Orienting Micro-Scale Parts with Squeeze and Roll Primitives

Mark Moll† mmoll@cs.cmu.edu
Ken Goldberg‡ goldberg@ieor.berkeley.edu
Michael A. Erdmann† me@cs.cmu.edu
Ron Fearing‡ ronf@eecs.berkeley.edu

† Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213
‡ IEOR and EECS Departments, UC Berkeley, Berkeley, CA 94720

Abstract— **Orienting parts that measure only a few micrometers in diameter introduces several challenges that need not be considered at the macro-scale. First, there are several kinds of sticking effects due to Van der Waals forces and static electricity which complicate hand-off motions and release of a part. Second, the degrees of freedom of micro-manipulators are limited. This paper proposes a pair of manipulation primitives and a complete algorithm that addresses these challenges. We will show that a sequence of these two manipulation primitives can uniquely orient any asymmetric part while maintaining contact without sensing. This allows us to apply the same plan to many (identical) parts simultaneously. For asymmetric parts we can find a plan of length $O(n)$ in $O(n)$ time that orients the part, where n is the number of vertices.**

Keywords— **manipulation, microassembly, parts orienting, parts feeding**

I. Introduction

Increased miniaturization of mass-produced consumer and industrial products such as disk drives, cameras, displays, and sensors will require fundamental innovations in parts handling. Conventional "pick-and-place" techniques do not work well at the micro-scale where sticking affects dominate. We will use the term 'sticking effects' to describe the combined effect of Van der Waals forces, electrostatic surface charges and other attractive forces that occur at the micro-level. Due to these sticking effects parts can stick to a manipulator without being grasped. To manipulate micro-parts, we propose a manipulation strategy that consists of applying simple operations requiring no more than two degrees of freedom. These operations are executed by a parallel jaw gripper. During an operation one degree of freedom will be active and the other will be compliant in order to maintain contact with the part. The gripper maintains complete control over the part's orientation. Eventually the gripper will have to release the part. This can be done, for instance, by chemically bonding the part to the assembly.

Figure 1 illustrates a plan that orients a part by rolling and squeezing it between two horizontal micro-scale jaws. Note that for this type of manipulation we only need to consider the convex hull of the part. Initially the part can be in any orientation, but after execution of the plan the part will be in one of two orientations. The state transitions after each operation are indicated by the arrows.

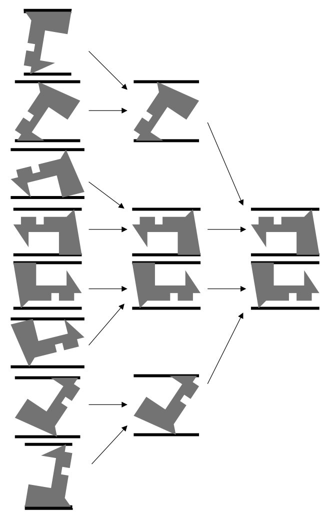


Fig. 1. Example of a squeeze-roll plan orienting a polygonal part. Initial orientations are shown on the left, final orientations on the right. Initially, the part is in one of six orientations. After two sensorless operations the part is oriented up to 180° symmetry.

This work was supported in part by the National Science Foundation under IIS-9820180, CDA-9726389 and DMI-0010069. Research funding was also provided by Adept Technology, and California State MICRO Grant 00-032.

II. Related Work

A. Micromanipulation

For a more complete overview of micromanipulation see, e.g., [11]. When parts are smaller than one millimeter the effect of adhesive forces becomes significant. Fearing [17] gives an overview of all the different adhesive forces that occur at this scale. Arai et al. [3] describe how these forces can be minimized.

Many researchers have worked on reliable pick and place operation of micro-scale objects. Miyazaki and Sato [28] and Saito et al. [34] describe a system where a needle is used to assemble 3D structures composed of particles ranging from 10nm to 1μm in size. The sticking effects enable the needle to pick up a particle. By translating along a so-called shearing trajectory the needle releases the particle. A human operator controls the manipulator, while a scanning electron microscope (SEM) provides the visual feedback. Koyano and Sato [21] describe a similar system for pick-and-place operation. They use two different sized needles and define different part release motions. One motion consists of clamping the part onto a surface with a thin needle, while the large needle releases the part. The sticking effects are not large enough to make the part stick to the thin needle. Another part release motion consist of rotating the needle around the part to minimize the contact area.

One problem with a human-controlled micromanipulation system is that the microscope only gives a two-dimensional slice of a three-dimensional object. Arai et al. [4] address this problem by letting the human operator manipulate the part in a 3D virtual reality environment.

Zesch and Fearing [40] explore how one can orient parts by pushing them with an AFM cantilever equipped with a force sensor. The force sensor is used to detect obstacles and changing contact conditions. In [35] a system is described consisting of two orthogonal one degree-of-freedom tweezers. The tweezers are equipped with force sensors. Shimada et al. describe two different ways to orient the part with these tweezers: one way is to roll the part between them, another is to pivot the part around a fixture. A similar approach is taken by [23], who analyze the finger forces required to manipulate a planar micro-scale part with two circular fingers. To plan a path from a start to a goal configuration Maeda et al. construct a graph where the nodes are configurations, and the edges correspond to push, tumble and regrasp operations.

Our work differs from most other research on micromanipulation in that no human controller is needed. To the best of the authors' knowledge this paper describes the first algorithmic complexity results for orienting polygonal parts at the micro-scale. In the following section we give an overview of parts orienting research. Almost all research in this area focuses on macro-scale parts and does not consider sticking effects.

B. Parts Orienting

The problem of how to bring parts into a desired orientation has been well studied at the macro-scale. It is not necessary to grasp an object to orient it. Mason [26, 27] showed how to orient parts by pushing them. This can be used to design a sequence of fences over a conveyor belt [5, 31, 39]. Akella et al. [1] showed that instead of a sequence of fences one can also use one fence with one rotational degree of freedom.

Erdmann and Mason [15] developed a tray-tilting sensorless manipulator that can orient planar parts in the presence of friction. If it is not possible to bring a part into a unique orientation, the planner would try to minimize the number of final orientations. In [16] it is shown how (with some simplifying assumptions) three-dimensional parts can be oriented using a tray-tilting manipulator. In particular, for polyhedral parts with n faces a sequence of 'tilts' of length $O(n)$ can be found in $O(n^3)$ time. Zumel [41] used a variation of the tray tilting idea to orient planar parts. Zumel used two actuated arms connected at a hinge to tilt parts from one arm to the other.

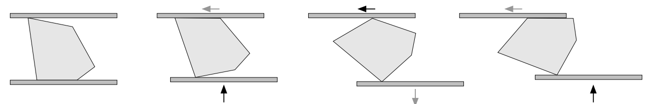


Fig. 3. A sequence of micro-manipulation primitives, from left to right. We automatically follow each roll by a squeeze. The small arrows indicate in which direction the jaws have moved during each operation. The black arrows indicate a controlled motion, the gray arrows indicate a compliant motion. Note that parts always remain in contact with both gripper jaws.

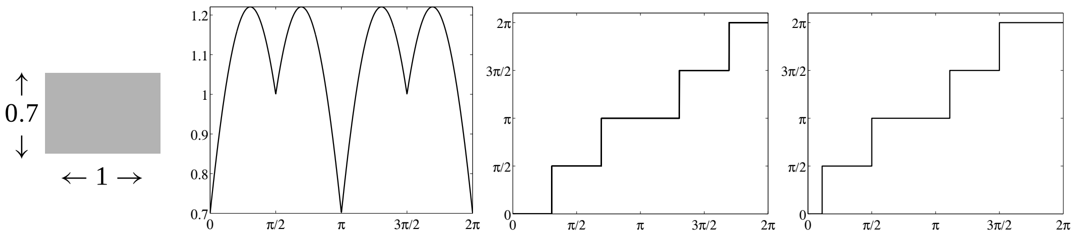
Goldberg [18] showed that it is possible to orient any polygonal part with a frictionless parallel-jaw gripper without sensors. Goldberg proved that for every n-sided polygonal part, a sequence of 'squeezes' can be computed in $O(n^2 \log n)$ time that will orient it up to symmetry. The length of such a sequence is bounded by $O(n^2)$. Chen and Ierardi [13] improved this bound to $O(n)$ and showed that the algorithm runs in $O(n^2)$. Van der Stappen et al. [37] extended the results to non-polygonal parts and showed that for many parts only a constant number of operations are required to orient them. Akella and Mason [2] showed that with partial sensor information the length of this sequence can be reduced to $O(m)$, where m is the maximum number of states with the same sensor value.

Bicchi and colleagues showed that by rolling an object between the two hands of a parallel-jaw gripper it is possible to orient and position polyhedral parts [12, 25] and smooth 3D parts [24]. The jaws are equipped with tactile sensors, which allows the system to reconstruct the shape of *unknown* smooth objects as well [6].

In [32] an algorithm is described to orient polyhedral parts using so-called pivot grasps. A part is grasped with two hard finger contacts and is then free to rotate around the axis formed by the contacts.

Another way to orient parts is to design a manipulator shape specifically for a given part. This approach was first considered for the Sony APOS system [19]. The design was done mainly by ad-hoc trial and error. Later, Moll and Erdmann [29] explored a way to automate this process.

In recent years a lot of work has been done on programmable force fields to orient parts [7, 10, 20, 33] The idea is that an abstract force field (implemented using e.g. MEMS actuator arrays) can be used to push the part into a certain orientation. Böhringer et al. used Goldberg's algorithm [1993] to define a sequence of 'squeeze fields' to orient a part. They also gave an example how programmable vector fields can be used to simultaneously sort different parts and orient them. Kavraki [20] presented a vector field that induced two stable configurations for most parts. In 2000, Böhringer et al. proved a long-standing conjecture that the vector field proposed in [9] is a universal feeder/orienter device, i.e., it induces a *unique* stable configuration for most parts. Recently, Sudsang and Kavraki [36] introduced another vector field that has that property.



(a) A rectangular part (b) Its diameter function (c) Its squeeze function (d) Its roll function with upper jaw translating 1 unit to the left

Fig. 2. The diameter, squeeze and roll functions. Note that the squeeze and roll functions are monotone.

III. Two Micro-Manipulation Primitives

We are trying to solve the problem of orienting micro-scale parts without sensing using limited degree-of-freedom manipulators. In this section we will state our assumptions and introduce the basic manipulation primitives. In the next section we will explain how the primitives can be composed to solve the parts orienting problem.

We manipulate parts with a pair of parallel jaw grippers. We assume there is no slip between the jaws and the part. Although this may not be true in general, we can for a given part compute a lower bound on the coefficient of friction such that no slip will occur. We also assume both jaws will always be in contact with the part. We can realize this to some extent mechanically by a remote center of compliance for each jaw [38]. The part will also tend to stick to the jaws due to adhesive forces. The exact magnitude of those forces is hard to predict, but if we cover the jaws with a uniform adhesive material such as GelPak (http://www.gelpak.com), the adhesion from this material would dominate the uncertain micro-scale forces. Finally, we assume that there are no sudden changes in the pose of the part due to sticking effects. For the operations described below these qualitative assumptions are sufficient, i.e., these operations do not rely on perfect knowledge of the sticking effects, but have the same result for a range of sticking effects. The *upper* jaw can translate in the horizontal direction, the *lower* jaw can translate in the vertical direction. With each pair of grippers we can perform the following two operations:

Squeeze We close the jaws by moving the lower jaw closer to the upper jaw and, simultaneously, allow the upper jaw to move compliantly until a stable grasp is reached. This is equivalent to a frictionless jaw grasp [18].

Roll We translate the upper jaw in the horizontal direction by a given amount and allow the lower jaw to move compliantly. To make sure that the part is always in one of a finite number of orientations, we automatically follow each roll by a squeeze.

These two operations are illustrated in figure 3. It follows that the operations on the part are deterministic, so that after each manipulation step the uncertainty regarding the part's orientation is either reduced or stays the same. These operations can be defined more formally as functions that map orientations to orientations. Let S^1 be the set of orientations in the plane. Consider the *diameter function* $d : S^1 \to \mathbb{R}$, which, given a part orientation, returns the distance between the jaws when they just touch the part in this orientation. We define the *squeeze function*, $s : S^1 \to S^1$, such that if θ is the initial part orientation, $s(\theta)$ is the orientation after the squeeze is completed. Note that for any θ, $s(\theta)$ is a local minimum of the diameter function.

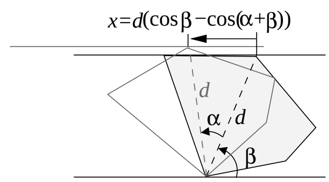
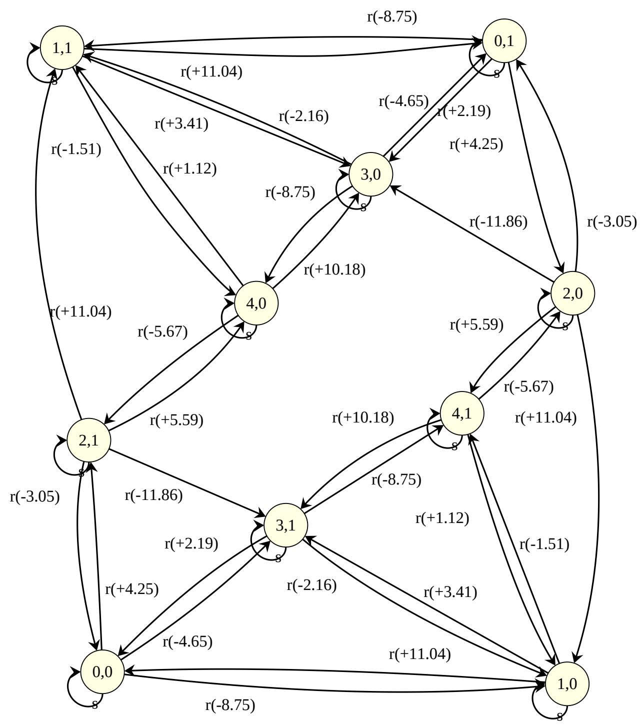


Fig. 4. Relationship between x, the translation of the upper jaw, and α, the change in orientation during a roll operation.

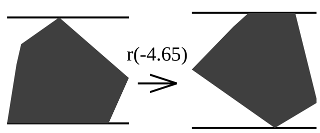
For the roll operation we can define a parametrized family of functions, $r_x : S^1 \to S^1$, such that if θ is the initial part orientation, $r_x(\theta)$ is the orientation after the upper jaw has been translated by x and the part has been squeezed. We define a local frame such that if x is greater than 0, the jaw moves to the left. A roll function corresponds roughly to a shifted squeeze function. Figure 2 shows these functions for a rectangular part. Suppose that during a roll the contact points do not change. For a given translation x of the upper jaw the change in orientation is then equal to $\alpha = \cos^{-1}(\cos\beta - x/d) - \beta$, where d is the distance between the contact points and β the angle between the x-axis and the line through the contact points. See figure 4. Note that x and α always have the same sign. If the contact points *do* change during a roll, we divide the roll into smaller steps (without squeezing) such that during each step the contact points do not change. The total change in orientation is simply the sum of changes during each step. It is not hard to see that for a given amount of translation the roll function is monotone in the orientation. We will use this property later.

IV. Planning Algorithm

Define a state as a pair $[e, j]$, where $e = 0, \ldots, n-1$ is the edge index and $j = 0, 1$ is the jaw index. In state $[e, j]$ edge e is aligned with jaw j. Since there are two jaws, an n-sided polygon can be in $2n$ states. The squeeze and roll primitives



(a) Transition graph



(b)
Transition from state [0, 0] to [3, 1]

Fig. 5. Each node consists of a pair $[e, j]$, where $e = 0, \ldots, n-1$ is the edge index and $j = 0, 1$ is the jaw index.

are closed under the set of part states: any primitive will map from any state to another state in the set. For each state we compute the minimum and maximum amount of translation needed to make a clockwise and counter-clockwise transition to the next stable edge. If the upper jaw translates less than the minimum, a subsequent squeeze brings the part to the original state. If the upper jaw translates any amount between the minimum and maximum, the subsequent squeeze will bring the part in the neighboring state. Translating more than the maximum can bring the part to any state. So these minima and maxima correspond to critical points where the outcome of a roll operation will change. Now consider the sorted list of *all* critical points for all states. To determine the possible outcomes of a roll operation applied to a set of states, it is sufficient to look up the outcomes at all the midpoints between consecutive critical points. In other words, even though we can roll a part by any continuous value, we only need to consider a finite set of $O(n)$ roll functions. This is not immediately obvious. Suppose we have only three critical points: two small ones and one that is larger than the sum of the first two. Then there exists at least one more critical point between the large critical points and the small ones. As we will show below, we are still able to orient the part with just the $O(n)$ roll functions above. The roll and squeeze functions induce a labeled graph on the states, where each edge is labeled with the appropriate function. Figure 5 shows an example of such a graph for a given part.

Based on the roll and squeeze functions we can also construct a graph where the nodes are *sets* of states, which we will call *hyperstates*. There exists an edge from node v to v' if and only if v' is the *forward projection* of v for a given operation [15]. In other words, v' is the smallest set such that this operation maps each element of v into v'. Each edge is labeled with the corresponding operation. The goal is now to find a path in this graph from the set of all states to a set with as few elements as possible. The goal set will always have at least two elements, because we can not distinguish any two orientations that are 180 degrees apart.

We are interested in finding the shortest paths from the node representing all states to nodes with a minimal number of states. These paths correspond to plans that orient a part with the smallest number of operations. Natarajan [30] was the first to analyze the complexity of this problem. He showed that given k functions a plan can be found, if one exists, in time $O(kn^4)$. Eppstein [14] presented an algorithm that given k *monotone* functions finds the shortest plan in $O(kn^2)$ time. Goldberg [18] and Chen and Ierardi [13] improved on this bound for the special case where functions correspond to squeezes in a finite number of directions. In this case a plan of length $O(n)$ can be found in time $O(n^2)$. Recently, Berretty et al. [5] analyzed another monotone function called the *push* function. The push function, $p_\alpha : S^1 \to S^1$, when given an orientation θ returns the orientation of the part $p_\alpha(\theta)$ after it has been pushed from direction α by a fence orthogonal to the push direction. With the push operation it is possible to uniquely orient a part. Berretty et al. presented an $O(n^3 \log n)$ algorithm to find the shortest plan.

Since we have $O(n)$ roll functions and one squeeze function, $k = O(n)$ and with Eppstein's algorithm we can find a plan in $O(n^3)$. Below we prove that for asymmetric parts the final hyperstate always contains exactly two states.

It is possible that there exist many paths of the same length that lead to nodes with the same number of states. We can impose additional constraints to find the 'best' path. For instance, we might prefer squeeze operations over roll operations. Or we can minimize the total amount of translation required by a plan. Finally, we can select the path that is the most robust to errors in the jaw positions.

It is possible to orient a polygonal part by repeatedly applying the same operation. Consider the set of minimal distances such that a counter-clockwise roll with this distance as parameter will cause a transition to another state. Let d be a distance between the second largest and largest elements of that set. If we perform a roll operation with distance d $n-1$ times, the part will be in the state corresponding to the largest

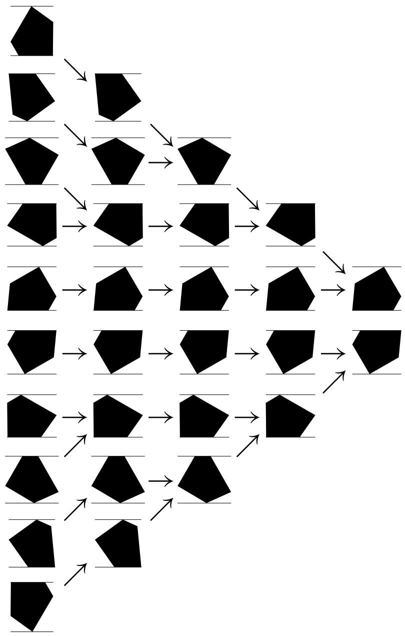


Fig. 6. For parts with many stable grasps many operations are needed to orient it. The plan used in this figure is the following sequence of roll operations: $r_{2.44}, r_{1.97}, r_{1.18}, r_{3.53}$.

element of the set. So we can compute a plan of length $n-1$ that will orient a part in linear time. An advantage of this method is that it is easy to compute and possibly easier to implement. A disadvantage, though, is that it *always* requires $n-1$ operations, whereas if we search for the shortest path we are often able to find a sequence of operations that orients the parts in very few steps. In the next section we explore the average number of steps needed to orient a part in more detail. Note that at least one of the midpoints between critical points satisfies the constraints on d. Therefore Eppstein's algorithm will also find a path to a node representing two states, in the case of asymmetric parts.

V. Random Polygonal Parts

To find out what kinds of polygons take many operations to orient, we tested the algorithm on a set of random convex polygons. Following [22], we generate random convex polygons in the following way. We can regard a convex n-sided polygon as a set of n vectors subject to the constraint that the vectors add up to the zero vector. We pick the x-coordinates (and y-coordinates) of the vectors as follows: we pick a uniformly random point inside a $n-1$ dimensional hypersphere and rotate the resulting point (padded with a zero to make its length n) to lie in the hyperplane defined by $\sum_{i=1}^{n} x_i = 0$. The vectors can be computed in $O(n)$ time. To create a polygon we need to sort the vectors by angle, causing the overall run time to be $O(n \log n)$.

In figure 6 an example is shown of a 5-sided polygon that requires 4 operations to orient it. The polygon is squeezed first, followed by three roll transitions. In general, the more stable grasps a part has, the more operations will be needed to orient it. If *all* states correspond to stable grasps, squeezing the part has no effect and only roll operations can be used to orient it.

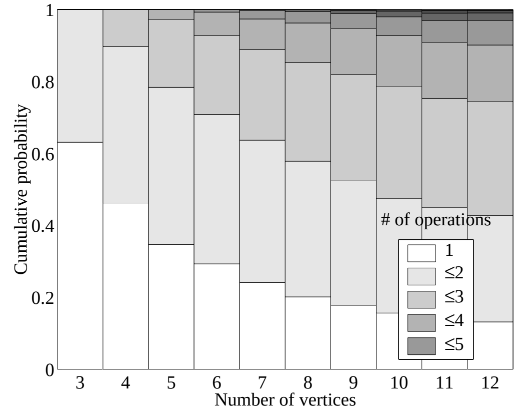


Fig. 7. Plot of number of operations vs. number of vertices. Cumulative distribution function of number of operations required to orient a part with 3, 4, …, 12 vertices.

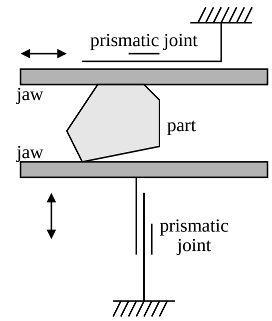


Fig. 8. The kinematics of a squeeze-roll micro-gripper

Under the probability distribution function induced by the algorithm described above, we found that almost all random polygonal part shapes can be oriented in less than four operations. In figure 7 is shown the cumulative distribution function of the number of operations required to orient a random polygon. This function is computed by sampling 5000 polygons for a given number of vertices. From this figure it can be concluded that the expected number of operations needed is small and increases slowly with the number of vertices.

VI. Discussion

We proposed a solution for the problem of orienting micro-scale parts without sensing using limited degree-of-freedom manipulators. At the micro-scale attractive forces tend to dominate, which causes parts to stick to the manipulator. Our solution relies on two manipulation primitives that address the sticking effects: the *squeeze* and the *roll*. These primitives can be written as transfer functions that map orientations to orientations. We have presented a complete algorithm that computes a plan using these transfer functions to orient a polygonal part at the micro-scale.

Figure 8 shows the kinematics of a gripper that can perform the squeeze and roll microassembly primitives. Future work is required to implement these kinematics in a micro-scale gripper. The recent paper by [35] suggests that micro-scale prismatic joints can be approximated by bending movements of relatively long cantilevered jaws. If the bending of the jaw occurs sufficiently far away from where the jaw touches the part, it can be approximated by a pure translation at the contact point.

The algorithm has useful applications in the assembly and mass production of MEMS devices. Since the algorithm does not rely on sensing, we can apply a plan to many (identical) parts at the same time.

References

[1] Akella, S., Huang, W. H., Lynch, K. M., and Mason, M. T. (2000). Parts feeding on a conveyor with a one joint robot. *Algorithmica*, 26:313–344.

[2] Akella, S. and Mason, M. T. (1999). Using partial sensor information to orient parts. *Intl. J. of Robotics Research*, 18(10):963–997.

[3] Arai, F., Ando, D., Fukuda, T., Nonoda, Y., and Oota, T. (1995). Micro manipulation based on micro physics. In *Proc. 1995 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, pages 236–241.

[4] Arai, F., Kawaji, A., Luangjarmekorn, P., Fukuda, T., and Itoigawa, K. (2001). Three-dimensional bio-micromanipulation under the microscope. In *Proc. 2001 IEEE Intl. Conf. on Robotics and Automation*, pages 604–609, Seoul, Korea.

[5] Berretty, R.-P., Goldberg, K., Overmars, M. H., and van der Stappen, A. F. (1998). Computing fence designs for orienting parts. *Computational Geometry: Theory and Applications*, 10:249–262.

[6] Bicchi, A., Marigo, A., and Prattichizzo, D. (1999). Dexterity through rolling: Manipulation of unknown objects. In *Proc. 1999 IEEE Intl. Conf. on Robotics and Automation*, pages 1583–1588, Detroit, Michigan.

[7] Böhringer, K. F., Bhatt, V., Donald, B. R., and Goldberg, K. Y. (2000a). Algorithms for sensorless manipulation using a vibrating surface. *Algorithmica*, 26(3/4):389–429.

[8] Böhringer, K.-F., Donald, B. R., Kavraki, L. E., and Lamiraux, F. (2000b). Part orientation with one or two stable equilibria using programmable force fields. *IEEE Trans. on Robotics and Automation*, 16(2):157–170.

[9] Böhringer, K. F., Donald, B. R., and MacDonald, N. C. (1996). Upper and lower bounds for programmable vector fields with applications to MEMS and vibratory plate parts feeders. In Overmars, M. and Laumond, J.-P., editors, *Workshop on the Algorithmic Foundations of Robotics*, pages 255–276. A. K. Peters.

[10] Böhringer, K. F., Donald, B. R., and MacDonald, N. C. (1999a). Programmable vector fields for distributed manipulation, with applications to MEMS actuator arrays and vibratory parts feeders. *Intl. J. of Robotics Research*, 18(2):168–200.

[11] Böhringer, K. F., Fearing, R. S., and Goldberg, K. Y. (1999b). Microassembly. In Nof, S. Y., editor, *Handbook of Industrial Robotics*, chapter 55, pages 1045–1066. John Wiley & Sons, New York.

[12] Ceccarelli, M., Marigo, A., Piccinocchi, S., and Bicchi, A. (2000). Planning motions of polyhedral parts by rolling. *Algorithmica*, 26(4):560–576.

[13] Chen, Y.-B. and Ierardi, D. (1995). The complexity of oblivious plans for orienting and distinguishing polygonal parts. *Algorithmica*, 14.

[14] Eppstein, D. (1990). Reset sequences for monotonic automata. *SIAM J. Computing*, 19(3):500–510.

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

[16] Erdmann, M. A., Mason, M. T., and Vaněček, Jr., G. (1993). Mechanical parts orienting: The case of a polyhedron on a table. *Algorithmica*, 10:226–247.

[17] Fearing, R. S. (1995). Survey of sticking effects for micro parts handling. In *Proc. 1995 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, volume 2, pages 212–217, Pittsburgh, PA.

[18] Goldberg, K. Y. (1993). Orienting polygonal parts without sensors. *Algorithmica*, 10(3):201–225.

[19] Hitakawa, H. (1988). Advanced parts orientation system has wide application. *Assembly Automation*, 8(3):147–150.

[20] Kavraki, L. E. (1997). Part orientation with programmable vector fields: Two stable equilibria for most parts. In *Proc. 1997 IEEE Intl. Conf. on Robotics and Automation*, pages 2446–2451, Albuquerque, New Mexico.

[21] Koyano, K. and Sato, T. (1996). Micro object handling system with concentrated visual fields and new handling skills. In *Proc. 1996 IEEE Intl. Conf. on Robotics and Automation*, pages 2541–2548, Minneapolis, MN.

[22] Lambert, T. (1994). *Empty-Shape Triangulation Algorithms*. PhD thesis, Department of Computer Science, University of Manitoba, Winnipeg, Manitoba.

[23] Maeda, Y., Kijimoto, H., Aiyama, Y., and Arai, T. (2001). Planning of graspless manipulation by multiple robot fingers. In *Proc. 2001 IEEE Intl. Conf. on Robotics and Automation*, pages 2474–2479, Seoul, Korea.

[24] Marigo, A. and Bicchi, A. (2000). Rolling bodies with regular surface: Controllability theory and applications. *IEEE Trans. on Automatic Control*, 45(9):1586–1599.

[25] Marigo, A., Chitour, Y., and Bicchi, A. (1997). Manipulation of polyhedral parts by rolling. In *Proc. 1997 IEEE Intl. Conf. on Robotics and Automation*, pages 2992–2997.

[26] Mason, M. T. (1982). *Manipulator Grasping and Pushing Operations*. PhD thesis, AI-TR-690, Artificial Intelligence Laboratory, MIT.

[27] Mason, M. T. (1985). The mechanics of manipulation. In *Proc. 1985 IEEE Intl. Conf. on Robotics and Automation*, pages 544–548, St. Louis.

[28] Miyazaki, H. and Sato, T. (1996). Fabrication of 3D quantum optical devices by pick-and-place forming. In *IEEE 9th Int'l Workshop on MEMS*, pages 318–324, New York, NY.

[29] Moll, M. and Erdmann, M. A. (2001). Manipulation of pose distributions. In Donald, B. R., Lynch, K. M., and Rus, D., editors, *Algorithmic and Computational Robotics: New Directions*, pages 127–141. A. K. Peters.

[30] Natarajan, B. K. (1989). Some paradigms for the automated design of parts feeders. *Intl. J. of Robotics Research*, 8(6):98–109.

[31] Peshkin, M. A. and Sanderson, A. C. (1988). Planning robotic manipulation strategies for sliding objects. *IEEE J. of Robotics and Automation*, 4(5).

[32] Rao, A., Kriegman, D., and Goldberg, K. (1995). Complete algorithms for reorienting polyhedral parts using a pivoting gripper. In *Proc. 1995 IEEE Intl. Conf. on Robotics and Automation*, pages 2242–2248.

[33] Reznik, D., Moshkovich, E., and Canny, J. (1999). Building a universal part manipulator. In Böhringer, K. and Choset, H., editors, *Distributed Manipulation*. Kluwer.

[34] Saito, S., Miyazaki, H., and Sato, T. (1999). Pick and place operation of a micro object with high reliability and precision based on micro physics under SEM. In *Proc. 1999 IEEE Intl. Conf. on Robotics and Automation*, pages 2736–2743, Detroit, MI.

[35] Shimada, E., Thompson, J., Yan, J., Wood, R., and Fearing, R. (2000). Prototyping millirobots using dextrous microassembly and folding. In *Symposium on Microrobotics ASME Int. Mechanical Engineering Cong. and Exp.*, Orlando, FL.

[36] Sudsang, A. and Kavraki, L. (2001). A geometric approach to designing a programmable force field with a unique stable equilibrium for parts in the plane. In *Proc. 2001 IEEE Intl. Conf. on Robotics and Automation*.

[37] Van der Stappen, A. F., Goldberg, K., and Overmars, M. H. (2000). Geometric eccentricity and the complexity of manipulation plans. *Algorithmica*, 26(3):494–514.

[38] Whitney, D. E. (1982). Quasi-static assembly of compliantly supported rigid parts. *J. of Dynamic Systems, Measurement, and Control*, 104:65–77.

[39] Wiegley, J., Goldberg, K., Peshkin, M., and Brokowski, M. (1996). A complete algorithm for designing passive fences to orient parts. In *Proc. 1996 IEEE Intl. Conf. on Robotics and Automation*, pages 1133–1139.

[40] Zesch, W. and Fearing, R. S. (1998). Alignment of microparts using force controlled pushing. In *SPIE Conference on Microrobotics and Micromanipulation*, Boston, MA.

[41] Zumel, N. B. (1997). *A Nonprehensile Method for Reliable Parts Orienting*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.