than β, then the coin can fall onto face f_{n+1}. Thus in G_2 there are now only two recurrent classes, given by the two flat faces. The facets on the circumference of the coin have become transient states, all of which eventually wind up on face f_{n+1}. For G_3, there are again two recurrent classes, but now the coin can fall onto either face f_0 or f_{n+1}. As the table is wobbled with angle slightly greater than $\pi/2 - \alpha$, face f_0 becomes a transient state as well. Thus G_4 contains a single recurrent class, given by face f_{n+1}. Finally, if the tilt angle is made very large, then the coin can tumble randomly. Thus G_5 has one recurrent class, consisting of all possible face-table contacts.

9 Forward Projections

Consider a set of possible face-table contacts $\mathcal{C}$ and some action A_i. We define the *forward projection of* $\mathcal{C}$ *under action* A_i as the set of final face-table contacts that might be attained at steady state after application of A_i, given that the system starts in a state of $\mathcal{C}$. We write this as $F_{A_i}(\mathcal{C})$. Clearly $F_{A_i}(\mathcal{C})$ is a subset of the possible recurrent classes. Furthermore, $F_{A_i}(\mathcal{C})$ may be computed in time $O(n)$ using a marking algorithm. Specifically, we do a depth-first search inside G_i starting from $\mathcal{C}$, marking vertices as we go until no more vertices can be marked. All vertices that are part of a recurrent class are retained.

Suppose that C is a recurrent class of graph G_j. In other words, C is some possible collection of states resulting from application of action A_j. If one now applies action A_i, for $i < j$, the system must wind up in some subset of the states of C. This is because the edges of G_i are a subset of the edges of G_j and because no edges lead out of the recurrent class C by definition of recurrent class. Said differently, if $\{C_{i\alpha}|C\}$ are the recurrent classes of G_i that are subsets of C, then the forward projection of

C under action A_i is the union of these recurrent classes:

$$F_{A_i}(C) = \bigcup_{\alpha} C_{i\alpha}|C.$$

10 Three Lemmas

We now prove three lemmas. The first lemma shows that whenever a certainty-maximizing strategy applies first action A_i and next action A_j, with $i < j$, then the strategy might as well apply the sequence of actions $A_i, A_{i+1}, \ldots, A_j$. The second lemma establishes a similar result for the opposite case in which $i > j$. Together these two lemmas show that any certainty-maximizing strategy need only step through wobbles in sequential order. In other words it is never necessary to apply actions A_i and A_j with $|i - j| > 1$.

Finally, the third lemma shows that nothing is to be gained by loops of actions of the form $A_j, A_{j-1}, \ldots, A_i, A_{i+1}, \ldots, A_j$, for $i < j$. As a result, we see that any certainty-maximizing strategy should be of the form $A_0, A_1, \ldots, A_j, A_{j-1}, \ldots, A_i$, with $0 \le i \le j \le m$. This then proves that any certainty-maximizing strategy has length $O(n)$.

Lemma 1 considers the difference between two simple strategies. In one strategy one applies a single action A_j. In the other strategy one first applies some action A_i with a lower tilt angle, and then applies action A_j. The claim of the lemma is that the possible final contacts of the polyhedron in the two-action case are a subset of those in the single-action case. Thus a certainty-maximizing strategy would prefer the two-action sequence.

Lemma 1 *Let f represent a face-table contact, and let $i < j$. Then*

$$F_{A_j}(F_{A_i}(\{f\})) \subseteq F_{A_j}(\{f\}).$$

Proof. The edges of G_i are a subset of the edges of G_j. □

The inclusion can in general be strict. Furthermore, the lemma need not hold for $i > j$. Fortunately, however, this other version of the lemma does hold if one replaces the single state f by a recurrent class corresponding to a tilt angle that is at least as steep as θ_j. This observation is the essence of the next lemma.

Lemma 2 *Let C be a recurrent class of the graph G_ℓ, for some ℓ. Suppose that $i < j \le \ell$. Then*

$$F_{A_i}(F_{A_j}(C)) \subseteq F_{A_i}(C).$$

Proof. Let $\{C_{j\alpha}|C\}$ be the recurrent classes of G_j that are subsets of C. Then

$$F_{A_i}(F_{A_j}(C)) = F_{A_i}(\bigcup_{\alpha} C_{j\alpha}|C) \subseteq F_{A_i}(C),$$

since each $C_{j\alpha}|C$ is a subset of C. □

Finally, we like to show that certain kinds of action loops are unnecessary. Specifically, it is never necessary to first perform a wobble at a given tilt angle, then wobble at a lower tilt angle for a while, only to wobble again at the original tilt angle. Induction on the claim of the following lemma establishes this fact.

Lemma 3 *Let the state of the system be known to lie in some collection of recurrent classes $\{C_{j\beta}\}$ of the graph G_j. If one now applies action A_{j-1} followed by action A_j, then the resulting set of states of the system is unchanged.*

Proof. By definition of forward projection

$$F_{A_j}(F_{A_{j-1}}(\bigcup_\beta C_{j\beta})) = \bigcup_\beta F_{A_j}(F_{A_{j-1}}(C_{j\beta})),$$

showing that it is enough to establish the lemma for a single recurrent class C of G_j. Therefore define, as usual, $\{C_{j-1,\alpha}|C\}$ to be the collection of recurrent classes of G_{j-1} that are subsets of C, and observe that

$$F_{A_j}(C_{j-1,\alpha}|C) = C$$

for each such recurrent class, since $C_{j-1,\alpha}|C \subseteq C$ and since C is a recurrent class. □

11 The Algorithm

Recall that we are given an initial set of face-table contacts $\mathcal{F}_I$ and possibly a final set of desired contacts $\mathcal{F}_F$. As we indicated in the previous section, the three lemmas establish that a certainty-maximizing strategy should be of the form $A_0, A_1, \ldots, A_j, A_{j-1}, \ldots, A_i$, for some i and j with $0 \leq i \leq j \leq m$. There are $O(n^2)$ choices of i and j. For each such choice we can forward project the set $\mathcal{F}_I$ through the relevant set of actions to obtain the possible resulting contact states of the polyhedron. Since forward projections can be computed in time $O(n)$ this yields a straightforward $O(n^4)$ algorithm for obtaining the result of all possible certainty-maximizing strategies applied to $\mathcal{F}_I$. By ordering the choices of i and j properly this may be reduced to an $O(n^3)$ algorithm. If the final set $\mathcal{F}_F$ is specified as well, then one can check, for each choice of i and j, whether the forward projection is a subset of $\mathcal{F}_F$. This decision may easily be implemented in time $O(n)$.

It remains to construct all of the graphs $\{G_i\}$ and their recurrent classes. As indicated, this may be accomplished in time $O(m\,n^3) = O(n^4)$. (Recall that m is the number of distinct tilt angles θ_{ij} obtained.)

As a final comment, we observe that in general it is not enough simply to perform actions with increasing tilt angles, ending with A_j. Instead, it is sometimes indeed necessary to include actions in which the tilt angles are again reduced, ending with some A_i, for $i < j$, as suggested by the statement of the algorithm. As an example, imagine that the face f_{n+1} in the coin example of Figure 1 is replaced with two smaller, unsymmetric faces that are nearly, but not quite coplanar. Thus the faces are distinguishable only for small tilt angles. A strategy for orienting the coin would consist of first increasing the tilt angle until the coin winds up on the recurrent class corresponding to the two subfaces of f_{n+1}, then reducing the tilt angle until the two subfaces are distinguishable, with the coin at rest on one particular subface of f_{n+1}. (For the example as drawn in Figure 1, there is no need to decrease the tilt angle. Rather one increases the tilt angle slightly beyond $\pi/2 - \alpha$, at which point the coin comes to rest on the face f_{n+1}.)

12 Special Case: Orienting Generic Polygons in the Plane

In the case of a two-dimensional planar polygon resting on a one-dimensional table, there is no need to perform a wobble of the table. Instead the table may be tilted either to the right or to the left, yielding either clockwise or counterclockwise rotations of the polygon, respectively. We can define the transition angles $\{\theta_{ij}\}$ similarly as we did before. In the case that no two of the angles θ_{ij} are equal there is a simple strategy for orienting the polygon. It consists of tilting the table at an angle slightly less than the maximum of the $\{\theta_{ij}\}$. The polygon will rotate until it winds up on the unique edge whose outgoing transition angle is equal to this maximum. The choice of whether to rotate right or left is determined by the direction of the maximum angle.

13 Conclusions and Future Work

This paper has explored automating the design of mechanical systems for orienting three-dimensional parts. The paper focused on the problem of orienting a three-dimensional polyhedron resting on a tiltable table. The polyhedron could be reoriented by wobbling the table. A wobble consisted of repeatedly tilting the table up to a specified tilt angle about a randomly chosen tilt axis.

The paper proposed a planner that determines a sequence of tilting operations designed to minimize the uncertainty in the part's orientation. The planner runs in time $O(n^4)$, where n is the number of faces of the polyhedron. The planner produces a sequence of $O(n)$ different wobbles.

The problem of orienting parts is a ubiquitous task in industrial assembly. The automated design of feeder equipment for orienting parts is thus an important problem. This paper suggests that a combination of probabilistic and geometric analyses may be used to design such feeder equipment.

The results presented in this paper suggest that parts may be oriented with very simple systems. Testing the practical validity of these systems requires further work. Future work should focus on the physical implementation of the system discussed in this paper. Additional work should consider the automated design of orienting systems for more complicated parts. Parallel orienting of several parts at once is one further direction to explore.

14 Bibliography

Boothroyd, G., Redford, A. H., Poli, C., and Murch, L. E. 1972. Statistical Distributions of Natural Resting Aspects of Parts for Automatic Handling. *Manufacturing Engineering Transactions.* 1:93–105.

Brost, R. C. 1988. Automatic Grasp Planning in the Presence of Uncertainty. *International Journal of Robotics Research.* 7(1):3–17.

Canny, J. F. 1988. *The Complexity of Robot Motion Planning.* Cambridge, Mass.: MIT Press.

Erdmann, M., and Mason, M. T. 1988. An Exploration of Sensorless Manipulation. *IEEE Journal of Robotics and Automation.* 4(4), (1988):369-379.

Grossman, D. D., and M. W. Blasgen. 1975. Orienting Mechanical Parts by Computer-Controlled Manipulator. *IEEE Transactions on Systems, Man, and Cybernetics.* SMC-5:561–565.

Goldberg, K., and Mason, M. 1990. Bayesian Grasping. *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, pp. 1264–1269.

Mani, M., and W. Wilson. 1985. A Programmable Orienting System for Flat Parts. *Proc., NAMRI XIII.*

Mason, M. T. 1982. Manipulator Grasping and Pushing Operations. AI-TR-690. Cambridge, Mass.: Massachusetts Institute of Technology, Artificial Intelligence Laboratory.

Mason, M. T. 1986. Mechanics and Planning of Manipulator Pushing Operations. *International Journal of Robotics Research.* 5(3):53–71.

Natarajan, B. K. 1986. An Algorithmic Approach to the Automated Design of Parts Orienters. *Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science*, pp. 132–142.

Natarajan, B. K. 1988. The Complexity of Fine Motion Planning. *International Journal of Robotics Research.* 7(2):36–42.

Peshkin, M. A. 1986. Planning Robotic Manipulation Strategies for Sliding Objects. Ph.D. Thesis. Carnegie-Mellon University, Physics Department.

Taylor, R. H., Mason, M. T., and Goldberg, K. Y. 1987. Sensor-Based Manipulation Planning as a Game with Nature. *Proceedings, Fourth International Symposium of Robotics Research.*