On Grasping and Manipulating Polygonal Objects with Disc-Shaped Robots in the Plane

Attawith Sudsang and Jean Ponce
Department of Computer Science and Beckman Institute
University of Illinois, Urbana, IL 61801, USA

Abstract: This paper addresses the problem of grasping and manipulating a polygonal object with three disc-shaped robots capable of translating in arbitrary directions in the plane. The main novelty of the proposed approach is that it does not assume that contact is maintained during the execution of the grasping/manipulation task. Nor does it rely on detailed (and a priori unverifiable) models of friction or contact dynamics. Instead, the range of possible object motions for a given position of the robots is characterized in configuration space. This allows the construction of manipulation plans guaranteed to succeed under the weaker assumption that jamming does not occur during the task execution.

1 Introduction

This paper addresses the problem of manipulating a planar polygonal object with three disc-shaped robots capable of translating in arbitrary directions in the plane. In practice, the discs may be the fingertips of a robot hand or independently-moving mobile platforms.

We propose an algorithm for grasping the object and bringing it to a desired position and orientation through sequences of individual straight-line robot motions. This algorithm guarantees that the object will never escape from the robots' grasp, even when contact is broken during the initial grasping phase or the subsequent manipulation stage. It does not require synchronizing the motion of the discs, and only assumes that each one of them can be moved in turn along a given straight line trajectory.

The proposed approach is based on a detailed analysis of the geometry of the joint object/robot configuration space. Instead of trying to predict the exact motion of the object, we characterize the range of possible motions associated with each position of the robots and identify the "minimal" robot configurations for which the object is totally immobilized as well as the "maximal" ones for which there is a non-empty open set of object motions within the grasp, but no escape path to infinity.

2 Background and Approach

When a hand holds an object at rest, the forces and moments exerted by the fingers should balance each other so as not to disturb the position of this object. We say that such a grasp achieves *equilibrium*. For the hand to hold the object securely, it should also be capable of preventing any motion due to external forces and torques. This is captured by the dual notions of *form and force closure* from screw theory [6, 13, 18], that constitute the traditional theoretical basis for grasp planning (see, for example, [8, 10, 11, 12, 17]). Recently, Rimon and Burdick have introduced the notion of *second-order immobility* [20] and shown that certain equilibrium grasps of a part which do not achieve form closure effectively prevent any *finite* motion of this part through curvature effects in configuration space. Algorithms for computing immobilizing grasps of planar and three-dimensional objects can be found in [15, 16, 23].

We introduced in [22, 23] the notion of *inescapable configuration space* (ICS) region for a grasp. This notion generalizes the concept of immobility: an object is immobilized when it rests at an isolated point of its free configuration space. By moving the fingers in an appropriate way, this isolated point transforms into a compact region of free space (the ICS) that cannot be escaped by the object. ICS regions were first introduced in the context of in-hand manipulation with a multi-fingered reconfigurable gripper [22, 23] (see [19] for a related notion in the two-finger case, and [1, 2, 3, 5, 7, 9, 14, 21] for other approaches to pushing and manipulation). Here, ICS regions will allow us to move an object by pushing it with three moving discs moving along straight lines: starting from some immobilizing configuration, we will move the robots one at a time in some direction, then choose another direction etc.. to achieve the desired translation and/or rota-

tion. The object will remain at all times in the ICS region associated with the fingers, so that the planned manipulation is guaranteed to be successful as long as friction forces are not large enough to cause jamming. In particular, our approach does not require that contact be maintained during grasping or manipulation, nor does it rely on any particular model of friction or contact dynamics.

Let us show an example to illustrate this idea (Fig. 1). The polygon shown in Fig. 1(a) is immobilized by the three discs since the three inward normals at the contacts intersect [20]. In Fig. 1(c), we translate one of the discs to a new position along the vector $\boldsymbol{v}$. During this motion, there is no path that will allow the polygon to escape the grasp of the three discs: the polygon can move, but is constrained to remain within the corresponding ICS region of free space. Figure 1(d)-(e) shows unsuccessful attempts to take the polygon out of the grasp.

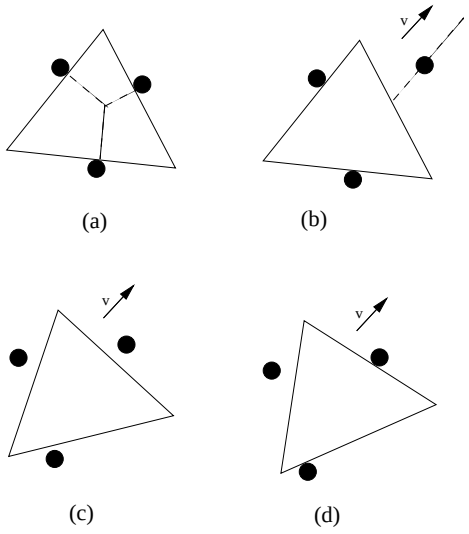

Figure 1: A polygon constrained to remain in the ICS region associated with the three robots.

3 Computing Maximum ICS Regions

We introduce formally in this section the concept of an inescapable configuration space region. The analysis proceeds along the lines of [22, 23] by identifying the constraints imposed by the robots in the configuration space of the polygon. The general approach is the same as in [22, 23] but the setting and the corresponding constraints are of course different.

3.1 Contact

We reduce the problem of achieving contact between a disc and a line to the problem of achieving point contact with a line. This is done without loss of generality by growing the object to be grasped by the disc radius and shrinking each disc into its center.

We attach a coordinate system (u, v) to the polygon, and write in this coordinate system the equations of the line supporting the edge e_i $(i = 1, 2, 3)$ as $u \cos\alpha_i + v \sin\alpha_i - d_i = 0$, where α_i is the angle between the u axis and the *internal* normal $\boldsymbol{n}_i$ to the edge, and d_i is the distance between the origin of the (u, v) coordinate system and the egde.

Without loss of generality, we also define a world coordinate system (q, r) such that the r axis is parallel to the motion direction $\boldsymbol{v}$ and goes through the center of the first (moving) disc. We denote by $\boldsymbol{q}_i = (q_i, r_i)^T$ the position of the center of disc number i in this coordinate system. In particular, $q_1 = 0$ and $r_1 = \delta$.

We can write the condition for contact between disc number i and the corresponding line as

$$\boldsymbol{q}_i = \mathcal{R}\boldsymbol{p}_i + \boldsymbol{t}, \tag{1}$$

where $\boldsymbol{p}_i = (u_i, v_i)^T$ and $\boldsymbol{q}_i = (q_i, r_i)^T$ denote the positions of the contact point in the two coordinate systems, $\mathcal{R}$ is a rotation matrix of angle θ and $\boldsymbol{t} = (x, y)^T$ is the translation between the two coordinate frames. Let $c_i = \cos(\theta + \alpha_i)$ and $s_i = \sin(\theta + \alpha_i)$, the above equation can be rewritten as

$$(x - q_i)c_i + (y - r_i)s_i + d_i = 0, \tag{2}$$

When the three contacts are achieved simultaneously, we have

$$\begin{pmatrix} c_1 & s_1 & \delta s_1 - d_1 \\ c_2 & s_2 & q_2 c_2 + r_2 s_2 - d_2 \\ c_3 & s_3 & q_3 c_3 + r_3 s_3 - d_3 \end{pmatrix} \begin{pmatrix} x \\ y \\ -1 \end{pmatrix} = 0.$$

For this equation to be satisfied, the determinant of the 3×3 matrix must be zero, which yields (after some simple algebraic manipulation):

$$\delta \sin(\theta + \alpha_1) + A_2 \cos(\theta + \beta_2) + A_3 \cos(\theta + \beta_3) - B = 0, \tag{3}$$

where β_2, β_3 and A_2, A_3, B are appropriate constants.

This condition defines a curve in θ, δ space, called the *contact curve*. This curve is defined on the $[0, 2\pi]$ interval, but an actual contact between the first disc and the corresponding edge can only occur when the angle between $\boldsymbol{v}$ and the internal normal to the edge is obtuse, i.e., when $\theta + \alpha_i \in [\pi, 2\pi]$. It follows from the form of its equation that the contact curve is in fact bounded by two vertical asymptotes on that interval.

3.2 Equilibrium

At equilibrium, the various forces and moments exerted at the contacts balance each other. This can be

written in the object's coordinate system as

$$\sum_{i=1}^{3} \lambda_i \begin{pmatrix} \boldsymbol{n}_i \\ \boldsymbol{p}_i \times \boldsymbol{n}_i \end{pmatrix} = 0, \quad \text{where} \quad \begin{cases} \lambda_1, \lambda_2, \lambda_3 > 0, \\ \lambda_1 + \lambda_2 + \lambda_3 = 1. \end{cases}$$

Using the change of coordinates (1) and taking advantage of the fact that $\sum_{i=1}^{3} \lambda_i \boldsymbol{n}_i = 0$ allows us to rewrite this equation as

$$\sum_{i=1}^{3} \lambda_i \begin{pmatrix} \boldsymbol{n}_i \\ (\mathcal{R}^{-1}\boldsymbol{q}_i) \times \boldsymbol{n}_i \end{pmatrix} = 0,$$

which can be interpreted as a 3×3 homogeneous equation in the coefficients $\lambda_1, \lambda_2, \lambda_3$. A necessary and sufficient condition for this equation to have a non-trivial solution is that its determinant be zero, i.e.,

$$\begin{vmatrix} \boldsymbol{n}_1 & \boldsymbol{n}_2 & \boldsymbol{n}_3 \\ (\mathcal{R}^{-1}\boldsymbol{q}_1) \times \boldsymbol{n}_1 & (\mathcal{R}^{-1}\boldsymbol{q}_2) \times \boldsymbol{n}_2 & (\mathcal{R}^{-1}\boldsymbol{q}_3) \times \boldsymbol{n}_3 \end{vmatrix} = 0.$$

Expanding the determinant yields, after some additional algebraic manipulation, the condition

$$\delta \cos(\theta + \alpha_1) - A_2 \sin(\theta + \beta_2) - A_3 \sin(\theta + \beta_3) = 0,$$

and eliminating δ between this equation and the contact constraint (3) yields an equation in θ only:

$$\cos(\theta + \alpha_1) = \frac{A_2}{B} \cos(\beta_2 - \alpha_1) + \frac{A_3}{B} \cos(\beta_3 - \alpha_1). \quad (4)$$

There are (at most) two solutions for this equation in the $[0, 2\pi]$ interval. When they exist, exactly one of them is in the interval of physically achievable contacts. It is also easy to show that the corresponding solution is a minimum of the contact curve. As in [22, 23], this minimum corresponds to an immobilizing configuration [20].[1]

Figure 2 shows an actual example in the object's and disc's coordinate frames. The triangle has to rotate 60 degrees counterclockwise to be immobilized by the matching discs (Fig. 2(c)). This is verified on the contact curve shown in Fig. 2(e) where the minimum occurs at 60 degrees in the physically realizable interval. The maximum of the curve corresponds to the configuration shown in Fig. 2(d), and it cannot be achieved in reality: the first disc would have to lie inside the triangle.

[1] The object will be immobilized even if there is no friction: although this appears to contradict classical screw theory, which states that three fingers are not sufficient to immobilize a two-dimensional object in that case [6], recall that screw theory is concerned with *infinitesimal* motions: there exists an escape velocity but no finite escape motion. See [20] for details.

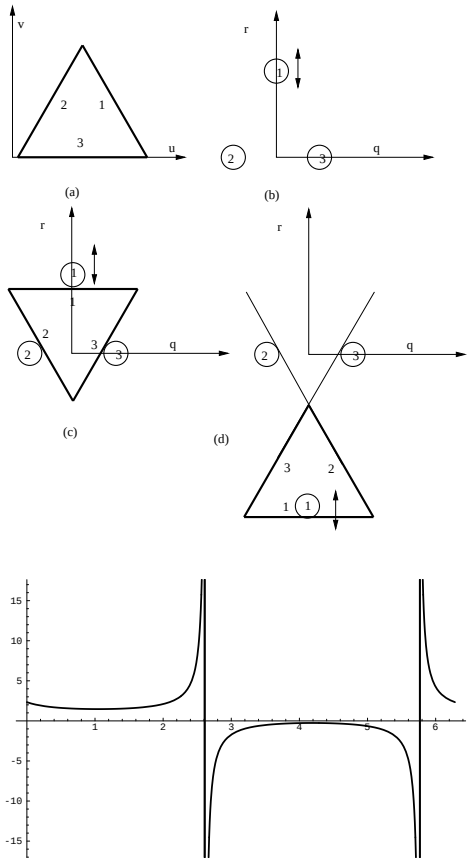

Figure 2: A grasp and the corresponding contact curve.

3.3 Free Configuration Space Regions

Let us consider an immobilizing configuration of the robots, and denote by $x_0, y_0, \theta_0, \delta_0$ the corresponding values of x, y, θ, δ. Let us also assume that the positions of robots 2 and 3 are held constant while the δ coordinate of the first robot may change.

We denote by S_i the set of object configurations (x, y, θ) for which contact between disc number i and the corresponding object edge is achieved. From (2), this is a ruled surface in (x, y, θ) space, whose intersection with a plane $\theta =$ constant is a line $L_i(\theta)$ at distance $-d_i$ from the fixed point (q_i, r_i) of the x, y plane, and the angle between the x axis and the normal to this line is $\theta + \alpha_i$. Changing θ corresponds to rotating each line about the point (q_i, r_i). Changing δ amounts to translating the line $L_1(\theta)$.

Together, the three ruled surfaces S_1, S_2 and S_3 bound a volume V of free configuration space. Given the setup of the robots, it is obvious that if a configuration lies in free space for some value δ_1 of δ, it also lies in free space for any other value $\delta_2 \geq \delta_1$. In other words, $V(\delta_1) \subset V(\delta_2)$ when $\delta_2 \geq \delta_1$, and it fol-

lows that the immobilizing configuration (x_0, y_0, θ_0) is always in free space for $\delta \geq \delta_0$.

In addition, the intersection of V with a plane $\theta =$ constant is a triangle $T(\theta)$ that may contain an open subset, be reduced to a point, or be empty. In the second case, the three contacts are simultaneously achieved, and (3) is satisfied.

It is easy to show that a necessary and sufficient condition for the triangle $T(\theta)$ to contain at least one point is that the point (θ, δ) be *above* the contact curve. This allows us to characterize qualitatively the range of orientations θ for which $T(\theta)$ is not empty: for a given δ, the condition (3) is an equation in θ that may have zero, one, or two real solutions, with a double root at the minimum $\delta = \delta_0$ of the curve. In this case, the range of orientations reduces to a single point. For any value $\delta_1 > \delta_0$, there are two distinct roots θ', θ'', and the range of orientations is the arc bounded by these roots and containing θ_0.

In particular, since the volume V is a stack of contiguous triangles $T(\theta)$, it is clear at this point that, for $\delta \geq \delta_0$, V is a non-empty, connected, compact region of $\mathbb{R}^2 \times S^1$. The analysis confirms that the minimum point (θ_0, δ_0) of the contact curve corresponds to an isolated point of configuration space or equivalently to an immobilizing configuration: indeed, for $\delta = \delta_0$, the triangle $T(\theta_0)$ is reduced to a point, and $T(\theta)$ is empty for any $\theta \neq \theta_0$.

3.4 ICS Regions

The discussion so far has characterized the contacts between the discs and the lines supporting the corresponding edges, ignoring the fact that each edge is a compact line segment. For a given value of δ, let us construct a parameterization of the set $E_i(\theta)$ of configurations (x, y) for which disc number i touches the edge e_i. Obviously, $E_i(\theta)$ is itself a line segment supported by the line $L_i(\theta)$.

We first parameterize the corresponding edge e_i by

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = d_i \begin{pmatrix} \cos\alpha_i \\ \sin\alpha_i \end{pmatrix} + \eta_i \begin{pmatrix} -\sin\alpha_i \\ \cos\alpha_i \end{pmatrix},$$

with η_i in some interval $[\eta_{i1}, \eta_{i2}]$. The segment $E_i(\theta)$ can now be parameterized by

$$\begin{pmatrix} x - q_i \\ y - r_i \end{pmatrix} = -d_i \begin{pmatrix} c_i \\ s_i \end{pmatrix} - \eta_i \begin{pmatrix} -s_i \\ c_i \end{pmatrix}. \qquad (5)$$

The constraints $\eta_{i1} \leq \eta \leq \eta_{i2}$ $(i = 1, 2, 3)$ define the regions of configuration space where actual contact will occur. When $E_i(\theta)$ and $E_j(\theta)$ intersect for all $i \neq j$, the three segments completely enclose the triangle $T(\theta)$, and we will say that the corresponding configuration satisfies the *enclosure condition* since there is no escape path for the object in the x, y plane with the corresponding orientation θ. More generally, when all triples of segments in the range of orientations associated with a given δ satisfy the enclosure condition, V itself is an *inescapable configuration space* (ICS) region: in other words, the object is free to move within the region V, but remains imprisoned by the grasp and cannot escape to infinity.

3.5 Maximum ICS Regions

We now address the problem of characterizing the maximum value δ^* for which $V(\delta)$ forms an ICS region for any δ in the $[\delta_0, \delta^*]$ interval. We know that at $\delta = \delta_0$ the three segments intersect at the immobilizing configuration, forming an ICS region reduced to a single point. Thus the enclosure condition holds at $\delta = \delta_0$. On the other hand, as $\delta \to +\infty$, the whole configuration space becomes free of obstacles, thus there must exist a critical point for some minimal value of δ greater than δ_0. This guarantees that δ^* has a finite value.

A critical configuration occurs when an endpoint of the segment $E_i(\theta)$ lies on the line $L_j(\theta)$, $j \neq i$. We intersect the lines $L_i(\theta)$ and $L_j(\theta)$ by substituting the parameterization (5) in the contact equation (2). Writing $\eta_i = \eta_{ik}$ $(k = 1, 2)$ yields

$$\eta_{ik} = -\frac{d_j - d_i\cos(\alpha_i - \alpha_j) + (q_i - q_j)c_j + (r_i - r_j)s_j}{\sin(\alpha_i - \alpha_j)}. \qquad (6)$$

It follows that critical points lie on one of the six *critical curves* of (θ, δ) space defined by (6) for $i, j \in \{1, 2, 3\}$ $(i \neq j)$ and $k = 1, 2$. Note that when $i, j \in \{2, 3\}$, (6) is a function of θ only, and the corresponding critical curves are vertical.

We seek the minimum value of $\delta^* > \delta_0$ for which the range of possible object orientations defined by the contact curve includes one of the critical configurations. Let us suppose first that a critical value lies in the interior of the orientation range associated with some $\delta_1 \geq \delta_0$, and denote by $\delta_{\min}$ the minimum value of δ on the critical curve. By definition, we have $\delta_1 \geq \delta_{\min}$. Suppose that $\delta_1 > \delta_{\min}$. Then by continuity, there exists some δ_2 such that $\delta_{\min} < \delta_2 < \delta_1$ and the corresponding range of orientations also contains a critical orientation. The argument holds for any value $\delta > \delta_{\min}$. In other words, either the range of orientations of $\delta_{\min}$ contains a critical orientation, in which case $\delta^* = \delta_{\min}$, or it does not, in which case the critical value associated with δ^* must be one of its range's endpoints. This is checked by intersecting the contact curve and the critical curve. Note that this process must be repeated six times (once per each

segment/vertex pair) to select the minimum value of δ^*.

Figure 3 shows an example, where the contact and critical curves have been constructed for some sample object (the contact curve is drawn with a thicker brush). In this case, the minimum of the critical curve occurs just below the contact curve, and the critical configuration is the intersection of the two curves, lying at the right endpoint of the corresponding range of orientations.

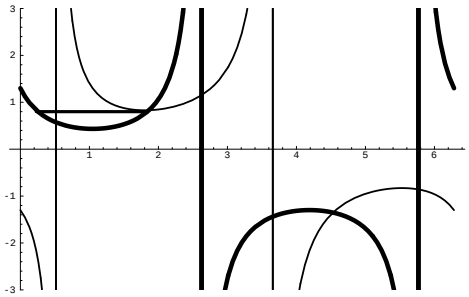

Figure 3: The contact and critical curves for a sample object. The critical range of orientations is shown as a horizontal line. See text for details.

Computing the minimum of the critical curve amounts to solving a trigonometric equation. It is easily shown that intersecting the critical curve and the intersection curve amounts to solving a quadratic equation in $\tan(\theta/2)$ when $i = 2,3$ and $j = 1$, and a quartic equation in the same variable when $i = 1$, $j = 2,3$. The intersection can be computed in closed form in both cases.

4 Planning Grasping and Manipulation Sequences

It is easily shown that the set of equilibrium (hence immobilizing) grasps of a polygon can be identified through linear programming, and various grasp optimality criteria (e.g., [4, 8, 10, 16]) can be defined to choose a particular immobilizing grasp among this set. In this section, we will assume that an immobilizing grasp has been selected and that the initial object position and orientation are known, and we will show how to actually execute the grasp and then manipulate the polygon, moving the three robots one at a time while guaranteeing that the object will not escape.

In the rest of this section, a joint configuration of the polygon and the robots will be denoted by $q = (\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3, x, y, \theta)$, where $\boldsymbol{q}_i$ is as before the position of disc number i and (x, y, θ) denotes the polygon configuration. Given an immobilizing configuration q, $\mathrm{MaxICS}(q, i, \boldsymbol{v})$ will denote the maximum distance that robot number i can travel in the direction $\boldsymbol{v}$ while guaranteeing that the object cannot escape.

4.1 Capturing and Grasping a Polygon

Given some input grasp configuration q, we choose one of the discs (say the first one) and some direction $\boldsymbol{v}$, say the external normal to the corresponding edge, and compute $\delta = \mathrm{MaxICS}(q, 1, \boldsymbol{v})$ as described in the previous section. To capture the object, we first move the robots one by one from their home position to $\boldsymbol{q}_1 + \frac{1}{2}\delta\boldsymbol{v}$, $\boldsymbol{q}_2 - \frac{1}{2}\delta\boldsymbol{v}$, and $\boldsymbol{q}_3 - \frac{1}{2}\delta\boldsymbol{v}$. The polygon is now guaranteed to be in the maximum ICS region associated with the robots. We then translate the first robot by $-\delta\boldsymbol{v}$.

Although the object may (and indeed will) move when contact occurs, it will end up in the planned immobilized configuration. Note that this approach is robust to uncertainty in the position of the object, but that it requires precise relative motions of the robots.

In the next two sections, we show how to achieve arbitrary translations and rotations of the object once it has been grasped. The overall motion will be decomposed into atomic translations of the three fingers along appropriate directions. The object will remained imprisoned in the grasp of the three robots during each motion.

4.2 Translating a Polygon

Let us assume that the object is currently immobilized by the discs in configuration q, and let us show how to apply the translation $d\boldsymbol{v}$ to the polygon. The immobilizing configuration is shown in Fig. 4(a). To translate the polygon, we will apply a translation $\delta\boldsymbol{v}$ to discs 2, 3, and 1 in succession. The problem is to compute the maximum value of δ guaranteeing that the polygon cannot escape at any time. If $d < \delta$, we will reset δ to d before applying the translation. If $d > \delta$, we will simply apply the same translation steps as many times as necessary.

The first thing to note is that the object should not escape the grasp when we move disc 2, so δ must be smaller than or equal to $\delta_2 = \mathrm{MaxICS}(q, 2, \boldsymbol{v})$. Likewise, once we have moved discs 2 and 3 by $\delta\boldsymbol{v}$, the polygon should still be unable to escape, which implies that δ must be smaller than $\delta_1 = \mathrm{MaxICS}(q, 1, -\boldsymbol{v})$.

Using the value $\delta = \min(\delta_1, \delta_2)$ is not sufficient because the polygon may move when the contact with discs 2 and 3 is broken. We use bisection to compute the maximum value of δ in the $[0, \min(\delta_1, \delta_2)]$ range such that both discs 2 and 3 can undergo a $\delta\boldsymbol{v}$ translation while maintaining inescapability (Fig. 4): For a given value of δ, we suppose first that disc 2 has already moved to its new position (Fig. 4(b)), then find the translation γ of disc 3 along $\boldsymbol{v}$ that will

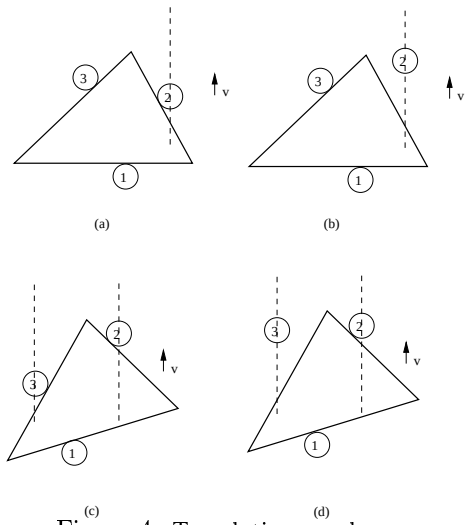

Figure 4: Translating a polygon.

yield a new immobilizing configuration, say q'(Fig. 4(c)). Consider now the net maximum translation $\delta' = \text{MaxICS}(q', 3, \boldsymbol{v}) - \gamma$ that disc 3 may undergo while guaranteeing that the polygon cannot escape. If $\delta' \geq \delta$, we know that disc 2, then disc 3 can safely undergo the translation $\delta\boldsymbol{v}$: the bisection step is successful and we increase the value of δ. If $\delta' \leq \delta$, then the bisection step has failed, and we try again with a lower value of δ. Figures 4 and 5 show respectively a failed bisection step and a successful one.

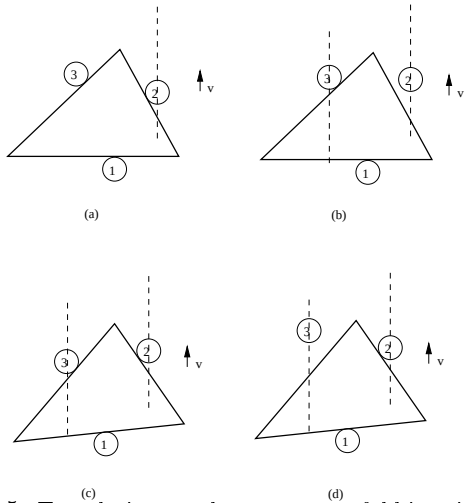

Figure 5: Translating a polygon: successful bisection step.

A pseudocode version of the algorithm is given below. In it, q denotes the initial configuration, the function $\text{ImTransl}_3(\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3, \boldsymbol{v})$ returns the distance that $\boldsymbol{q}_3$ must travel along $\boldsymbol{v}$ to reach an immobilizing configuration, and the function $\text{ImConfig}(\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3)$ returns the object configuration associated with the contacts $\boldsymbol{q}_i$. The output of these two functions is easily computed using the contact and equilibrium conditions (2), (3) and (4). The variables ν and δ below are respectively the low and high values used to bound the bisection.

```
1:    q = (q1, q2, q3, x, y, θ);
2:    δ1 = MaxICS(q, 1, −v);
3:    δ2 = MaxICS(q, 2, v);
4:    ν = 0.0;
5:    if δ2 > δ1 then δ = δ1
6:         else δ = δ2;
7:    do {
8:         q2' = q2 + δv;
9:         γ = ImTransl3(q1, q2', q3, v);
10:        q3' = q3 + γv;
11:        (x', y', θ') = ImConfig(q1, q2', q3');
12:        q' = (q1, q2', q3', x', y', θ');
13:        δ3 = MaxICS(q', 3, v);
14:        if δ3 − γ ≥ δ
15:             then ν = (δ + ν)/2
16:             else δ = (δ + ν)/2
17:   } until (δ − ν < ε);
18:   return δ.
```

So far, we have shown how to translate the polygon in one direction. We can translate the polygon in arbitrary directions using plans computed for three directions only by switching the roles of the discs and alternating between the three directions. If the chosen directions positively span the plane, it is easy to see that we can arrange a sequence of translations to bring the polygon to any position. One simple choice for these directions is the inward normals at the contacts.

4.3 Rotating a Polygon

Some of the steps involved in translating a polygon also prove useful in rotating it (Fig. 4): Starting in some configuration q, we rotate the polygon counterclockwise by first translating disc 2 in direction $\boldsymbol{v}$ for some distance δ (Fig. 4(b)), before translating disc 3 in the direction $-\boldsymbol{v}$ until the polygon is immobilized in a new configuration q' (Fig. 4(c)). To guarantee that the polygon will not escape, we must have $\delta \leq \text{MaxICS}(q, 2, \boldsymbol{v})$ and $\delta \leq \text{MaxICS}(q', 3, \boldsymbol{v})$. Note that in this case the translation of disc 2 is actually performed, but disc 1 does not move, and thus does not constrain the value of δ. The maximum value of δ is found as before through bisection. Clearly, for a given vector $\boldsymbol{v}$, this value yields the maximum possible rotation. Achieving a smaller rotation is done by using a smaller δ (the corresponding positions of discs 2 and

3 are once again easily calculated in this case using the equilibrium and contact conditions). Achieving a larger one involves repeating the same elementary rotation several times.

Note that the rotation steps will change the positions of the discs along the edges of the polygon (Fig. 6(a)). We would like these positions to remain unchanged so that we can apply the same rotating steps repeatedly and achieve a pure rotation (Fig. 6(b)).

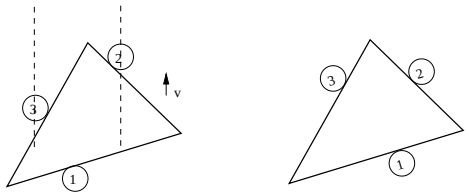

Figure 6: Effect of a rotation: (a) the position of the discs along the edges has changed; (b) corrected disc positions.

To correct the disc positions, we move them one by one toward their initial configuration along the edges. The object may move during this correction stage, but it will eventually come back to its original position and orientation, which are uniquely determined by the disc positions in their initial configurations. The difficulty is to guarantee that the polygon will remain imprisoned by the discs during the correction steps. This corresponds to computing the maximum ICS region in the direction of the corresponding edges. The ICS computation degenerates in this case to the following construction (Fig. 7): In Fig. 7(a), the polygon is immobilized, and disc 2 touches the edge E in A. Let L denote the line supporting E in this configuration (L is fixed but E will move when the position of disc 2 changes), and let $\boldsymbol{w}$ denote the direction in which we want to move disc 2 along L. We want to find the furthermost point on L to which we can move disc 2 from A in the direction $\boldsymbol{w}$.

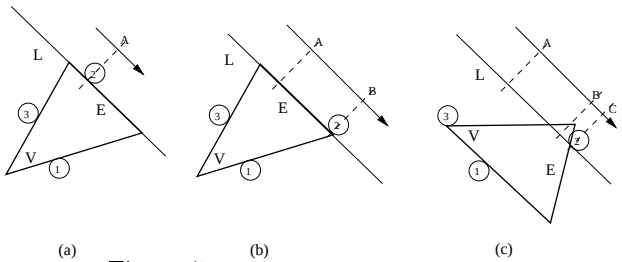

Figure 7: Moving a disc along a line.

If this point is further than the edge endpoint B, the polygon will obviously be able to escape (Fig. 7(b)). In addition, moving disc 2 in the direction $\boldsymbol{w}$ will allow the polygon to rotate clockwise. Figure 7(c) shows the maximum clockwise rotation preventing the polygon to escape (vertex V touches disc 3). Let C denote the intersection of L and E in that case. Clearly, the point we seek must not be further than C, since this would also allow the polygon to escape. In general, it can be shown that we can safely move disc 2 along L anywhere from A to the closest of the two points B and C.

With this method, we move a disc toward its initial position along the corresponding edge (if the position is not in the allowable range, we move to the point closest to the position), update the current position of the disc, apply the method to another disc, and continue this procedure until all three discs reach the desired positions.

Note that the rotation steps also affect the overall position of the polygon. To perform a pure rotation, a final translation stage has to be performed, using the technique presented in the previous section.

5 Implementation and Results

We have implemented the algorithm for planning manipulation sequences described in the previous section. Figure 8 shows intermediate immobilizing configurations in a manipulation sequence that brings an equilateral triangle with edges of unit length from some initial configuration (lower right) to a goal configuration (upper left). The triangle is first rotated to the desired orientation, and then translated to the desired position. The entire sequence has 28 steps. By choosing a grasp with contacts at the center of the edges, and the inward normals as the translation directions, we obtain a maximum translation distance of 0.092, and a maximum rotation angle of 14.98 degrees. The program takes less than 1 second to compute the sequence on a 200MHz PC.

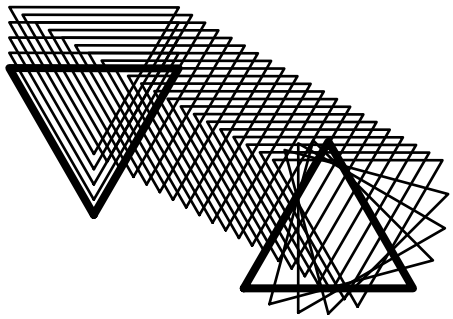

Figure 8: A manipulation sequence.

Figure 9 shows snapshots of some of the translation and rotation steps used in the experiment of Fig. 8.

Figure 9(a) shows the input grasp. Figure 9(b)-(d) shows the elementary stages of a translation step, and Fig. 9(e)-(i) shows the elementary stages of a rotation step, including the disc motions along the edges.

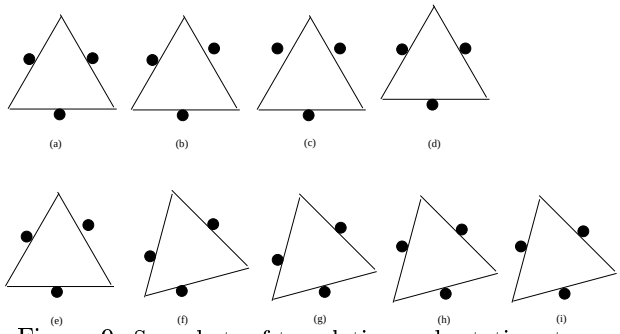

Figure 9: Snapshots of translation and rotation steps.

Acknowledgments. This research was supported in part by the National Science Foundation under grant IRI-9634393, a UIUC Research Board grant, a UIUC Critical Research Initiative planning grant, and an equipment grant from the Beckman Institute for Advanced Science and Technology.

References

[1] S. Akella and M.T. Mason. Parts orienting by push-aligning. In *IEEE Int. Conf. on Robotics and Automation*, pages 414–420, Nagoya, Japan, May 1995.

[2] Y. Citour, A. Marigo, D. Prattichizzo, and A. Bicchi. Rolling polyhedra on a plane, analysis of the reachable set. In J.-P. Laumont and M. Overmars, editors, *Algorithmic Foundations of Robotics II*, pages 277–285. AK Peters, Ltd., 1997.

[3] M.A. Erdmann and M.T. Mason. An exploration of sensorless manipulation. *IEEE Journal of Robotics and Automation*, 4:369–379, 1988.

[4] C. Ferrari and J.F. Canny. Planning optimal grasps. In *IEEE Int. Conf. on Robotics and Automation*, pages 2290–2295, Nice, France, June 1992.

[5] K.Y. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10(2):201–225, 1993.

[6] K. Lakshminarayana. Mechanics of form closure. Technical Report 78-DET-32, ASME, 1978.

[7] K.M. Lynch and M.T. Mason. Stable pushing: mechanics, controllability, and planning. In K.Y. Goldberg, D. Halperin, J.-C. Latombe, and R. Wilson, editors, *Algorithmic Foundations of Robotics*, pages 239–262. A.K. Peters, 1995.

[8] X. Markenscoff and C.H. Papadimitriou. Optimum grip of a polygon. *International Journal of Robotics Research*, 8(2):17–29, April 1989.

[9] M.T. Mason. Mechanics and planning of manipulator pushing operations. *International Journal of Robotics Research*, 5(3):53–71, 1986.

[10] B. Mirtich and J.F. Canny. Optimum force-closure grasps. Technical Report ESRC 93-11/RAMP 93-5, Robotics, Automation, and Manufacturing Program, University of California at Berkeley, July 1993.

[11] B. Mishra, J.T. Schwartz, and M. Sharir. On the existence and synthesis of multifinger positive grips. *Algorithmica, Special Issue: Robotics*, 2(4):541–558, November 1987.

[12] V-D. Nguyen. Constructing force-closure grasps. *International Journal of Robotics Research*, 7(3):3–16, June 1988.

[13] M.S. Ohwovoriole. An extension of screw theory. *Journal of Mechanical Design*, 103:725–735, 1981.

[14] M.A. Peshkin and A.C. Sanderson. Planning robotic manipulation strategies for workpieces that slide. *IEEE Journal of Robotics and Automation*, 4(5), 1988.

[15] J. Ponce. On planning immobilizing fixtures for three-dimensional polyhedral objects. In *IEEE Int. Conf. on Robotics and Automation*, pages 509–514, Minneapolis, MN, April 1996.

[16] J. Ponce, J.W. Burdick, and E. Rimon. Computing the immobilizing three-finger grasps of planar objects. In *1995 Workshop on Computational Kinematics*, pages 281–300, Nice, France, 1995.

[17] J. Ponce, S. Sullivan, A. Sudsang, J-D. Boissonnat, and J-P. Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *International Journal of Robotics Research*, 16(1):11–35, February 1997.

[18] F. Reulaux. *The kinematics of machinery.* MacMillan, NY, 1876. Reprint, Dover, NY, 1963.

[19] E. Rimon and A. Blake. Caging 2D bodies by one-parameter two-fingered gripping systems. In *IEEE Int. Conf. on Robotics and Automation*, pages 1458–1464, Minneapolis, MN, April 1996.

[20] E. Rimon and J. W. Burdick. Towards planning with force constraints: On the mobility of bodies in contact. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 994–1000, Atlanta, GA, May 1993.

[21] D. Rus. Dexterous manipulation of polyhedra. In *IEEE Int. Conf. on Robotics and Automation*, pages 2758–2763, Nice, France, June 1992.

[22] A. Sudsang and J. Ponce. In-hand manipulation: geometry and algorithms. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 98–105, Grenoble, France, September 1997.

[23] A. Sudsang, J. Ponce, and N. Srinivasa. Algorithms for constructing immobilizing fixtures and grasps of three-dimensional objects. In J.-P. Laumont and M. Overmars, editors, *Algorithmic Foundations of Robotics II*, pages 363–380. AK Peters, Ltd., 1997.