Algorithmica (2000) 26: 313–344
DOI: 10.1007/s004539910016

Algorithmica

Parts Feeding on a Conveyor with a One Joint Robot[1]

S. Akella,[2] W. H. Huang,[2] K. M. Lynch,[3] and M. T. Mason[4]

Abstract. This paper explores a method of manipulating a planar rigid part on a conveyor belt using a robot with just one joint. This approach has the potential of offering a simple and flexible method for feeding parts in industrial automation applications. In this paper we develop a model of this system and of a variation which requires no sensing. We have been able to characterize these systems and to prove that they can serve as parts feeding devices for planar polygonal parts. We present the planners for these systems and describe our implementations.

Key Words. Robotics, Manipulation, Mechanics, Planning, Minimalism, Automation, Manufacturing, Parts feeding.

1. Introduction. The most straightforward approach to planar manipulation is to use a rigid grasp and a robot with at least three joints, corresponding to the three motion freedoms of a planar rigid part, but three joints are not really necessary to manipulate a part in the plane. In this paper we achieve effective control of all three planar motion freedoms using a one joint robot working over a constant speed conveyor belt.

A central issue in this work is to develop a precise notion of "effective control" that is suited to the parts feeding application. Our robot cannot impart arbitrary motions to a part on the conveyor, but it does have a set of actions sufficient to position and orient a wide class of shapes. To make this precise we define the *feeding property*:

> A system has the *feeding property* over a set of parts $\mathcal{P}$ and set of initial configurations $\mathcal{I}$ if, given any part in $\mathcal{P}$, there is some output configuration $\mathbf{q}$ such that the system can move the part to $\mathbf{q}$ from any location in $\mathcal{I}$.

One of the goals of this paper is to demonstrate that a one joint robot can possess this property over a useful set of initial configurations and a broad class of part shapes. (We note that this definition of the feeding property can be generalized by considering *states* instead of configurations. We can also define a *stochastic* feeding property.)

The key to our approach is to use a single revolute joint to push the parts around on a constant speed conveyor belt. This approach, which we refer to as "1JOC" (one-joint-over-conveyor, pronounced "one jock") was initially conceived as a variation on the Adept Flex Feeder (see Figure 1). The Flex Feeder uses a system of conveyors to recirculate parts, presenting them with random orientation to a camera and robotic

[1] This work was supported by ARPA and NSF under Grant IRI-9318496.
[2] Department of Computer Science, Renselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180-3590, USA.
[3] Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, USA.
[4] The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA.

Received November 15, 1996; revised February 20, 1998. Communicated by R. Motwani and P. Raghavan.

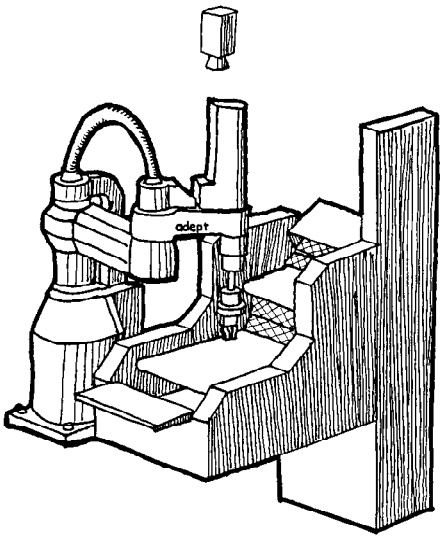


Fig. 1. The Adept Flex Feeder system. A SCARA robot picks parts off the middle of three conveyors. These three conveyors, along with an elevator bucket, circulate parts; an overhead camera looks down on the back-lit middle conveyor to determine the position and orientation of parts.

manipulator. Those parts that are in a graspable configuration may then be picked up by the robot and assembled into a product, placed in a pallet, or otherwise processed.

The question addressed in this paper is whether, at least in parts feeding applications, we can replace the SCARA robot with a simpler and more flexible robot, and also replace the Flex Feeder's servoed programmable conveyor with a fixed speed conveyor. Figures 2 and 3 show a possible variation on the Flex Feeder, where the SCARA robot has been replaced by a fence with a single revolute joint. By a sequence of pushing operations, punctuated by drift along the conveyor, the fence positions and orients a part and directs it into the entry point of a feeder track which carries the part to the next station.

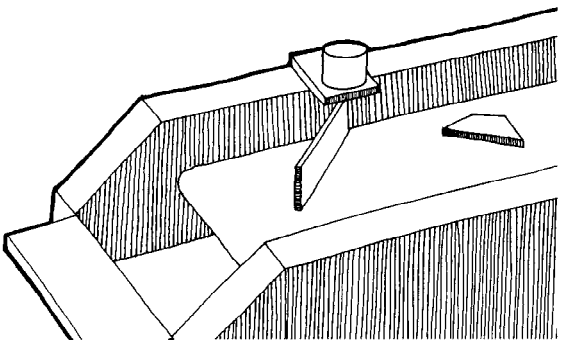


Fig. 2. The Flex Feeder with a rotatable fence.

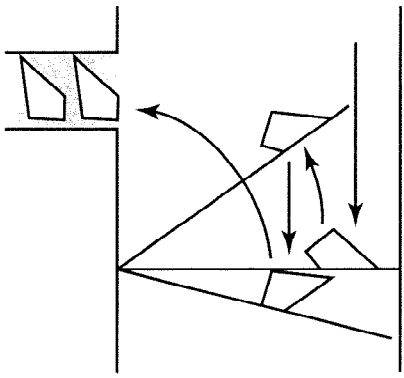


Fig. 3. We can feed a part by alternately pushing it with the fence and letting it drift along the conveyor.

There are many variations on this basic idea: a 2JOC, multiple 1JOCs working in parallel, curved fences, and so on. However, the purpose of this paper is an initial study of the fundamental characteristics of the idea, so we will focus on the simplest version as described above and in Figures 2 and 3. The main result for the 1JOC is:

> It is possible to move an arbitrary polygon from a broad range of initial configurations to a specific goal; and that goal can be chosen from a broad range of possible goals. (This is a generalization of the feeding property.)

The 1JOC uses a camera to sense the part's initial position. In the second half of the paper, we study a variant called the *sensorless* 1JOC that allows us to eliminate the camera. The main result for the sensorless 1JOC is:

> An arbitrary polygon can be moved from a broad range of initial configurations to a single goal configuration (up to certain symmetries in the pushing mechanics) without sensing.

The paper is organized as follows. After we review previous work and terminology from nonlinear control theory, Section 2 develops a progression of models leading to the 1JOC. The feeding property for the 1JOC is proved in Section 3, and the planner and the implementation are described in Sections 4 and 5. We then introduce the sensorless 1JOC in Section 6, and the planner and the implementation are described in Sections 7 and 8. Other possible variants are briefly discussed in Section 9.

1.1. *Previous Work.* We previously reported portions of this work in [1] and [2]. Below we discuss some related work on manipulation by pushing, parts feeding, underactuated systems, and minimalism.

1.1.1. *Manipulation by Pushing.* Our analysis of 1JOC is based on earlier work on the mechanics of pushing by Mason [47], [50], Peshkin and Sanderson [55], Goyal et al. [36], [37], Alexander and Maddocks [5], and Lynch and Mason [42], [43]. Pushing and squeezing have been used to orient, position, or grasp parts [45], [22], [49], [56], [34], [4] and to plan motions to "catch" an object by pushing it [23], [10].

1.1.2. *Parts Feeding.* Several parts feeding systems have already found success in industry. Boothroyd et al. [18] describe several. Bowl feeders vibrate in such a way that parts are forced to climb a narrow spiral track in single file. Along the spiral track is a sequence of obstacles that either reject incorrect orientations or turn the part to the correct orientation. Recent work focusing on automating the design of bowl feeders includes that by Boothroyd [17], Berkowitz and Canny [13], Caine [25], and Christiansen et al. [28].

In the SONY Advanced Parts Orienting System (APOS) [59] parts slide across a vibrating tilted tray with an array of depressions. The depressions and the vibration are designed so parts fall into and remain in the depressions if they are in the proper orientation, but otherwise are ejected and slide off the tray. Brost [23] explored a closely related idea, using configuration space to analyze the interactions of a part with a shape. A qualitative analysis of the APOS process is provided by Krishnasamy et al. [38].

Parts orienting by an arrangement of fixed fences over a conveyor was explored by Peshkin and Sanderson [56], and extended by Brokowski et al. [21] and Wiegley et al. [65].

Bowl feeders, APOS, and fixed fences all work by forcing the part to interact with shapes that are designed for the specific part. By contrast, 1JOC uses a generic shape (the fence) with a *motion* designed for the specific part. Erdmann and Mason [33] used a tilting tray as a generic shape, and computed part-specific motions. Christiansen [27] applied machine learning to the tray tilting approach.

Another method using generic shapes and part-specific motions is to squeeze a part with a parallel-jaw gripper. Goldberg [34] used a sequence of squeezing operations by a parallel-jaw gripper to orient parts. See [64] for further analysis of orienting parts by grasping and by fences.

Parts supported by a vibrating plate can be oriented by controlling the vibration [16], [62]. Böhringer et al. [15] are also studying the use of a planar array of micromanipulators to orient parts supported by the array.

1.1.3. *Underactuated Systems.* The 1JOC is an underactuated system with a drift field. Such systems have been studied in nonlinear control theory; see, for instance, [20] and [29]. For an introduction to nonlinear control see [53].

The 1JOC can be likened to an underactuated manipulator in a gravity field, where the proximal "shoulder" (fence pivot) is directly actuated, but the distal degrees of freedom (the object) are not. Research on the controllability of such serial link manipulators has been carried out by Oriolo and Nakamura [54], Arai et al. [8], [9], [6], and Bergerman et al. [12]. Sørdalen et al. [60] recently developed a nonholonomic gear which allowed them to construct a controllable n-link planar arm with only two motors.

Some other notable systems are the batting/juggling robots described by Bühler et al. [24], [58], which control the motion of a ball or puck bouncing on an actuated beam or paddle. A comparison of 1JOC and batting robots has prompted the observation that 1JOC juggles very slowly by using a conveyor instead of gravity.

Arai and Khatib [7] used a paddle held by a PUMA robot to demonstrate rolling of an object on the paddle. More recently, Lynch and Mason [44] explored dynamic tasks such as snatching, rolling, throwing, and catching using a single degree-of-freedom arm working in the vertical plane.

Brock [19] took advantage of gravity and contact with other objects to reorient objects within the hand using controlled slip. Rao et al. [57] used pivoting between fingers of a robot hand in order to control more degrees of freedom. By rolling an object between two parallel palms, Bicchi and Sorrentino [14] showed that the spatial position and orientation of the object can be controlled by as few as three motors.

Manikonda and Krishnaprasad [46] studied controlling the position and orientation of a hovercraft with a single thruster, a situation closely related to manipulating a planar object, except our "thrusters" are unilateral frictional contacts.

1.1.4. *Minimalism.* Much of the work reported above shares a common interest in minimalism: exploring the capabilities of systems with fewer sensors, fewer motors, or limitations in computation or communication [15]. Donald et al. [30] have begun to develop a theory of information invariants to compare the information content embedded in various subsystems, such as sensing, actuation, computation, and communication. Erdmann [32] proposed that the true test of a sensor is whether it gives information that allows the system to proceed toward the goal. *Progress cones* bound the set of directions that move toward the goal, and these progress cones can be used to design the simplest effective sensors. Canny and Goldberg [26] argue that RISC (reduced intricacy in sensing and control) robotics using simple devices results in cheaper, more flexible systems. McGeer's walking machines [51], [52] are dramatic examples of minimalism; a stable walking gait is obtained without sensors, actuators, or computation.

1JOC may be viewed as a minimalist system in two respects. Besides using very few effectors, it does not use grasping. Much of the work reported above also exploits graspless or *nonprehensile* manipulation. Other examples include sliding and rolling manipulation with two flat "palms" [31], [66] or with the surfaces of a multibody manipulator [63].

1.2. *Terminology.* The central issue of this paper is showing whether the 1JOC—an underactuated system—is "rich" enough to serve as a parts feeder. To characterize the system precisely, we borrow definitions of reachable sets from nonlinear control theory [61].

A part's configuration $\mathbf{q} = (x, y, \varphi)^T$ is *controllable from* $\mathbf{q}$ if, starting from $\mathbf{q}$, the part can reach every configuration in the configuration space. The part is *controllable to* $\mathbf{q}$ if $\mathbf{q}$ is reachable from every configuration. The part is *accessible from* $\mathbf{q}$ if the set of configurations reachable from $\mathbf{q}$ has nonempty interior in the configuration space.

We can also define local versions of these properties. The part is *small-time accessible from* $\mathbf{q}$ if, for any neighborhood U of $\mathbf{q}$, the set of reachable configurations without leaving U has nonempty interior. The part is *small-time locally controllable from* $\mathbf{q}$ if, for any neighborhood U of $\mathbf{q}$, the set of reachable configurations without leaving U contains a neighborhood of $\mathbf{q}$.

The phrases "from $\mathbf{q}$" and "to $\mathbf{q}$" can be eliminated in these definitions if they apply to the entire configuration space.

In these terms, if the configuration of the part is accessible, we know that the 1JOC has "enough" degrees of freedom—it can transfer the part from the initial configuration to a three-dimensional subset of the configuration space. Better yet, if the part is controllable, the 1JOC can transfer the part from any configuration to any other configuration. Small-

time local controllability, an even stronger condition, implies that the part can follow any path arbitrarily closely. This is a useful property for repositioning parts under time and workspace constraints.

Although controllability and small-time local controllability guarantee the ability to feed parts, they are not absolutely necessary. The minimum we require of a parts feeding system is the feeding property—the parts must be controllable to a single configuration from a set of initial configurations.

2. A Progression of Models. The 1JOC approach arises from a desire to explore the simplest mechanisms to accomplish a task. Planar parts have three degrees of freedom, suggesting the use of a robot with three or more actuated joints. However we can easily see that two controls suffice: a car (with only a steering wheel and accelerator) can be arbitrarily positioned in the plane. In fact, a car can be arbitrarily positioned in the plane even if the steering wheel has only two settings. Each setting of the steering wheel defines a rotation center of the car. By driving appropriate distances alternating between the two rotation centers, the car can be driven to any location in the plane.

These two rotation centers move with the car. To see that two *fixed* rotation centers suffice to produce arbitrary motions of the plane, adopt the car's viewpoint. From the car's point of view, the two rotation centers are fixed, and the wheels drive the plane around. Thus it is apparent that two fixed rotation centers can arbitrarily position a part in the plane.

In this section we start from this basic idea of using fixed rotation centers to manipulate a part in the plane and develop a progression of models which lead to the 1JOC model. These idealized models provide some insight into the capabilities of simple robotic manipulators.

2.1. *One Rotation Center.* First we consider an idealization of a rotating fence. We envision an infinite turntable to which the part can be affixed. This turntable can give the part an arbitrary angular velocity about its pivot (rotation center).

From any initial point in the part configuration space, the reachable set is one-dimensional. If we consider the full pushing mechanics for a rotating fence, the reachable set is still just one-dimensional. If the fence can swing all the way around and push in the opposite direction, the part's configuration may be accessible, but we will not consider this case further.

2.2. *Two Rotation Centers*. Now consider a pair of overlapping (yet independent) ideal turntables. At any given instant, the part is affixed to one of the two turntables and rotates about the center of that turntable. We assume that we can instantaneously switch the part from one turntable to the other.

There are two cases to consider:

- Both turntables are bidirectional. If each turntable can be driven in either forward or reverse, the situation is similar to driving a car with two different steering angles. It is well known that this system is small-time locally controllable and can approximate an arbitrary trajectory as closely as desired.

- One or both turntables are unidirectional. Now the situation is analogous to a car without reverse. The system is controllable, but not small-time locally controllable. It may be necessary to take a large excursion to accomplish a small motion.

We can model the conveyor as the limiting case of a turntable whose pivot approaches infinity. As above, the system is small-time locally controllable if both the turntable and conveyor are bidirectional, and it is simply controllable if either or both is unidirectional.

For this simple system, we can study controllability by considering the vector fields X_1 and X_2 corresponding to the conveyor and the turntable:

$$X_1 = (0, 1, 0)^T, \tag{1}$$

$$X_2 = (-y, x, 1)^T. \tag{2}$$

When the part tracks the conveyor, the part motion is given by $\dot{\mathbf{q}} = vX_1$, where v is the velocity of the conveyor in the y direction. When the part tracks the turntable, the part motion is given by $\dot{\mathbf{q}} = \omega X_2$, where ω is the angular velocity of the turntable and the origin is at the center of the turntable. The Lie bracket $[X_1, X_2]$ of these two vector fields is given by

$$\begin{aligned}[X_1, X_2] &= \frac{\partial X_2}{\partial \mathbf{q}} X_1 - \frac{\partial X_1}{\partial \mathbf{q}} X_2 \\ &= \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}. \end{aligned} \tag{3}$$

The new vector field $[X_1, X_2]$ is linearly independent of X_1 and X_2, yielding a third controlled freedom (provided both v and ω can be nonzero). The system therefore satisfies the Lie Algebra Rank Condition, and it is small-time accessible. If both the conveyor and the turntable are bidirectional (both v and ω can be positive or negative), the system is also small-time locally controllable [53]. If either or both is unidirectional, the system is not small-time locally controllable, but it can be shown to be controllable by a simple constructive argument [43].

The 1JOC system consisting of a conveyor and a rotating fence resembles the system above with a unidirectional conveyor. One difference is that rotating the part about the pivot is only practical on a subset of the configuration space, specifically when an edge of the part is aligned with the fence.

2.3. *Detailed Pushing Model.* We now add pushing mechanics to the model. We assume that the friction coefficient at the pushing contact and the distribution of support forces are known, and we place no restriction on the fence motions. Although this model makes unrealistic assumptions about the available information, any negative results will also apply to less detailed models.

Figure 4 shows that the system is not small-time locally controllable for the case of a thin rod. If the fence pushes from below, the rod's angular velocity is positive. If the fence swings around and pushes the part from above, the angular velocity is still positive.

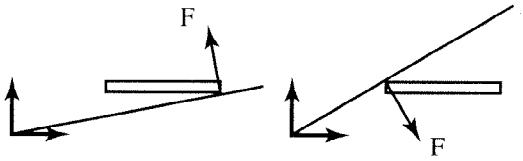


Fig. 4. The thin rod is not small-time locally controllable from this configuration—regardless of whether the fence pushes the rod from above or below, the resulting force can produce only positive torque.

There is no maneuver combining these actions and conveyor drift that can locally achieve negative rotations.

2.4. *Practical Pushing Models*. The detailed pushing model is difficult to use because it requires full knowledge of the pressure distribution. We can define more abstract models by restricting the allowable actions. In particular, we would like to use actions with predictable outcomes that do not impose unrealistic demands for information.

Any abstraction of the detailed pushing model will inherit its limitations, thus it is immediately clear that small-time local controllability is impossible.

3. The 1JOC. In this section we focus on a particular model for the 1JOC and prove that it can transfer any polygonal part from any initial configuration (sufficiently far from the edges of the conveyor) to a single goal configuration. To prove this property, we identify a subset of the actions available to the 1JOC and show that they are sufficient to demonstrate the feeding property.

3.1. *The* 1*JOC Model*. The conveyor is the half-plane $x > 0$, with a constant drift velocity v in the $-y$ direction. The fence is a line that pivots about a fixed point on the line. The origin of a fixed world frame $\mathcal{F}_w$ coincides with the pivot point. The fence angle θ is measured with respect to the x axis of $\mathcal{F}_w$, and its angular velocity is given by ω. The fence angular velocity ω is our single control input.

The part can be any polygon, but because the manipulator is a line, we need only consider its convex hull. The center of mass of the part lies in the interior of the convex hull and is at a known location. We assume that the support pressure distribution is unknown and that Coulomb friction applies to the contact between polygon and conveyor, with a uniform coefficient of friction. The *center of friction* is therefore coincident with the center of mass. When we refer to a "part" in the rest of the paper, we assume these properties. For the 1JOC, we also assume the coefficient of friction between the fence and the part is nonzero.

The analysis relies on the definitions of a *stable edge* and a *metastable edge*. An edge of a polygon is a *stable edge* if the perpendicular projection of the center of friction to the edge is in the interior of the edge segment. An edge of a polygon is a *metastable edge* if the center of friction projects to an endpoint of the edge.

We assume quasistatic mechanics—as the fence pushes the part over the conveyor, the motion of the part in the world frame $\mathcal{F}_w$ is sufficiently slow that inertial forces are negligible compared with the frictional forces. The support friction acting on the part

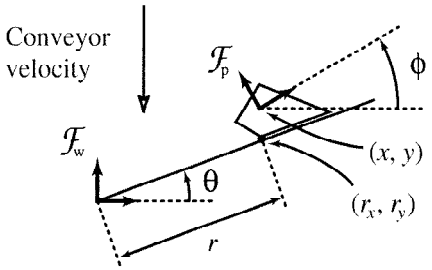


Fig. 5. Notation.

during pushing is determined by the motion of the part relative to the conveyor, not the world frame $\mathcal{F}_w$.

It is convenient to represent the linear and angular velocity of the part relative to the conveyor in a frame $\mathcal{F}_p$ attached to the center of friction of the part. A velocity direction is simply a unit velocity vector. Velocity directions may also be represented as rotation centers in the frame $\mathcal{F}_p$.

The configuration of the part frame $\mathcal{F}_p$ in the world frame $\mathcal{F}_w$ is given by $(x, y, \varphi)^T \in \Re^2 \times \mathcal{S}^1$. When an edge of the part is aligned with the fence, the contact radius r defines the distance from the fence's pivot point to the closest point on the edge (the *contact vertex*), and the configuration of the part can be given in the polar coordinates (r, θ). The location of the point at r on the fence is (r_x, r_y) in the world frame. See Figure 5.

3.2. 1*JOC Primitives*. We now describe the basic 1JOC primitives (illustrated in Figure 6) which yield the feeding property. The starting point for these primitives is with the fence held at 0 degrees (perpendicular to the conveyor velocity) with the part in stable contact with the fence.

3.2.1. *Stable Pushes*. A stable push occurs when one edge of the part is aligned with the fence and frictional forces keeps the part fixed against it. The simplest example of

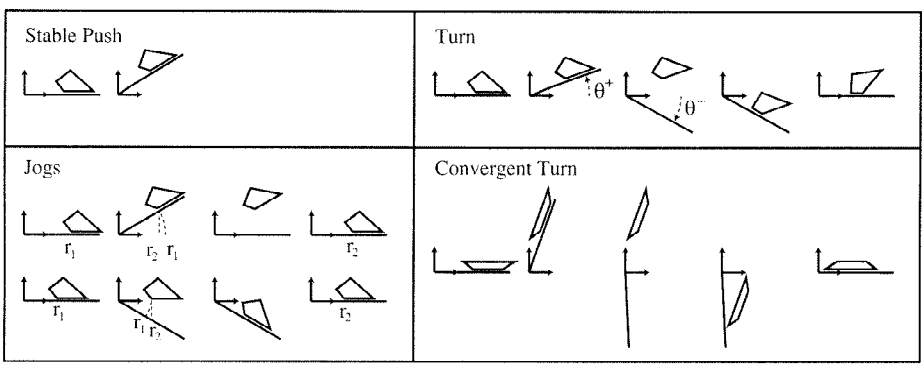


Fig. 6. Step by step illustrations of the 1JOC primitives.

a stable push occurs when the fence is held at 0 degrees, and a part on the conveyor drifts into contact and comes to rest on the fence. When the part is at rest, the fence is executing a stable push.

Of course, a stable push may also occur while the fence is rotating. Lynch and Mason [42] described the procedure STABLE that finds a conservative set of stable pushing directions $\mathcal{V}_{\text{stable}}$ for a given pushing edge, pushing friction coefficient, and center of friction of the part. This set of pushing directions is fixed in the part frame $\mathcal{F}_p$. The fence is guaranteed to execute a stable push if the edge is aligned with the fence and the motion of the fence and the conveyor combine to yield a rotation center in $\mathcal{V}_{\text{stable}}$. See Figure 7 for details.

For simplicity, we only consider the edges of the part that yield a stable push while the fence is held at 0 degrees (the pushing direction is a translation normal to the edge). The existence of these stable edges is indicated by the following lemma [42]:

LEMMA 1. *All polygonal parts have at least one stable edge. For a stable edge, the normal translational pushing direction, along with a neighborhood of this pushing direction, belongs to* $\mathcal{V}_{\text{stable}}$ (*because friction is nonzero*).

REMARK. We avoid metastable edges by slightly perturbing the fence (after the part comes to rest on such an edge) to bring the part to a stable edge.

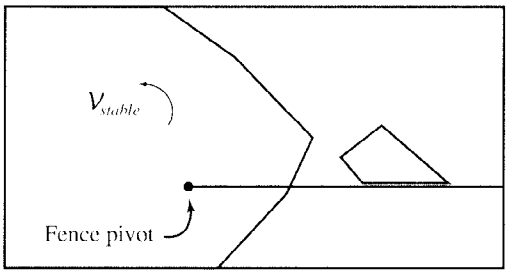
If the contact radius r is sufficiently large, the fence pivot will be inside the set of stable rotation centers $\mathcal{V}_{\text{stable}}$ fixed in the part frame $\mathcal{F}_p$. If the fence angular velocity ω is large enough relative to the conveyor velocity v, the combined pushing motion is nearly a pure rotation about the pivot, and the pushing motion is stable. Therefore, it is possible to stably push the part from any θ_0 to any θ_1 in a counterclockwise (CCW) direction, where $-\pi/2 < \theta_0 < \theta_1 < \pi/2$. Figure 8 illustrates a method for finding the *minimum stable radius*: the minimum contact radius r such that the push is stable for all fence angles in the range $[-\pi/2, \pi/2]$ for a given fence angular velocity ω and conveyor velocity v.

There are two other types of stable pushes that are possible but which will not be used in this paper: (1) The fence rotates CCW, but the fence pivot is not in $\mathcal{V}_{\text{stable}}$. Such a push is stable if the rotation center remains in $\mathcal{V}_{\text{stable}}$ during the push. The fact that $\mathcal{V}_{\text{stable}}$ is a conservative approximation to the set of stable pushing directions may also help. (2) The fence rotates clockwise (CW). This type of push may be stable if the magnitude of the fence angular velocity is small enough with respect to the conveyor velocity.

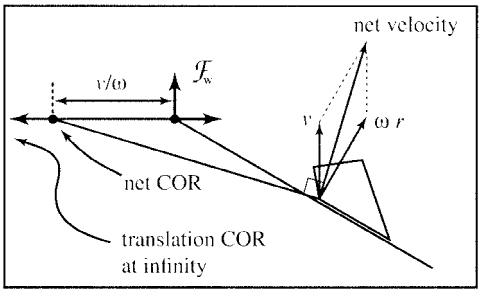
3.2.2. *Jogs.* Jogs are simple maneuvers that allow us to change the contact radius r from the fence pivot to the part without changing the stable resting edge.

LEMMA 2. *A part resting on a stable edge can be moved from any initial contact radius* r *on the fence to any desired* r *in the range* $(0, \infty)$ *by a series of jogs.*

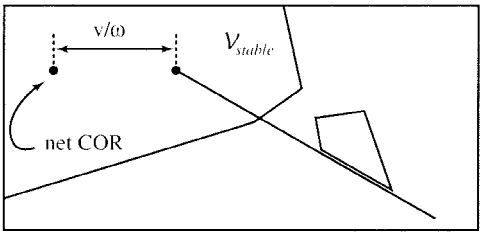
PROOF. From Lemma 1, we use the fact that a neighborhood of pushing directions about the normal translation direction is stable for a stable edge. In fact, there is a range $[\theta^{\min}, \theta^{\max}]$ of fixed fence angles about 0 degrees that are also stable. The fence does not



For a given edge contact, we can determine $\mathcal{V}_{\text{stable}}$, the set of pushing rotation centers that keep the part fixed to the pusher. Note that $\mathcal{V}_{\text{stable}}$ is fixed in the part frame, not the world frame.



The conveyor velocity v and the rotational velocity ω about the fence pivot combine to form a net velocity (relative to the conveyor) at each point. This can be expressed as a net center of rotation (COR), which lies on the line through the fence pivot and perpendicular to the conveyor velocity.



Provided the net COR is contained in $\mathcal{V}_{\text{stable}}$, the part will remain stationary with respect to the fence for that combination of conveyor and fence velocities. Note that as the fence rotates, the conveyor velocity changes direction with respect to the part, and the net COR in the part frame rotates about the fence pivot at a radius of v/ω.

Fig. 7. Determining the stability of a push.

have to be motionless to execute a stable push, however; if the fence angular velocity ω is small enough with respect to the conveyor velocity v, then the fence can execute a stable push while moving in the angle range $[\theta^{\min}, \theta^{\max}]$, regardless of the contact radius r.

To decrease the contact radius r, the fence is raised to an angle θ^+ ($\theta^{\max} \geq \theta^+ > 0$), keeping the part fixed to the fence by a stable push. The fence then drops to 0 degrees, immediately releasing the part. The part drifts back into contact with the fence and settles

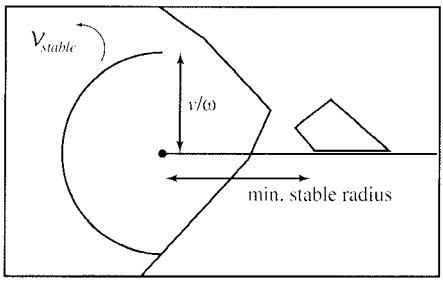


Fig. 8. The locus of net CORs forms a semicircle about the fence pivot as the fence swings from $-\pi/2$ to $\pi/2$. Given a fence angular velocity ω and a conveyor velocity v (which determine the radius of this semicircle), we pick a minimum stable radius so that this semicircle lies completely within $\mathcal{V}_{\text{stable}}$. Recall that $\mathcal{V}_{\text{stable}}$ is fixed with respect to the part, so increasing the contact radius moves the semicircle deeper into $\mathcal{V}_{\text{stable}}$. This conservative estimate of the minimum stable radius guarantees stable pushing in the CCW direction at any fence angle in $[-\pi/2, \pi/2]$.

on the same stable edge. We assume there is no slip between the part and the fence. (The no-slip assumption is important for open-loop feeding plans.) The result of the jog is to decrease the contact radius from r to $r \cos\theta^+$.

To increase the contact radius r, the fence is quickly lowered to an angle θ^- ($\theta^{\min} \leq \theta^- < 0$), releasing the part. The part drifts back into contact with the fence and settles on the same stable edge. The fence is then raised to 0 degrees again, pushing the part with a stable push. The contact radius of the part has changed to $r/\cos\theta^-$.

A series of jogs, either increasing or decreasing r, can bring the part to any r in the range $(0, \infty)$. □

There are other possibilities for jogging a part which will not be considered in this paper: (1) By moving the fence quickly, it may be possible to raise the fence to an angle $\theta^+ > \theta^{\max}$, in some cases allowing larger inward jog distances. Similarly, it may be possible to obtain a larger outward jog by recontacting the part at an angle $\theta^- < \theta^{\min}$ (while the fence is moving). (2) If an edge is stable against a fixed fence in an angle range that does not include 0 degrees, it may be jogged by a procedure similar to that above. In these cases, however, the contact radius r can only be increased or decreased, not both.

A jog is directly analogous to the Lie bracket motion for the ideal turntable plus conveyor system analyzed in Section 2.2. The difference from the ideal system is that the part only tracks the fence when it is in stable contact. Nonetheless, we can now show:

THEOREM 1. *The configuration of a polygonal part in stable edge contact with a fence held at* 0 *degrees is small-time accessible using stable pushes, jogs, and conveyor drift.*

PROOF. Given any neighborhood of the part configuration while it is in stable edge contact with the fence at 0 degrees, it is possible to jog the part a nonzero distance in both directions without leaving this neighborhood. The effect of the jog is to change the x coordinate of the part configuration, exactly like the Lie bracket motion of Section 2.2.

The other two vector fields of Section 2.2 can be obtained directly by conveyor drift and stable pushes. □

We call $\mathcal{I}$ the set of initial part configurations that drift to stable edge contact with the fence at 0 degrees. From a stable edge contact, Theorem 1 indicates that the configuration of the part is small-time accessible. Therefore, the configuration of the part is accessible from the set $\mathcal{I}$.

We still have not proven the feeding property. To get this property, we need the ability to turn a part to a new stable edge.

3.2.3. *Turns*. Turns allow us to change the edge of the part aligned with the fence, subject to the constraint that the initial and final edges are both stable edges. A turn consists of raising the fence to an angle $\theta^+ \geq 0$ by executing a stable push; dropping the fence to an angle $\theta^- \leq 0$, which releases the part; and reacquiring the part on the new edge with a stable push, raising the fence back to 0 degrees. The final push commences at the moment the new edge contacts the fence.

LEMMA 3. *Any polygonal part has at most one stable edge from which it is impossible to turn the part to the next CW stable edge.* Exception: *a part has two such stable edges if they are the only stable edges and they are parallel to each other.*

PROOF. The fence can execute a stable push to any angle less than $\pi/2$, and it can reacquire the part with a stable push at any angle greater than $-\pi/2$. This indicates that it is possible to reach any final stable edge less than π CW of the initial edge. If the part has two or more stable edges, there can only be one stable edge from which the next CW stable edge is greater than π removed. The exception is a part which has two stable edges that are exactly π removed from each other. □

There are a few important points to make about turns:

- The contact radius r, for both the initial and final edges, must be large enough that the pushing directions always remain in $\mathcal{V}_{\text{stable}}$. Jogs may be used before the turn to satisfy this condition.
- The quantity $\theta^+ - \theta^-$ is determined by the part geometry, but the actual values of θ^+ and θ^- are not. To some extent, the contact radius r after the turn can be controlled by the choice of these values. If $\theta^+ - \theta^- \geq \pi/2$, then r can be changed to any value in the range $(0, \infty)$ during the turn.
- This primitive requires precise knowledge of the conveyor's motion—the final stable push must begin precisely when the new edge contacts the fence.

Consider the case where turning a part to a new stable edge requires raising the fence to $\pi/2 - \delta$ and catching the part at $-\pi/2 + \varepsilon$, where δ and ε are small positive values. It may be necessary to move the part to a large contact radius r before executing the turn to ensure that the part remains completely on the conveyor ($x > 0$) after the initial push.

3.2.4. *Convergent Turns*. To perform a turn between two stable edges which are parallel, the fence would have to be raised to $\pi/2$, pushing the part off the conveyor. To

change the resting edge of such parts, we can introduce a convergent turn: raise the fence to $\pi/2 - \delta$, allow the part to drift down, and begin pushing it again at $-\pi/2 + \varepsilon$. Assuming no slip at the pushing contact, the part will converge to the new stable edge as the fence is raised to 0 degrees. Convergent turns allow us to strengthen Lemma 3 to apply to all polygonal parts, with no exceptions. In this paper, the sole function of convergent turns is to handle parts with only two stable edges which are parallel.

3.3. *The Feeding Property.* Using the lemmas proven above, we can now demonstrate the feeding property for the 1JOC.

THEOREM 2. *The* 1*JOC possesses the feeding property*:

- *for all polygonal parts,*
- *for the three-dimensional space $\mathcal{I}$ of initial configurations such that the part initially comes to rest on a stable edge, completely on the conveyor, against the fence fixed at* 0 *degrees, and*
- *using only stable pushes, jogs, turns, convergent turns, and conveyor drift.*

Furthermore, the goal configuration can be chosen from a three-dimensional subset of the configuration space.

PROOF. If the part has a stable edge such that it cannot be turned to a new stable edge, then this edge must be chosen as the goal edge. Otherwise, any stable edge can be chosen as the goal edge. By Lemma 3 (strengthened by the convergent turn of Section 3.2.4), this goal edge can be reached using turns. Once at the goal edge, the part can be jogged to any contact radius r in the range $(0, \infty)$ by Lemma 2), and the part is stable against the fence for a range of fence angles θ by Lemma 1. The goal edge therefore allows a two-dimensional set of final stable configurations of the part in the (r, θ) space. If the fence is quickly lowered to $-\pi/2$, releasing the part, this set drifts to a set of reachable configurations with nonempty interior in the world configuration space. Any configuration in this set can be chosen as the goal. □

If the application of the 1JOC is to feed parts into a feeder track, then the fence can be rotated $\pi/2$, pushing the part off the conveyor into the feeder track, instead of releasing it to continue on the conveyor.

4. The 1JOC Planner. Theorem 2 establishes that for any polygonal part, there exists a sequence of jogs and turns to feed it. In this section we describe how to find a sequence of jogs and turns to feed a given part in the minimum amount of time.

We assume that we know the shape and center of friction of the part, the coefficient of friction between the fence and part, and the conveyor velocity. For simplicity, we pick the fence angular velocity to be one of three discrete values: ω_{stable} (for stable pushes), ω_{drop} (when lowering the fence), and zero.

Although the 1JOC model of Section 3 assumes an infinite fence length and an infinite conveyor half-plane, we can always find a finite dimension fence and conveyor to feed a given part.

A singulated part drifts down the conveyor until it comes to rest against the fence. The feeding problem consists of getting the part from this start configuration to a goal configuration, where the configurations are specified by the edge aligned with the fence, and the contact radius (distance from pivot to the closest vertex of the resting edge). We assume that the initial fence orientation is zero, and, for convenience, that the final fence orientation is zero as well.

Turns rotate the part CCW so the new edge aligned with the fence is CW from the initial edge. Turns may skip over one or more edges, so we must consider all CW sequences of edges beginning with the start edge e_s and ending at the goal edge e_g that do not require rotating the part more than 2π. For a sequence of edges S, a feasible plan exists if there is a valid sequence of jogs and turns to move the part in sequence through the edges in S and change the contact radius by the required amount. Of all feasible plans, we select the one that requires minimum time.

4.1. *A Simple Example.* Consider a part resting on edge e_j (orientation φ_j) at radius r_j which is to be moved to the neighboring CW stable edge e_k (orientation φ_k) at radius r_k (Figure 9). Here the sequence of edges is $S = \{e_j, e_k\}$. Assume r_j and r_k are greater than the minimum stable radii for the corresponding edges, and are close enough in magnitude that no jog is required. Here finding a plan consists of determining the fence angles θ^+ and θ^- for the turn. From the geometry, we have

$$r_k = r_j \frac{\cos\theta^+}{\cos\theta^-} - l_k, \tag{4}$$

$$\theta^+ - \theta^- = \varphi_k - \varphi_j. \tag{5}$$

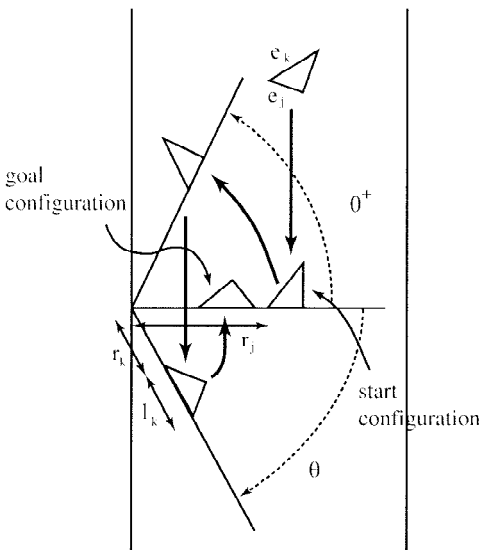


Fig. 9. A feeding plan to move a triangle from start configuration (e_j, r_j) to goal configuration (e_k, r_k) by a turn with raise and drop angles of θ^+ and θ^-. Length of edge e_k is l_k.

Solving these equations, we find

$$\theta^- = \tan^{-1}\left(\cot(\varphi_k - \varphi_j) - \frac{r_k + l_k}{r_j \sin(\varphi_k - \varphi_j)}\right), \tag{6}$$

$$\theta^+ = \varphi_k - \varphi_j + \theta^-. \tag{7}$$

Note that when $(\varphi_k - \varphi_j) \geq \pi/2$, the radius change can be arbitrarily large or small.

4.2. *Nonlinear Programming Approach.* For a given sequence of edges, we must determine the parameters for each edge transition. For each turn, we must ensure that the part has a contact radius that is no less than the minimum stable radius for the particular edge in contact with the fence. If this is not the case, the part must be translated prior to the turn. Such a *translation* may consist of a single jog or a series of jogs.

A feeding plan to transfer a part from its start configuration (e_s, r_s) to its goal configuration (e_g, r_g) consists of several stages, each of which accomplishes an edge transition. Each stage consists of a translation followed by a turn; the translation for a given stage may be zero. A plan with the edge transition sequence S will take $n - 1$ stages, where $n = |S|$. A final translation may also be required after the goal edge has been reached.

Consider the ith stage in a plan to accomplish the transition from edge e_j to edge e_k (Figure 10). To perform a turn, we must execute a translation of g_i to move the part to a contact radius of r_i^+ that is greater than r_j^{stable}, the minimum stable radius for edge e_j. We then perform a turn with a raise angle of θ_i^+ and a drop angle of θ_i^-. The stable push at the end of the turn brings the fence to $\theta = 0$ with the new contact edge e_k. For this stable push on edge e_k, the contact radius r_i^- must be greater than the minimum stable radius for e_k, r_k^{stable}. We also have to ensure that the fence raise and drop angles are in valid ranges, the part is within fence and conveyor bounds, and the fence contacts the part only at the end of the drift phase.

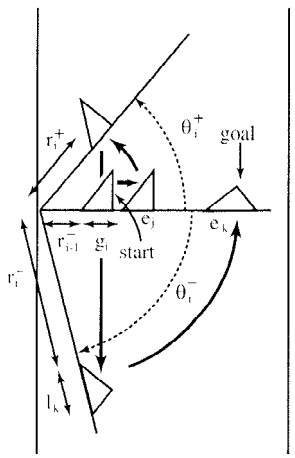


Fig. 10. The ith transition from edge e_j to edge e_k with a translation of g_i, start and end turn radii r_i^+ and r_i^-, and raise and drop angles θ_i^+ and θ_i^-.

The resulting constraints are (for $i = 1, \ldots, n-1$):

$$r_i^+ = r_{i-1}^- + g_i, \tag{8}$$

$$r_i^- = \frac{(r_i^+ - d_j)\cos\theta_i^+}{\cos\theta_i^-} - (l_k + d_k), \tag{9}$$

$$\theta_i^+ - \theta_i^- = \varphi_k - \varphi_j, \tag{10}$$

$$r_j^{\max} \geq r_i^+ \geq r_j^{\text{stable}}, \tag{11}$$

$$r_k^{\max} \geq r_i^- \geq r_k^{\text{stable}}, \tag{12}$$

$$\frac{\pi}{2} > \theta_i^+ \geq 0, \tag{13}$$

$$0 \geq \theta_i^- > -\frac{\pi}{2}, \tag{14}$$

$$(r_i^+ + a_j^v)\cos\theta_i^+ - p_j^v \sin\theta_i^+ > 0, \qquad \forall v \in V, \tag{15}$$

$$\omega_{\text{drop}} r_i^+ \cos\theta_i^+ > v_c, \tag{16}$$

$$\frac{(r_i^+ - d_j)\sin\theta_i^+ - (r_i^- + l_k + d_k)\sin\theta_i^-}{v_c} > \frac{\theta_i^+ - \theta_i^-}{\omega_{\text{drop}}}, \tag{17}$$

where l_k is the length of edge k, φ_j and φ_k are the orientations of edges e_j and e_k, v_c is the conveyor velocity, $r_j^{\max}$ and $r_k^{\max}$ are maximum valid contact radii for edges e_j and e_k, V is the set of part vertices, and a_j^v and p_j^v are the components along and perpendicular to edge e_j of the vector from the contact vertex to a part vertex v. If e_j and e_k are neighboring edges, $d_j = d_k = 0$. Else, d_j and d_k are the distances from the (virtual) intersection vertex of (extended) edges e_j and e_k to the contact vertices of these edges.

The start and goal constraints are

$$r_0^- = r_s, \tag{18}$$

$$r_g = r_{n-1}^- + g_n. \tag{19}$$

These define a set of nonlinear constraints over the variables g_i, θ_i^+, θ_i^-, r_i^+, and r_i^-. Since the radius at the start of a turn depends on the radius at the end of the previous turn, an optimal solution must consider all variables simultaneously.

The time taken for each stage of the plan is the sum of the jog time, fence raise time, drift time on the conveyor, and fence recovery time. We use an approximation to the total time of the plan as our objective function:

$$\sum_{i=1}^{n-1}\left[t_i g_i^2 + \frac{\theta_i^+}{\omega_{\text{stable}}} + \frac{(r_i^+ - d_j)\sin\theta_i^+ - (r_i^- + l_k + d_k)\sin\theta_i^-}{v_c} - \frac{\theta_i^-}{\omega_{\text{stable}}}\right] + t_n g_n^2. \tag{20}$$

The approximate time cost coefficient of the ith jog series is $t_i = k \cos\alpha_i / v_c(r_i^+ + r_{i-1}^-)$ where k is a constant chosen to be 1000, and $\alpha_i = \min(|\theta_i^{\min}|, |\theta_i^{\max}|)$ is a safe jog angle for the ith jog series.

This is a nonlinear programming problem whose solutions yield valid feeding plans and minimize the total time of the plan. Once we have found a solution, we must still determine a series of jogs to execute the translations g_i.

An inward translation distance $g_i < 0$ requires a series of decreasing jogs, such that $r_i^+ = r_{i-1}^-(\cos\gamma_i)^m \cos\rho_i$ where m is a nonnegative integer, and $\theta_i^{\max} \geq \gamma_i \geq \rho_i \geq 0$. So this radius decreasing translation is accomplished by a series of m jogs of angle γ_i followed by a jog of ρ_i. Similarly, an outward translation distance $g_i > 0$ requires a series of increasing jogs, such that $r_{i-1}^- = r_i^+(\cos\gamma_i)^m \cos\rho_i$ where m is a nonnegative integer, and $0 \geq \rho_i \geq \gamma_i \geq \theta_i^{\min}$.

A feasible solution always exists if the fence is longer than the minimum stable radius of all edges. A plan with a maximum of three stages to get the part from the start to the goal configuration exists, although it may not be optimal.

4.3. *Details of Generating Feeding Plans.* We now outline the process of generating feeding plans. From the part description and coefficient of friction, we compute the minimum stable radii for the stable edges and the transition angles between these edges. For the start and goal configurations, we generate candidate edge transition sequences. For each such sequence, the corresponding objective function and constraints are input to a commercial nonlinear programming package called GINO [40]. See the example plans in Figure 11.

We use minimum stable radius values empirically determined to be robust instead of the conservative estimate described in Figure 8. Also, the planning procedure can be extended to use convergent turns to handle parts with only two stable edges which are parallel.

5. 1JOC Implementation.

We have implemented several feeding plans on a conveyor with the fence being (somewhat ironically) actuated by one joint of an Adept 550 SCARA

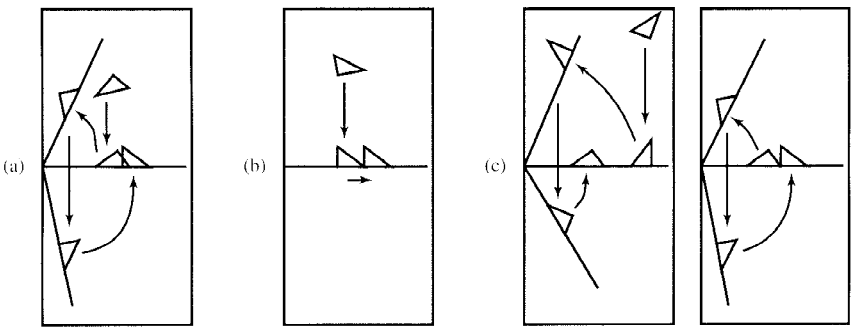


Fig. 11. Example feeding plans for a triangle. The goal configuration ($\varphi_g = 269^\circ$, $r_g = 150$) is the same, while the start configurations are different: (a) ($\varphi_s = 126^\circ$, $r_s = 100$) (b) ($\varphi_s = 269^\circ$, $r_s = 100$) (c) ($\varphi_s = 0^\circ$, $r_s = 200$).

robot. The conveyor speed and fence rotational speed have been selected to be 20 mm/s and 30 degrees/s respectively. The fence is covered with a foam material to avoid slip between the part and the fence.

Figure 11 illustrates three plans we ran that achieve the same goal configuration from different starting configurations. Plan (a) requires a single turn to reach the goal, plan (b) requires just a translation, and plan (c) requires two turns. These plans had maximum position errors at the goal configuration of 3 mm. The final position is sensitive to part shape measurements, fence turn angles, and timing in the turns. Occasional slip also causes some error.

We have implemented the 1JOC planner in C++. We provide the planner with the part shape, center of mass position, empirically estimated minimum stable radii for the stable edges, and the start and goal part edges and contact radii. The planner computes the necessary part information and for each feasible sequence of edges traversed using turns, generates the nonlinear constraints automatically and runs the nonlinear programming package GINO on the constraints. It then identifies the best plan as defined by the objective function. Currently the planner can automatically generate one and two stage plans. The average planner run times were 1.27 s for single stage plans and 1.62 s for two stage plans over a set of 60 trials each on a Sun Sparcstation20.

An overhead camera takes a picture to determine the pose of a the part along the fence. The configuration is passed to our planner, which finds a plan and then generates a V+ program for the Adept robot to execute. The system takes 5–10 s from the time the camera takes an image to the time the robot starts executing the generated plan. This time could be significantly reduced with a few optimizations. Note that the feeding plan to get the part to the goal configuration requires no sensing after the initial camera snapshot.

6. The Sensorless 1JOC.

We have so far assumed that a camera gives us the initial position and orientation of the part after it first contacts the fence. With some modifications, we can perform sensorless parts feeding with a one joint system over a conveyor. We call this variant the sensorless 1JOC. We show that any polygon can be transferred from a random initial configuration to a suitable goal configuration without sensing, up to symmetries in the push function.

In this section we describe the sensorless 1JOC and its manipulation primitives. These primitives allow us to use Goldberg's algorithm [34] to prove that the sensorless 1JOC possesses the feeding property.

6.1. *The Sensorless* 1*JOC Model*.

The sensorless 1JOC is similar to the 1JOC with the following exceptions. The fence is frictionless with a "stop" at each end, and it is centered above the conveyor (Figure 12). Instead of pivoting about a fixed point, the fence rolls without slipping on a *pivot circle* so that each point on the fence follows an evolute. If this pivot circle has radius r, then when the fence has rotated CW (CCW) by π, it will also have moved up the conveyor by $2r$ and to the left (right) by πr. Moving up the conveyor by $2r$ gives the space necessary for the fence to push a part to an angle π, release it, and then rotate back to 0 degrees and catch the part as it drifts down the conveyor.

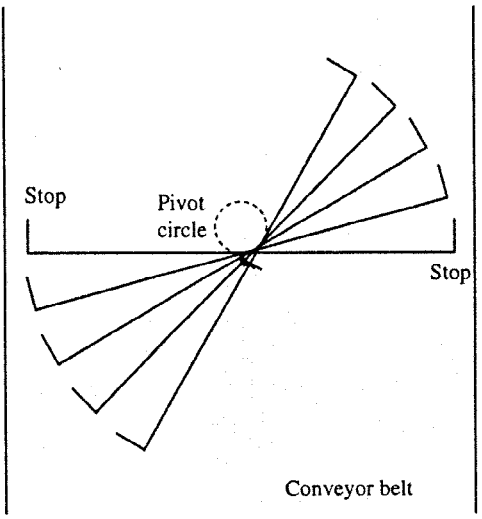


Fig. 12. The sensorless 1JOC.

6.2. *Sensorless* 1*JOC Primitives*. The three primitives we use for sensorless orienting of polygons are *catches*, *tilts*, and *stable pushes* (see Figure 13):

- A *catch* begins with the part either above or on the fence. The fence is held at 0 degrees until the part settles to a stable orientation on the fence.
- A *tilt* begins with the fence horizontal and the part resting on a stable edge. The fence is then tilted CW (CCW) and the part slides without rolling to the right (left) stop.
- A *stable push* begins with the part on a stable edge at the left stop for a CW push, or at the right stop for a CCW push. The fence rotates, pushing the part, so that the part remains fixed relative to the fence. The push is stabilized by the stop. (For the original 1JOC, the push is stabilized by friction.)

A sensorless feeding plan consists of a sequence of catches, tilts, and stable pushes. We describe the behavior of each primitive in detail next.

6.2.1. *Catches*. A catch occurs when a part on the conveyor contacts the fence held stationary at 0 degrees and rotates onto a stable edge. Viewed from a frame fixed in the conveyor, each catch is a *linear normal push*, meaning that the fence is pushing the part in a direction normal to the fence face. To determine the rotation of the part during a catch, we follow Goldberg [34] in using the *radius function* (Figure 14). The radius of a part is the perpendicular distance from a reference point in the part to a supporting line. The radius function $r\colon \mathcal{S}^1 \to \Re$ is a plot of the radius as the supporting line is rotated, and has a period of 2π. When the center of friction is the reference point and the fence is the supporting line, the local minima of the radius function occur at stable edges of the part. A part pushed by a fence rotates to achieve a minimum radius. Each local minimum in the radius function determines a convergent orientation, and each local maximum determines a divergent orientation. Hence a normal push has the net effect of mapping the entire interval between two divergent orientations to the enclosed convergent orientation.

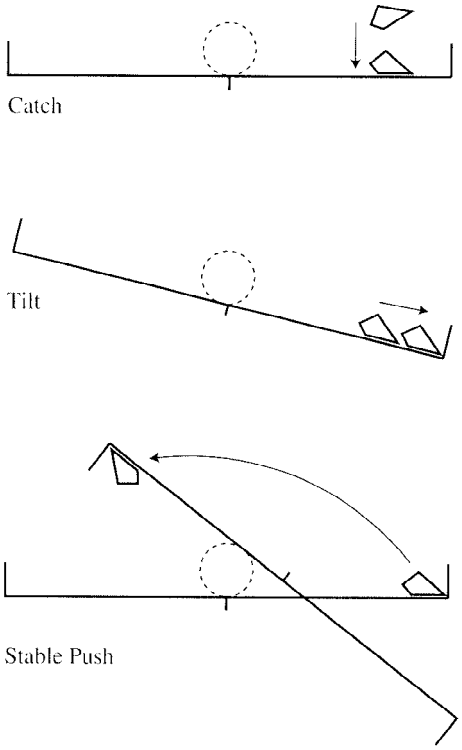


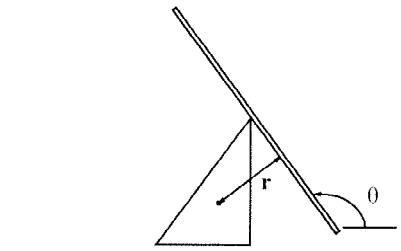
Fig. 13. Primitives used for sensorless orienting.

The *push function* [34], obtained from the radius function, is an alternative representation of the effect of a normal push. It is a mapping $p : \mathcal{S}^1 \rightarrow \mathcal{S}^1$ from the initial relative orientation of a part to the resulting orientation (Figure 15) , and has a period of 2π. We say the part has a symmetric push function if it has a period less than 2π. A compact representation of the push function is shown in Figure 16.

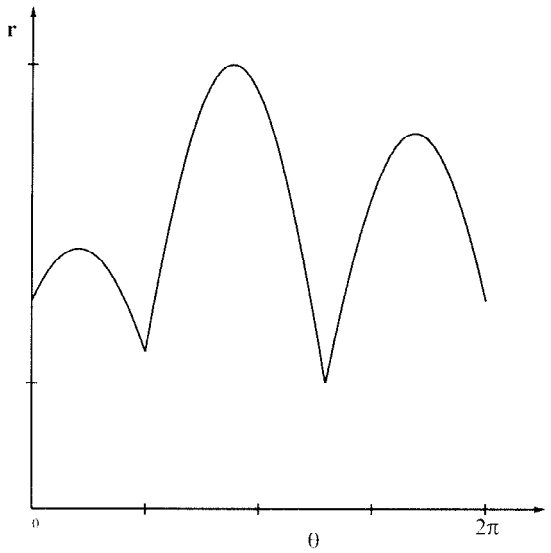
We avoid catches in which the part contacts the fence at a divergent orientation. There are some additional practical issues:

- *Convergence time*. Peshkin and Sanderson's results [56] can be applied to give an upper bound on how long the system must wait for the part to rotate to a stable edge.
- *Metastable edges*. At the end of a catch we perturb the fence so that the polygon does not linger indefinitely on a metastable edge. A very small rotation of the fence, opposite to the desired part rotation, moves the contact normal off of the center of friction, inducing a torque that causes the desired part rotation. In the absence of metastable edges this step is skipped.
- *Stop interference*. As a part converges to a stable edge, it may come into contact with a stop. We assume this does not affect the outcome of a catch.

6.2.2. *Tilts*. A tilt is a CW or CCW rotation of the fence so that a part resting on a stable edge on the fence at 0 degrees slides without rotating to the right or left stop. The



(a)



(b)

Fig. 14. The radius function for a triangle with its center of friction indicated by the solid dot. (a) The radius r of a part at a fence orientation θ is the perpendicular distance from the center of friction to the fence. (b) The radius function is the plot of the part radius as the fence orientation is varied. Based on [34].

sensorless 1JOC uses a tilt before each stable push, and one at the end of the plan to eliminate positional uncertainty of the part.

The part slides faster for steeper tilts, but there is a limit to how steep we can tilt the fence without rotating the part when it contacts the stop. Figure 17 illustrates a simple way of determining the maximum tilt angle.

6.2.3. *Stable Pushes*. When the part is resting against the right (left) fence stop, a stable push allows the fence to rotate the part CCW (CW). The part remains fixed relative to the fence as it moves. To show that these pushes are stable, we must ensure that the contact forces provided by the fence and the stop can balance the support friction force for any pressure distribution of the part consistent with the center of friction. In this section we

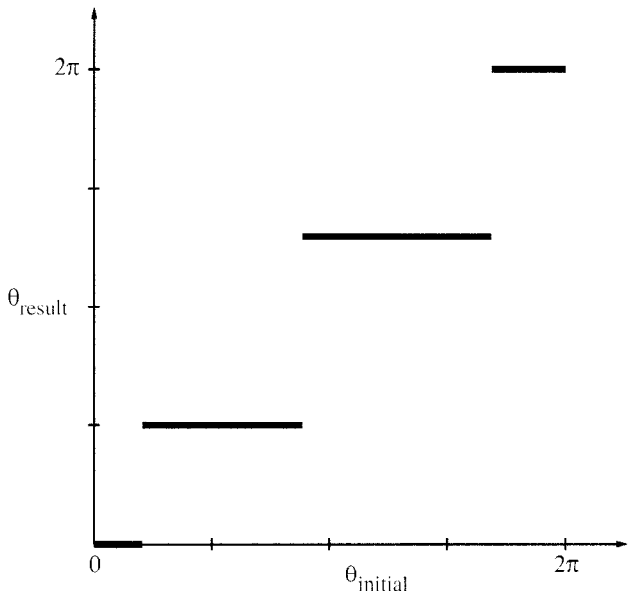


Fig. 15. The push function for the triangle. The horizontal and vertical axes are the initial and resulting relative orientations of the part. Based on [34].

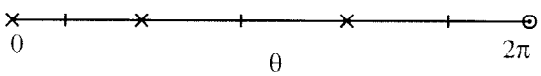


Fig. 16. A compact representation of the push function for the triangle. The $\times$'s and the vertical bars indicate the convergent and divergent fence orientations respectively, and the circle indicates wraparound at 2π.

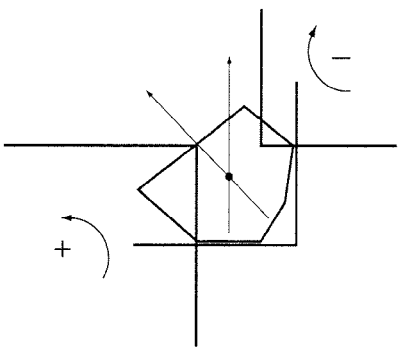


Fig. 17. The maximum tilt angle can be derived using simple force–torque balance. Here the constraints are represented using moment-labeling regions [48]. The convex combination of the contact forces from the fence and stop consists of the set of all forces that yield positive moment about the region labeled $+$ and negative moment about the region labeled $-$. Two forces are shown—the force required for force balance for a tilt angle of zero, and a force which just begins to violate the constraint, defining the maximum tilt angle. A nonzero maximum always exists, because the polygon is on a stable edge.

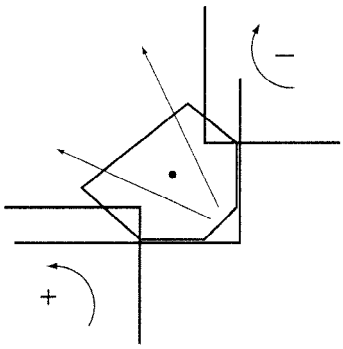


Fig. 18. Frictionless contact forces from the fence and stop can be represented as moment-labeling regions as in Figure 17.

show that it is always possible to choose a pivot circle radius r, fence length, and ratio of the conveyor velocity v to the fence's pushing angular velocity ω to make the pushes stable.

Assume the part is resting against the right stop. The set of possible contact forces is given by the convex combination of the frictionless forces at each contact (Figure 18). For a set of contact forces, we can find a set of stable pushing motions using the algorithm described in [41]. (This algorithm is more general than the procedure STABLE, which applies to a single line contact.) Figure 19 shows the CCW stable pushing motions $\mathcal{V}_{\text{stable}}$ for the example of Figure 18. The stable pushing motions are represented as rotation centers fixed in the part frame $\mathcal{F}_p$, relative to the conveyor it is sliding on. For these rotation centers, the contact forces can balance the support friction force for any pressure distribution of the part.[5]

Now we assume that the conveyor velocity v is negligible. Then during a stable push, the rotation center in the part frame $\mathcal{F}_p$ is the point on the fence in contact with the pivot circle. Defining the zero location on the fence to be the point in contact with the circle when the fence is held at 0 degrees, the rotation centers on the fence sweep the range $-r\pi/2$ to $r\pi$, where r is the radius of the pivot circle, as the fence rotates from $-\pi/2$ to π. (Note that $-\pi/2$ is the steepest possible tilt angle.) We can see in the example of Figure 19 that the push is stable for all angles of the fence.

We can always choose the fence to be long enough that the push is stable for all fence angles. When the rotation center is an infinite distance away along the fence, the required contact force $\mathbf{f}$ is normal to the fence and through the center of friction. As the rotation center moves in along the fence, the required contact force smoothly rotates CCW because the part's support friction is "above" the fence. This CCW force is provided by the stop (see Figure 18). More precisely, we can define the largest ball $\mathcal{B}$ of forces about $\mathbf{f}$ such that all forces in $\mathcal{B}$ and CCW of $\mathbf{f}$ can be provided by the contacts. Because $\mathcal{B}$ always has nonzero radius, the distance to the nearest stable rotation center

[5] Technically, to prove the motion is stable, we must also prove that no other motions of the part are possible. Unlike for the procedure STABLE, there is no general proof of this. Here we simply assume it to be the case.

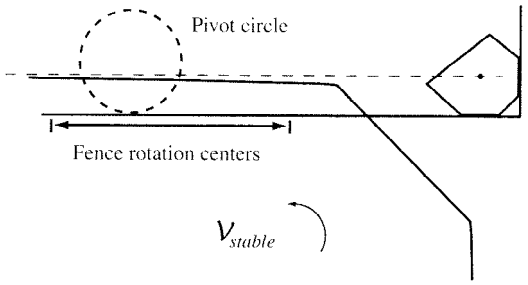


Fig. 19. The set of CCW stable pushing motions $\mathcal{V}_{\text{stable}}$ represented as rotation centers. $\mathcal{V}_{\text{stable}}$ is always bounded from above by a line parallel to the stable edge and through the center of friction. Also shown is an example pivot circle and the points on the fence that roll on the pivot circle as it rotates from $-\pi/2$ to π.

is finite, implying that we can always choose a sufficiently long fence to make pushing stable if the conveyor velocity v is negligible.

If the conveyor velocity v is not negligible with respect to the pushing angular velocity ω, the rotation center of the part relative to the conveyor during stable pushing can be written as a function of the ratio v/ω. For a fence angle θ, the rotation center location is (x, y), $x = \theta r + (v/\omega)\cos(\pi - \theta)$, $y = (v/\omega)\sin(\pi - \theta)$, where the x axis is aligned with the fence and the origin is the point on the fence in contact with the pivot circle at $\theta = 0$. The rotation center traces out a curve in the part frame $\mathcal{F}_p$ (Figure 20).

For a push to be stable, the curve must stay inside $\mathcal{V}_{\text{stable}}$. For a given pivot circle, fence length, and pushing velocity ω (perhaps limited by the quasistatic assumption), we would like to find the highest possible conveyor velocity v. To do this, we increase v until the curve no longer remains in $\mathcal{V}_{\text{stable}}$ (Figure 21). By doing this for each possible configuration of a given part, we can find a maximum conveyor velocity guaranteed to give stable pushes for all configurations. Maximizing the conveyor speed maximizes the feeder's throughput.

Because $\mathcal{V}_{\text{stable}}$ is bounded from above by a line parallel to the stable edge and through the center of friction (Figure 19), one condition for the rotation center curve to be contained in $\mathcal{V}_{\text{stable}}$ is that $v < \omega d$, where d is the distance from the center of friction

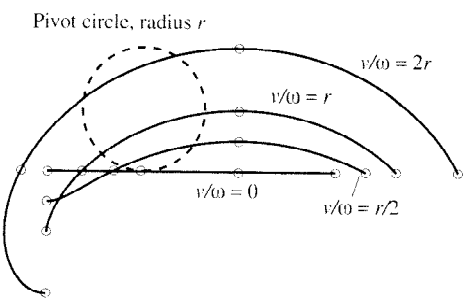


Fig. 20. Curves of the rotation center of the part relative to the conveyor during stable pushing. The curves shown are for several different ratios of v/ω. Each curve is drawn by varying the fence angle θ in the range $[-\pi/2, \pi]$. (If the stable pushes in a sensorless plan use a smaller angle range, the curves can be truncated.) Circles are drawn around the rotation centers on each curve at fence angles $-\pi/2, 0, \pi/2, \pi$.

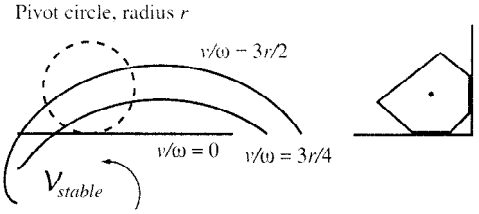


Fig. 21. Rotation center curves for different ratios v/ω. For this example, the maximum conveyor velocity v that guarantees stable pushes is $3r\omega/4$. For larger values, the curve leaves $\mathcal{V}_{\text{stable}}$.

to the stable edge. When d is small, the conveyor velocity v must be small relative to the fence angular velocity ω. By moving the pivot circle "down" relative to the fence, however, we can move the rotation center curves further into $\mathcal{V}_{\text{stable}}$, allowing a higher conveyor velocity v. There are other possible variations, but we do not consider them further.

6.3. *The Feeding Property.* We can now demonstrate the feeding property for the sensorless 1JOC.

THEOREM 3. *The sensorless* 1*JOC possesses the feeding property*:

- *for all polygonal parts,*
- *for the three-dimensional space $\mathcal{I}$ of initial configurations such that the part initially comes to rest on a stable edge, completely on the conveyor, against the fence fixed at* 0 *degrees without touching either of the fence stops, and*
- *using only catches, tilts, stable pushes, and conveyor drift.*

PROOF. Given a part in stable edge contact with the fence, we can execute a tilt to bring the part to either stop. The fence can then rotate the part by any angle in the range $(-\pi, \pi)$ using a stable push. Each stable push is followed by a catch when the part drifts into contact with the fence. Goldberg [34] developed an algorithm that finds the shortest sequence of parallel-jaw grasps to orient any polygonal part up to symmetry. We can use this algorithm to find the shortest sequence of fence rotations guaranteed to bring any polygonal part from any initial stable edge to some goal stable edge, up to symmetry in its push function. The goal position must be at one of the stops; a final tilt will be required to bring the part to the goal stop. □

To ensure that the sensorless 1JOC can orient all parts up to symmetry in the push function, the fence must be capable of stable pushing anywhere in the range $(-\pi, \pi)$ before releasing and catching. To see that this is necessary, consider the push function of Figure 22. There are two stable edges at angles 0 and $\pi - \varepsilon$, where ε is a small positive value, so the push function is not symmetric. The angle range from both stable angles to the CCW limits of their convergence regions is δ, where δ is a small positive value.

Because the distances to both CCW limits are equal, we must distinguish between the stable edges by using the different distances to their CW limits. The fence must

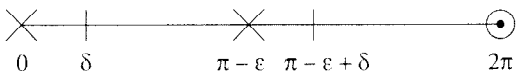


Fig. 22. This push function requires the fence to push the part at least $\pi - \delta - \varepsilon$ radians before releasing it and catching it. This allows the fence to change its orientation relative to the part by an angle $-\pi + \delta + \varepsilon$, the minimum required to distinguish between the two stable edges.

move CW relative to the part by a magnitude of at least $\pi - \delta - \varepsilon$. This implies a CCW stable push (before the release and catch) of the same magnitude. As $\delta \to 0$, $\varepsilon \to 0$, the required push angle approaches π.

Figure 23 constructs a part with the push function of Figure 22, requiring a stable push of up to π to orient it. The reflection of this part requires a stable push of up to $-\pi$, indicating that the sensorless 1JOC must be capable of stable pushes in the range $(-\pi, \pi)$ to orient all parts up to symmetry in the push function.

Most parts can be oriented with an angle range much less than $(-\pi, \pi)$, and we are interested in characterizing such parts. For example, it is easy to show that all triangles are orientable using the angle range $(-\pi/2, \pi/2)$. There exist quadrilaterals, however, that cannot be oriented up to symmetry using this angle range. Goldberg and Overmars [35] showed that $(-\pi/2, \pi/2)$ is sufficient if parts possess no "partial" symmetries—the distances from each stable angle to each edge of its convergence range are unique.

7. The Sensorless 1JOC Planner. The sensorless 1JOC can use the primitives described above to position and orient parts without sensing. A sensorless orienting plan begins with a part in stable edge contact with the fence. A sequence of stages, each consisting of a tilt, a stable push, and a catch, brings the part to a known orientation. A final tilt brings the part to one of the fence stops. Orienting a part with n stable edges

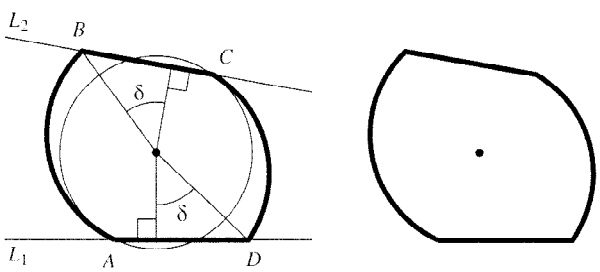


Fig. 23. Constructing a part with the push function of Figure 22. Start with a circle centered at the center of friction of the part. Cut the bottom of the circle with the horizontal line L_1, and cut the top of the circle with the line L_2, which is at an angle $-\varepsilon$ to the horizontal. The intersections of these lines with the circle define the points A and C as shown. Draw the normals to L_1 and L_2 which pass through the center of friction. Pivot these normals an angle δ about the center of friction and intersect them with the lines L_1 and L_2, giving the points B and D as shown. Now construct a curve from A to B which spirals away from the center of friction. Construct a similar curve between C and D. The part is defined by the curve from A to B, the stable edge from B to C, the curve from C to D, and the stable edge from D to A. By making the cuts, ε, and δ arbitrarily small and replacing the curves with an arbitrarily large number of line segments (none of these edges is stable), we create a polygonal part that requires a stable push of up to π to orient it.

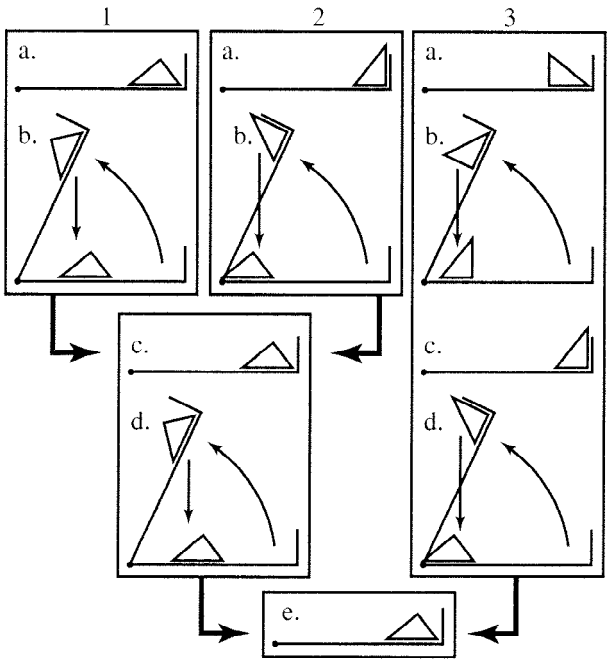


Fig. 24. Example sensorless feeding plan. Three different initial orientations converge to a single final orientation. The tilts are not shown, and only the right half of the sensorless 1JOC is shown in this plan.

takes $O(n)$ stages, as can be shown using Goldberg's algorithm [34]. Since any polygon shares its push function with an infinite set of polygons [3], a single plan can orient multiple nonsimilar parts that share the same push function.

We use breadth-first search in the space of representative actions to find the shortest sequence of fence rotations to orient the part. The representative actions are obtained from the push function and provide a finite discretization of the action space that covers it. See Figure 24 for an example plan.

The sequence of fence rotations specifies the sequence of stable pushes, each of which is followed by a catch. Stable pushes in the interval $(-\pi, \pi)$ are sufficient to rotate the parts as desired. We have to precede each stable push with an appropriate tilt. Prior to a CCW stable push in the interval $(0, \pi)$, we tilt the fence CW to bring the part to the right stop. Prior to a CW stable push in the interval $(-\pi, 0)$, we tilt the fence CCW to bring the part to the left stop. Once the tilt direction is determined, the tilt angle is computed for the set of possible part orientations for the stage.

8. Sensorless 1JOC Implementation. Our implementation of the sensorless 1JOC uses one joint of an Adept 550 robot to rotate the fence about a fixed pivot point since most parts do not require rotations close to π. We used a plexiglass fence and delrin parts to test the plans. We generated plans to test the right triangle and 8-gon shown in Figure 25. We used a conveyor velocity of 20 mm/s and fence angular velocities of

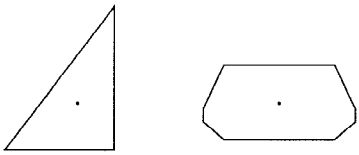


Fig. 25. The two test parts used in the experiments. The plan to feed the triangle is illustrated in Figure 24.

30 degrees/s for stable pushes and 6 degrees/s for tilts. The tilt angles were 45 degrees. The times for catches and tilts to proceed to completion were found empirically. Since we observed that stable push actions selected from smaller action ranges are less robust than those selected from larger action ranges, we specify a minimum action range size of 5 degrees for valid actions.

We ran 20 trials for the triangle starting in different configurations along the fence. All 20 trials were successful. We ran 20 trials on the 8-gon for different start configurations, of which 16 were successful. The failures occurred during tilts when the polygon rotated away from the stable edge it was sliding on, probably due to the nonzero friction coefficient of the fence. From our tests on this and other parts, we observed that edges that are nearly unstable sometimes cause undesired part rotations, particularly when a part slides on such an edge during a tilt. Such edges also sometimes require small tilt angles, which may not allow sliding to proceed to completion. Another occasional failure mode occurs when a part contacts the fence stop in the middle of a tilt, leading to undesired part rotation as the tilt progresses.

9. Variations on a Theme. The 1JOC approach uses a fixed velocity conveyor in combination with a single servoed joint to obtain the diversity of motions required for planar manipulation. We have shown by proof and demonstration that the 1JOC is capable of useful planar manipulation: any polygon is controllable from a broad range of initial configurations to any goal chosen from a broad range of goal configurations. We have also shown that a variant, the sensorless 1JOC, can position and orient polygons without sensing.

For this paper we designed the system and the set of actions to simplify analysis and planning. There are many variations on the approach which may be more suitable in different contexts. Some variations on the system configuration are: a curved fence; a prismatic fence in place of the revolute fence; a rotary table in place of the conveyor; or use of gravity rather than a conveyor. Some variations on the actions are: optimization of fence rotation rates instead of using fixed values; complete revolutions of the part to reduce jogs; and speeds high enough to require dynamic analysis instead of quasistatic. Another variation for three-dimensional parts would be to use techniques similar to Bicchi and Sorrentino's [14] to roll parts on the conveyor.

When addressing potential industrial applications, it is important to consider the whole system, including interactions with surrounding equipment. Several different scenarios have occurred to us: a 1JOC to pose parts followed by a simple pick-and-place device; a 1JOC to singulate parts for a SCARA; or two or three 1JOCs pipelined to singulate and feed parts.

Acknowledgments. Many thanks to Costa Nikou and Evan Shechter for implementations of the 1JOC on the Adept robot. We thank Mike Erdmann, Garth Zeglin, and Nina Zumel for early discussions about this paper. We thank Adept, NSK, and SONY for helping us learn about automation and for making equipment available.

References

[1] S. Akella, W. Huang, K. M. Lynch, and M. T. Mason. Planar manipulation on a conveyor with a one joint robot. In *Proceedings of the International Symposium on Robotics Research*, pages 265–276. Springer-Verlag, New York, Oct. 1995.

[2] S. Akella, W. Huang, K. M. Lynch, and M. T. Mason. Sensorless parts feeding with a one joint robot. In *Proceedings of the Workshop on the Algorithmic Foundations of Robotics*, pages 229–238. A. K. Peters, Boston, MA, July 1996.

[3] S. Akella and M. T. Mason. Parts orienting with partial sensor information. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Volume 1, pages 557–564, Leuven, Belgium, May 1998.

[4] S. Akella and M. T. Mason. Posing polygonal objects in the plane by pushing. *The International Journal of Robotics Research*, 17(1):70–88, Jan. 1998.

[5] J. C. Alexander and J. H. Maddocks. Bounds on the friction-dominated motion of a pushed object. *International Journal of Robotics Research*, 12(3):231–248, 1993.

[6] H. Arai. Controllability of a 3-dof manipulator with a passive joint under a nonholonomic constraint. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3707–3713, 1996.

[7] H. Arai and O. Khatib. Experiments with dynamic skills. In *Proceedings of the* 1994 *Japan–USA Symposium on Flexible Automation*, pages 81–84, 1994.

[8] H. Arai and S. Tachi. Position control of a manipulator with passive joints using dynamic coupling. *IEEE Transactions on Robotics and Automation*, 7(4):528–534, Aug. 1991.

[9] H. Arai, K. Tanie, and S. Tachi. Dynamic control of a manipulator with passive joints in operational space. *IEEE Transactions on Robotics and Automation*, 9(1):85–93, Feb. 1993.

[10] Z. Balorda. Reducing uncertainty of objects by robot pushing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1051–1056, Cincinnati, OH, 1990.

[11] M. S. Bazaraa and C. M. Shetty. *Nonlinear Programming*: *Theory and Algorithms*. Wiley, New York, 1979.

[12] M. Bergerman, C. Lee, and Y. Xu. Experimental study of an underactuated manipulator. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Volume 2, pages 317–322, 1995.

[13] D. Berkowitz and J. Canny. Designing parts feeders using dynamic simulation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1127–1132, 1996.

[14] A. Bicchi and R. Sorrentino. Dexterous manipulation through rolling. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 452–457, 1995.

[15] K. Böhringer, R. Brown, B. Donald, J. Jennings, and D. Rus. Distributed robot manipulation: experiments in minimalism. In *Proceedings of the International Symposium on Experimental Robotics*, 1995.

[16] K. F. Böhringer, V. Bhatt, and K. Y. Goldberg. Sensorless manipulation using transverse vibrations of a plate. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1989–1996, 1995.

[17] G. Boothroyd. *Assembly Automation and Product Design*. Marcel Dekker, New York, 1992.

[18] G. Boothroyd, C. Poli, and L. E. Murch. *Automatic Assembly*. Marcel Dekker, New York, 1982.

[19] D. L. Brock. Enhancing the dexterity of a robot hand using controlled slip. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 249–251, 1988.

[20] R. W. Brockett. Nonlinear systems and differential geometry. *Proceedings of the IEEE*, 64(1), Jan. 1976.

[21] M. Brokowski, M. Peshkin, and K. Goldberg. Curved fences for part alignment. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Volume 3, pages 467–473, Atlanta, GA, 1993.

[22] R. C. Brost. Automatic grasp planning in the presence of uncertainty. *International Journal of Robotics Research*, 7(1):3–17, Feb. 1988.
[23] R. C. Brost. Dynamic analysis of planar manipulation tasks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2247–2254, 1992.
[24] M. Bühler and D. E. Koditschek. From stable to chaotic juggling: theory, simulation, and experiments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1976–1981, Cincinnati, OH, 1990.
[25] M. Caine. The design of shape interactions using motion constraints. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 366–371, 1994.
[26] J. F. Canny and K. Y. Goldberg. "RISC" industrial robotics: recent results and open problems. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1951–1958, 1994.
[27] A. D. Christiansen. Manipulation planning from empirical backprojections. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 762–768, Sacramento, CA, 1990.
[28] A. D. Christiansen, A. D. Edwards, and C. A. Coello Coello. Automated design of parts feeders using a genetic algorithm. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 846–851, 1996.
[29] P. E. Crouch. Spacecraft attitude and stabilization: applications of geometric control theory to rigid body models. *IEEE Transactions on Automatic Control*, 29(4):321–331, Apr. 1984.
[30] B. R. Donald, J. Jennings, and D. Rus. Information invariants for cooperating autonomous mobile robots, In *Proceedings of the International Symposium on Robotics Research*, Hidden Valley, PA, 1993. MIT Press, Cambridge, MA.
[31] M. A. Erdmann. An exploration of nonprehensile two-palm manipulation: planning and execution. In *Proceedings of the International Symposium on Robotics Research*, 1995.
[32] M. A. Erdmann. Understanding action and sensing by designing action-based sensors. *International Journal of Robotics Research*, 14(5):483–509, Oct. 1995.
[33] M. A. Erdmann and M. T. Mason. An exploration of sensorless manipulation. *IEEE Transactions on Robotics and Automation*, 4(4):369–379, Aug. 1988.
[34] K. Y. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10:201–225, 1993.
[35] K. Y. Goldberg and M. Overmars. Personal communication, 1996.
[36] S. Goyal, A. Ruina, and J. Papadopoulos. Planar sliding with dry friction. Part 1. Limit surface and moment function. *Wear*, 143:307–300, 1991.
[37] S. Goyal, A. Ruina, and J. Papadopoulos. Planar sliding with dry friction. Part 2. Dynamics of motion. *Wear*, 143:331–352, 1991.
[38] J. Krishnasamy, M. J. Jakiela, and D. E. Whitney. Mechanics of vibration-assisted entrapment with application to design. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 838–845, 1996.
[39] J.-C. Latombe. *Robot Motion Planning*. Kluwer, Dordrecht, 1991.
[40] J. Liebman, L. Lasdon, L. Schrage, and A. Waren. *Modeling and Optimization with GINO*. The Scientific Press, South San Francisco, CA, 1986.
[41] K. M. Lynch. The mechanics of fine manipulation by pushing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 112–119, Nagoya, Japan, 1995.
[42] K. M. Lynch and M. T. Mason. Controllability of pushing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 112–119, Nagoya, Japan, 1995.
[43] K. M. Lynch and M. T. Mason. Stable pushing: Mechanics, controllability, and planning. In *Algorithmic Foundations of Robotics*, pages 239–262. A. K. Peters, Boston, MA, 1995.
[44] K. M. Lynch and M. T. Mason. Dynamic underactuated nonprehensile manipulation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1996.
[45] M. Mani and W. Wilson. A programmable orienting system for flat parts. In *Proceedings of the North American Manufacturing Research Institute Conference XIII*, 1985.
[46] V. Manikonda and P. S. Krishnaprasad. Dynamics and controllability of a planar rigid body with a thruster. In *Proceedings of the* 30*th Annual Conference on Information Sciences and Systems*, 1996.
[47] M. T. Mason. Mechanics and planning of manipulator pushing operations. *International Journal of Robotics Resaerch*, 5(3):53–71, Fall 1986.

[48] M. T. Mason. Two graphical methods for planar contact problems. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 443–448, Osaka, Japan, Nov. 1991.

[49] M. T. Mason and R. C. Brost. Automatic grasp planning: an operation space approach. In *Proceedings of the Sixth Symposium on Theory and Practice of Robots and Manipulators*, pages 321–328, Cracow, Poland, Sept. 1986. Alma Press.

[50] M. T. Mason and J. K. Salisbury, Jr. *Robot Hands and the Mechanics of Manipulation*. The MIT Press, Cambridge, MA, 1985.

[51] T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9(2):62–82, 1990.

[52] T. McGeer. Passive walking with knees. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1640–1645, 1990.

[53] H. Nijmeijer and A. J. van der Schaft. *Nonlinear Dynamical Control Systems*. Springer-Verlag, Berlin, 1990.

[54] G. Oriolo and Y. Nakamura. Control of mechanical systems with second-order nonholonomic constraints: Underactuated manipulators. In *Proceedings of the Conference on Decision and Control*, pages 2398–2403, 1991.

[55] M. A. Peshkin and A. C. Sanderson. The motion of a pushed, sliding workpiece. *IEEE Journal of Robotics and Automation*, 4(6):569–598, Dec. 1988.

[56] M. A. Peshkin and A. C. Sanderson. Planning robotic manipulation strategies for workpices that slide. *IEEE Journal of Robotics and Automation*, 4(5):524–531, Oct. 1988.

[57] A. Rao, D. Kriegman, and K. Y. Goldberg. Complete algorithms for reorienting polyhedral parts using a pivotal gripper. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2242–2248, 1995.

[58] A. A. Rizzi and D. E. Koditschek. Further progress in robot juggling: the spatial two-juggle. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Volume 3, pages 919–924, Atlanta, GA, 1993.

[59] M. Shirai and A. Saito. Parts supply in SONY's general-purpose assembly system SMART. *Japanese Journal of Advanced Automation Techniques*, 1:108–111, 1989.

[60] O. J. Sørdalen, Y. Nakamura, and W. J. Chung. Design of a nonholonomic manipulator. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 8–13, 1994.

[61] H. J. Sussmann. Lie brackets, real analyticity and geometric control. In R. W. Brockett, R. S. Millman, and H. J. Sussmann, editors, *Differential Geometric Control Theory*. Birkhäuser, Boston, MA, 1983.

[62] P. J. Swanson, R. R. Burridge, and D. E. Koditschek. Global asymptotic stability of a passive juggler: a parts feeding strategy. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1983–1988, 1995.

[63] J. C. Trinkle, R. C. Ram, A. O. Farahat, and P. F. Stiller. Dexterous manipulation planning and execution of an enveloped slippery workpiece. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 442–448, 1993.

[64] A. F. van der Stappen, K. Y. Goldberg, and M. Overmars. Geometric eccentricity and the complexity of manipulation plans. *Algorithmica*, this issue, pp. 494–514.

[65] J. Wiegley, K. Goldberg, M. Peshkin, and M. Brokowski. A complete algorithm for designing passive fences to orient parts. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1133–1139, 1996.

[66] N. B. Zumel and M. A. Erdmann. Nonprehensile two palm manipulation with non-equilibrium transitions between stable states. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3317–3323, 1996.