Figure 13: Grasp plan to resolve cliques in the graph of stable diameters for the parts P_1, P_2, P_3 shown in the last figure.

Figure 12: The graph of stable diameters for the three parts P_1, P_2, P_3.

Figure 11: Storing good partitions for a graph in tabular form. The columns are the vertices of the graph and the rows are the good partitions. In considering good partitions for a particular clique of vertices C, look only at the columns under vertices in C. Partitions are specified by attaching labels to vertices. In a particular partition, for a particular set of vertices, two vertices belong to the same partition if and only if they have the same label associated with them in that partition (the actual label itself is not important). $\{v_1, v_2\} \uplus \{v_3\}$ is a good partition for $\{v_1, v_2, v_3\}$ generated by x_1. x_1 also generates the good partition $\{v_1, v_2, v_5\} \uplus \{v_3\} \uplus \{v_4\}$ for $\{v_1, \ldots, v_5\}$. x_1 does not generate a good partition for $\{v_1, v_2\}$. (Why? Because $x_1 \notin R((v_1, v_2))$.) The number of good partitions (number of rows), t, is $O(n^3)$.

Figure 10: σ-equivalence. At the top are different transfer functions with three steps each. Only the portion between $[0, \pi)$ (the period of symmetry) is shown. The transfer function on the left, $,_1$ has fixed points $\{0, 90, 130\}$ degrees in and for $,_2$ they are $\{0, 60, 120\}$. The diameter value to which each step collapses to is shown in parenthesized italics just above the step. Let the parts that correspond to these transfer functions be P_1, P_2, respectively. The graph of stable diameters for these two parts, $G(P_1, P_2)$ is shown below. A vertex is shown as a (part, stable orientation) tuple. For example, u is the vertex $(P_1, 0)$ and v is $(P_2, 0)$. $R((u, v))$ can be verified to be empty. P_1, P_2 are σ-equivalent with $(0, 0)$ being the witnessing orientations. Given parts P_1 and P_2, both in orientation 0, no grasp action can distinguish between them. $\{u, v\}$ is one maximal clique in $G(P_1, P_2)$. The other maximal clique consists of four vertices and all the six edges e in that clique have non-empty $R(e)$.

Figure 9: Transfer function of a rectangular part.

Figure 7: Two minimal polygons that have the same diameter function.

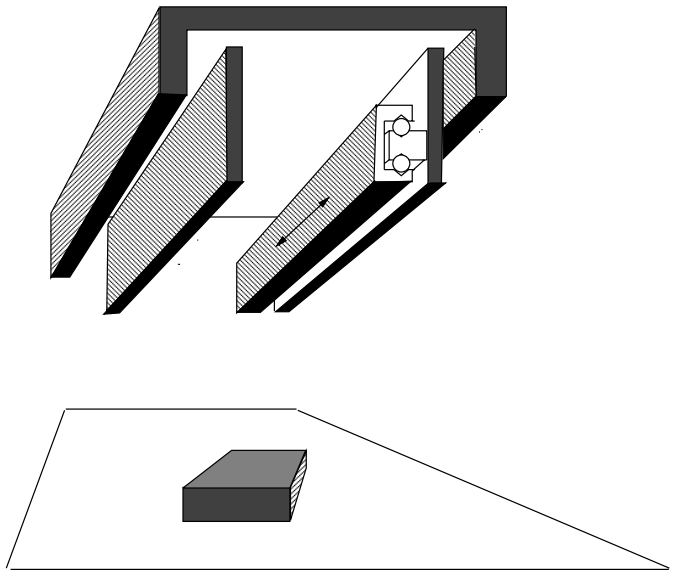

Figure 8: Schematic of the frictionless parallel jaw gripper poised above a rectangular part.

Figure 6: Infinitely many polygons having the same diameter function as a given polygon.

Polygons with Identical Diameter Function

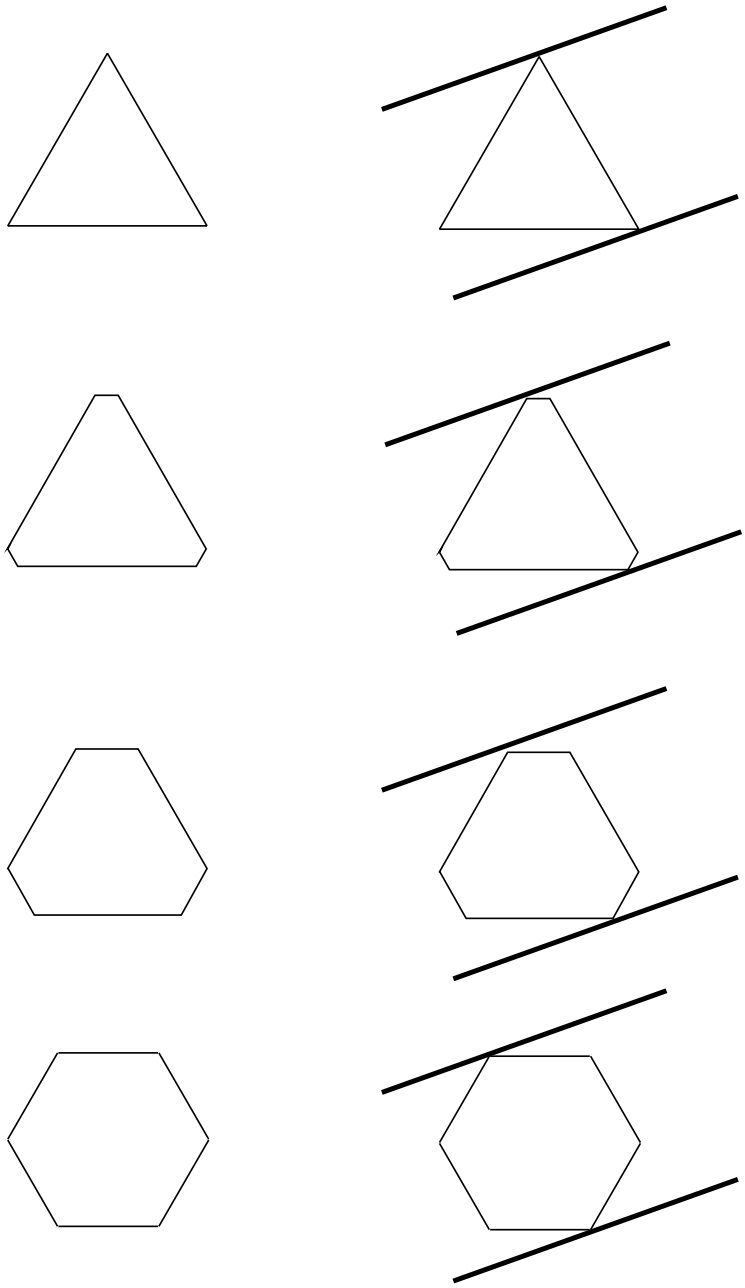

Figure 5: Triangles and hexagons having the same diameter function.

Figure 4: Infinitely many hexagons/octagons having same diameter function as given triangle/quadrilateral.

Figure 3: Orientation 0 is a kink.

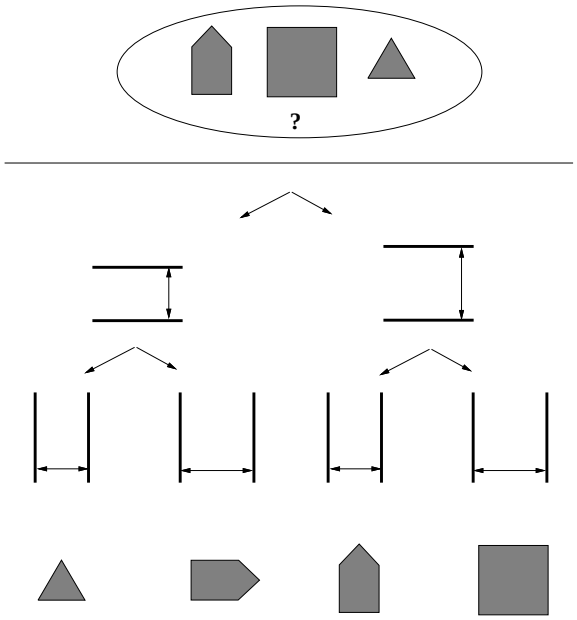

Figure 1: An example grasp plan for distinguishing the three parts shown at the top.

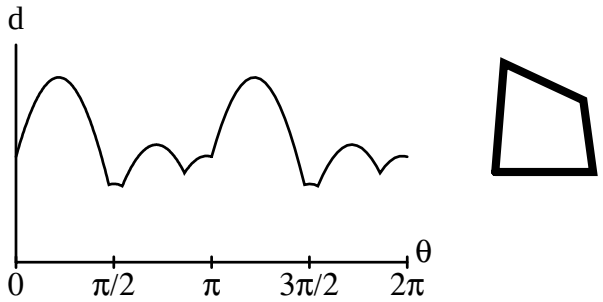

Figure 2: The diameter function for the four-sided part shown at the right.

Triple-spaced copy of text

List of Figures (with captions)

[Rao and Goldberg, 1992c] A. S. Rao and K. Y. Goldberg. Shape from diameter: Strategies for recognizing polygonal parts. Technical Report # 292, University of Southern California, Institute of Robotics and Intelligent Systems (IRIS), Los Angeles, Calif. 90089-0273, April 1992.

[Rao, 1992] A. S. Rao. *Algorithmic Plans for Robotic Manipulation.* PhD thesis, University of Southern California, Department of Electrical Engineering–Systems, December 1992.

[Rappaport, 1987] D. Rappaport. Computing simple circuits from a set of line segments is *NP*-complete. In *Annual Symposium on Computational Geometry*, pages 322–330. ACM, 1987.

[Raviv, 1991] D. Raviv. A quantitative approach to camera fixation. In *International Conference on Computer Vision and Pattern Recognition* (CVPR), pages 386–392, Maui, Hawaii, June 1991. IEEE.

[Rich, 1983] E. Rich. *Artificial Intelligence.* series in AI. McGraw Hill, New York, 1983.

[Skiena, 1988] S. S. Skiena. *Geometric probing.* PhD thesis, University of Illinois, Dept. of Computer Science, Urbana, Ill., 1988.

[Skiena, 1989] S. S. Skiena. Problems in geometric probing. *Algorithmica*, pages 599–605, 1989.

[Spyridi and Requicha, 1990] A. J. Spyridi and A. A. G. Requicha. Accessibility analysis for the automatic inspection of mechanical parts by coordinate measuring machines. In *International Conference on Robotics and Automation.* IEEE, May 1990.

[Taylor *et al.*, 1987] R. H. Taylor, M. T. Mason, and K. Y. Goldberg. Sensor-based manipulation planning as a game with nature. In *Fourth International Symposium on Robotics Research*, August 1987.

[Wallack and Canny, 1991] A. S. Wallack and J. F. Canny. Linear time algorithm for object localization using scanning. manuscript, September 1991.

[Wallack and Canny, 1992] A. S. Wallack and J. F. Canny. Object localization using finger gap sensing. Technical Report ESRC 92-3/RAMP 92-2, Univ. of California, Berkeley, Dept. of Computer Science, February 1992.

[Wallack and Canny, 1993] A. S. Wallack and J. F. Canny. A geometric matching algorithm for beam scanning. In *SPIE symposium on optical tools for manufacturing and advanced automation*, Boston, 1993.

[Yaglom and Boltyanskii, 1951] I. M. Yaglom and V.G. Boltyanskii. *Convex Figures.* Holt, Rinehart and Winston, New York, 1951.

[Goldberg and Furst, 1992] K. Y. Goldberg and M. Furst. Low friction gripper. U.S. Patent # 5,098,145, March 1992.

[Goldberg, 1990] K. Y. Goldberg. *Stochastic plans for robotic manipulation.* PhD thesis, Carnegie-Mellon University, School of Computer Science, August 1990.

[Goldberg, 1993] K. Y. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10(2):201–225, Aug 1993. (Special issue on Computational Robotics).

[Grimson and Lozano-Perez, 1987] W. E. L. Grimson and T. Lozano-Perez. Model-based recognition and localization from sparse range or tactile data. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 9:469–482, July 1987.

[Jameson, 1985] J. Jameson. *Analytic techniques for automated grasp.* PhD thesis, Stanford University, June 1985.

[Kang and Goldberg, 1992] D. Kang and K. Y. Goldberg. Shape recognition by random grasping. In *International Conference on Intelligent Robots and Systems.* IEEE/RSJ, July 1992. Submitted to the *IEEE Transactions on Robotics and Automation* in June 1993.

[Kolzow *et al.*, 1989] D. Kolzow, A. Kuba, and A. Volcic. An algorithm for reconstructing convex bodies from their projections. *Discrete and Computational Geometry*, 4:205–237, 1989.

[Koutsou, 1988] A. Koutsou. Object exploration using a parallel-jaw gripper. GRASP Lab Tech Report MS-CIS-88-48, University of Pennsylvania, 1988.

[Laumond, 1987] J. P. Laumond. Obstacle growing in a non-polygonal world. *Information Processing Letters*, 25(1):41–50, 1987.

[Li, 1988] S-Y. R. Li. Reconstruction of polygons from projections. *Information Processing Letters*, 28:235–240, 12 Aug. 1988.

[Mason *et al.*, 1988] M. T. Mason, K. Y. Goldberg, and R. H. Taylor. Planning sequences of squeeze-grasps to orient and grasp polygonal objects. Technical Report CMU-CS-88-127, Carnegie Mellon University, Computer Science Dept., Pittsburgh, PA 15213, April 1988.

[Mason, 1986] M. T. Mason. On the scope of quasi-static pushing. In O. Faugeras and G. Giralt, editors, *The Third International Symposium on Robotics Research.* MIT Press, 1986.

[Peshkin, 1986] M. A. Peshkin. *Planning Robotic Manipulation Strategies for Sliding Objects.* PhD thesis, Carnegie-Mellon University, Department of Physics, Pittsburgh, Pennylvania, Nov 1986. Also published as a book: *Robotic Manipulation Strategies*, Prentice Hall, 1990, New Jersey.

[Preparata and Shamos, 1985] F. P. Preparata and M. I. Shamos. *Computational Geometry, An Introduction.* Springer-Verlag, New York, 1985.

[Rao and Goldberg, 1992a] A. S. Rao and K. Y. Goldberg. Grasping planar curved parts with a parallel-jaw gripper. Technical Report # 299, University of Southern California, Institute of Robotics and Intelligent Systems (IRIS), Los Angeles, Calif. 90089-0273, August 1992. Submitted to the IEEE *Transactions on Robotics and Automation.*

[Rao and Goldberg, 1992b] A. S. Rao and K. Y. Goldberg. Orienting generalized polygonal parts. In *International conference on Robotics and Automation* (ICRA), Nice, France, May 1992. IEEE.

References

[Bajcsy, 1988] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8), 1988.

[Benson, 1966] R. V. Benson. *Euclidian Geometry and Convexity.* McGraw-Hill Book Company, 1966.

[Boissonnat and Yvinec, 1992] J. D. Boissonnat and M. Yvinec. Probing a scene of non-convex polyhedra. *Algorithmica*, 8:321–342, 1992.

[Boissonnat *et al.*, 1992] J. D. Boissonnat, D. Devillers, R. Schott, M. Teillaud, and M. Yvinec. Application of random sampling to on-line algorithms in computational geometry. *Discrete and Computational Geometry*, 8:51–71, 1992.

[Brost, 1988] R. C. Brost. Automatic grasp planning in the presence of uncertainty. *The International Journal of Robotics Research*, 8(1), February 1988.

[Canny and Goldberg, 1993] J. F. Canny and K. Y. Goldberg. A RISC paradigm for industrial robotics. Technical Report ESRC 93-4/RAMP 93-2, University of California at Berkeley, Engineering Systems Research Center, February 1993.

[Chandru and Venkataraman, 1991] V. Chandru and R. Venkataraman. Circular hulls and Orbiforms of simple polygons. In *Symposium on Discrete Algorithms* (SODA). SIAM-ACM, 1991.

[Chen and Ierardi, 1991] Y-B. Chen and D. J. Ierardi. Distinguishing polygons by sensing diameters. Technical Report USC-CS-92-503, University of Southern California, Dept. of Computer Science, December 1991.

[Clarkson and Shor, 1989] K. L. Clarkson and P. W. Shor. Applications of random sampling in computational geometry, II. *Discrete and Computational Geometry*, 4:387–421, 1989.

[Cole and Yap, 1987] R. Cole and C. K. Yap. Shape from probing. *Journal of Algorithms*, 8(1):19–38, 1987.

[Dobkin *et al.*, 1986] D. P. Dobkin, H. Edelsbrunner, and C. K. Yap. Probing convex polytopes. In *Symposium on Theory of Computing* (STOC), pages 424–432. ACM, 1986.

[Edelsbrunner, 1987] H. Edelsbrunner. *Algorithms in Combinatorial Geometry.* EATCS Monographs on Theoretical Computer Science. Springer-Verlag, Berlin, 1987.

[Ellis, 1987] R. Ellis. Acquiring tactile data for the recognition of planar objects. In *IEEE International Conference on Robotics and Automation*, 1987.

[Erdmann and Mason, 1986] M. A. Erdmann and M. T. Mason. An exploration of sensorless manipulation. In *IEEE International Conference on Robotics and Automation*, 1986.

[Garey and Johnson, 1979] M. Garey and D. B. Johnson. *Computers and Intractibility: A guide to the theory of* NP*-completeness.* Freeman, New York, 1979.

[Gaston and Lozano-Perez, 1984] P. C. Gaston and T. Lozano-Perez. Tactile recognition and localization using object models: The case of polyhedra on a plane. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 6:257–266, May 1984.

$$C_{i+1}\sin(\phi_{i+1} \Leftrightarrow \phi_i) = d_{i+2}\sin(\phi_{i+1} \Leftrightarrow \phi_i) \Leftrightarrow d_{i+1}\sin(\phi_{i+2} \Leftrightarrow \phi_i) + d_i \sin(\phi_{i+2} \Leftrightarrow \phi_{i+1}). \quad (6)$$

From the remaining $3Z \Leftrightarrow 3$ equations, we can similarly get $Z \Leftrightarrow 1$ other linear equations in the d_j. By observing the coefficients in the linear equations, we can see that they will have at most one solution (*i.e.* the case of infinite solutions is impossible).

Rote, Russ Taylor, and Chee Yap for rewarding discussions on this subject. We also thank the anonymous referees for their constructive criticism and useful suggestions.

A Unique solution to $3Z$ equations

Here we show that the $3Z$ equations from the proof of Theorem 2,

$$l_j \cos(\phi_j + \alpha_j) = d(\phi_j), l_j \cos(\phi_{j+1} + \alpha_j) = d(\phi_{j+1}),$$

$$t(\phi_j) = \frac{d(\phi_{j+1}) \Leftrightarrow l_{j-1} \cos(\phi_{j+1} + \alpha_{j-1})}{\sin(\phi_{j+1} \Leftrightarrow \phi_j)},$$

$0 \le j < Z$, have at most one solution.

Let us abbreviate $d(\phi_j)$ as d_j, and $t(\phi_j) \cdot \sin(\phi_{j+1} \Leftrightarrow \phi_j)$ as C_j. Notice that C_j are strictly positive and are "known" quantities. Finally, we introduce Z new variables $x_j = \phi_j + \alpha_j$ to replace the α_j.

Thus, the $3Z$ unknowns are d_j, l_j, x_j, and the equivalent $3Z$ equations are:

$$l_j \cos(x_j) = d_j, l_j \cos(x_j + \phi_{j+1} \Leftrightarrow \phi_j) = d_{j+1}, \tag{2}$$

$$d_{j+1} \Leftrightarrow l_{j-1} \cos(x_{j-1} + \phi_{j+1} \Leftrightarrow \phi_{j-1}) = C_j. \tag{3}$$

A solution must comprise of strictly positive l_j, d_j, and therefore by equation 2 x_j can be restricted to the range $(\Leftrightarrow \pi/2, \pi/2)$.

Suppose we know all the d_j. Then, the x_j, l_j can be uniquely determined by the following argument. Consider equations 2. If the d_j are known, we can eliminate the l_j into equations of the form (from 2)

$$\cos(\phi_{j+1} \Leftrightarrow \phi_j) \Leftrightarrow \sin(\phi_{j+1} \Leftrightarrow \phi_j) \tan(x_j) = \frac{d_{j+1}}{d_j}.$$

From these equations, the x_j can be uniquely determined in the range $(\Leftrightarrow \pi/2, \pi/2)$. Once the x_j are uniquely determined, so are the l_j.

So the problem reduces to showing that there is at most one solution for the d_j. We do this by constructing Z linear equations in the d_j alone.

Out of the $3Z$ equations in (2),(3), consider three particular equations:

$$l_i \cos(x_i) = d_i, l_i \cos(x_i + \phi_{i+1} \Leftrightarrow \phi_i) = d_{i+1},$$

and

$$C_{i+1} = d_{i+2} \Leftrightarrow l_i \cos(x_i + \phi_{i+2} \Leftrightarrow \phi_i).$$

From the first two, we get

$$d_i \cos(\phi_{i+1} \Leftrightarrow \phi_i) \Leftrightarrow [l_i \sin(x_i)] \sin(\phi_{i+1} \Leftrightarrow \phi_i) = d_{i+1}. \tag{4}$$

From the first and third, we get

$$C_{i+1} = d_{i+2} \Leftrightarrow d_i \cos(\phi_{i+2} \Leftrightarrow \phi_i) + [l_i \sin(x_i)] \sin(\phi_{i+2} \Leftrightarrow \phi_i). \tag{5}$$

Between equations 4,5, eliminate $[l_i \sin(x_i)]$ to get a linear equation in d_{i+2}, d_{i+1}, d_i:

Identifying σ-equivalent parts Our planning algorithms can identify any part (and its final orientation up to symmetry) from among a set of parts as long as no two are σ-equivalent. Testing for σ-equivalence of parts is straightforward from the graph of stable diameters G (Section 3.4). However, can we distinguish between σ-equivalent parts P_u, P_v (that are not d-equivalent)? While the chance that two industrial parts are σ-equivalent is very remote, this is nevertheless a theoretically interesting problem.[19] Let ϕ_u, ϕ_v be the orientations that witness the σ-equivalence For the edge $e = (u, v)$ in G, $R(e) = \emptyset$. Therefore, consider the set $R_2(e) \subset S^1 \times S^1$ defined by

$$R_2(e) = \{(\phi_1, \phi_2) \mid d_u(, {}_u(, {}_u(\phi_1 + \phi_u) + \phi_2)) \neq d_v(, {}_v(, {}_v(\phi_1 + \phi_v) + \phi_2))\}.$$

If $R_2(e)$ is non-empty, then a 2-tuple of orientations from $R_2(e)$ can be applied to disambiguate between the σ-equivalent parts. In fact, we can run the planning algorithms substituting $R_2(e)$ for $R(e)$. $R_2(e)$ can be computed in $O(|, {}_u||, {}_v|)$ time. If $R_2(e)$ is also empty, then $R_3(e) \subset (S^1)^3$ needs to be computed. Given the $R_2(e)$ for all edges, $R_3(e)$ can be computed again in $O(n^2)$ time and so on. Surely, for d-equivalent parts, all the $R_i(e)$ will be empty (a trivial bound on i is n^2) and no grasp plan can possibly distinguish between them. However, we conjecture that for σ-equivalent parts that are not d-equivalent, there exists a constant c such that $R_c(e) \neq \emptyset$. If true, this would result in an $O(n^2 \log n)$ planning algorithm, with $O(n^4)$ preprocessing, that gives a plan of length at most cn to distinguish between any set of parts as long as no two of them are d-equivalent.

Non-polygonal parts To extend these results to non-polygonal shapes, we note that one additional feature of non-polygonal diameter functions is the possible existence of flat regions in the diameter function. Also, the diameter function need not be piecewise sinusoidal anymore. It will be interesting to come up with an analog to Theorem 1 characterizing diameter functions of even a class of non-polygonal shapes, say the class of shapes made up of piecewise linear and piecewise circular segments (such parts are called *generalized polygonal* [Laumond, 1987; Rao and Goldberg, 1992b]). Shape recovery is another question. The existence of curves of constant width (or "orbiforms" [Chandru and Venkataraman, 1991]) is known since Euler. Hence complete shape recovery is impossible. However, can we have analogs to Theorem 2 stating how much information about the geometry of the shape can be recovered? If so, we may be able to come up with grasp plans to recognize generalized polygonal parts from among a set.

5 Conclusion

We have considered the planar problem of determining the convex shape of a polygonal part from a sequence of measurements taken with a frictionless parallel-jaw gripper. We have derived both negative and positive results.

The negative results are related to the study of curves of constant width (*Gleichdicke*) in classical geometry, establishing a new link between computational geometry and robotics. The positive results are examples of sensor-based manipulation planning, but differ from previous work in that we use a frictionless gripper and give a computational analysis of the planning problem. The required hardware is inexpensive and well-known. This is consistent with our aim to develop new software for "old" hardware.

Acknowledgments We thank Randy Brost, Duk Kang, Mike Erdmann, Doug Ierardi, Matt Mason, Babu Narayanan, Mark Overmars, Viktor Prasanna, Govindan Rajeev, Ari Requicha, Günter

[19] Our algorithms in this paper can distinguish between σ-equivalent parts as long as the parts do not enter into orientations that witness their σ-equivalence.

3.5.3 An example

Consider Fig. 12. Three polygonal parts P_1, P_2, P_3 are shown shaded and in zero orientation. P_1 is an equilateral triangle with altitude d_1; P_2, a house shaped part with two stable orientations, one with diameter d_1 and another with a $d_2 > d_1 + \epsilon$; and P_3 is a square of dimension d_2. N, the total number of faces of all the parts is 12. The graph of stable diameters for this set of parts, G, has $n = 7$ vertices, $v_1, v_2, \ldots, v_7$. G is shown just below P_1 in the figure. Vertices of G are indicated by solid circles: those associated with the stable orientations of P_1 with the smallest circles, P_2 slightly larger, and P_3 the largest. Each vertex v_i is shown associated with the following information: the part P_i, orientation ϕ_i and the diameter of P_i at orientation (in degrees) ϕ_i.

Recall that edges in G are between two stable orientations having the same diameter value. Thus G can be decomposed into two maximal cliques: $C_1 = \{v_1, v_2, v_3, v_4\}$ and $C_2 = \{v_5, v_6, v_7\}$. For any edge e in $\{v_1, v_2, v_3\}$, $R(e) = \emptyset$. Same is true for any edge in $\{v_6, v_7\}$. The $R(e)$ for other edges are not shown individually but it is easy to note that each of these $R(e)$ is non-null and includes the orientation[18] 90 degrees.

The graph for resolving the cliques in G is shown in Fig. 13. The nodes of the graph strategy tree are indicated as little square boxes such as $\Box$. There are 7 nodes in all forming a complete binary tree of depth 2. Each node is associated with a subgraph (shown enclosed in a rectangular box) of G, children associated with disjoint portions of the parent and the root associated with all of G. Every internal node is also associated with a orientation angle (shown enclosed in parenthesis) which indicates the angle the gripper must be rotated before the next grasp. For the root this can be any angle, wlog 0, since the first grasp is random. The possible choices for diameter measurement from the first grasp is d_1 or d_2. The two edges out of the root are therefore labelled d_1 and d_2, respectively. If in the first grasp we measure d_1, we travel down the left edge to a node associated with clique C_1 (and analogously if we measure d_2). The node associated with C_1 discovers that $R(\{v_1, v_2, v_3\}) = \emptyset$ and therefore computes the partition $\{v_1, v_2, v_3\} \uplus \{v_4\}$ generated by orientation $x = 90$. So we rotate the gripper by 90 degrees and grasp again. If measured diameter is still d_1, we have part 1, otherwise part 2. Likewise along other paths down the tree.

So, in this case of the three polygons, the complete strategy is as follows. Grasp, rotate by 90 and grasp again. If diameter values measured are (d_1, d_1), then we have part 1, if (d_2, d_2) its part 3, and part 2 otherwise. This was also shown in Fig. 1 earlier.

4 Future work

We would like to address the following issues in the future.

Length of grasp strategy The first planning algorithm we presented for recognizing polygonal parts returned a grasp plan with length no worse than q', q' being the size of the maximum clique in G. However, we believe that the average length of this grasp plan to be much better than q', perhaps closer to $\log q'$. The reason is that we believe the partition of S_i generated by a arbitrary x from a arbitrary non-empty $R(e)$ (Steps 2,3 in Planning Algorithm SUBOPTIMAL) to be fairly even (*i.e.* no set of the partition having more than $c|S_i|$ vertices, for some fixed $c < 1$). Thus, for practical implementations, perhaps this planning algorithm is more suitable. We would like to compare the length of the grasp plan computed by the two planning algorithms. Application of randomized constructions in geometry [Clarkson and Shor, 1989; Boissonnat *et al.*, 1992] would be useful in proving good expected behavior.

[18] Possibly not the best orientation : "best" meaning that allowing maximal sensor or rotation error.

Planning Algorithm OPT_GRASP

`for` $i = 1$ `to` q `do`:
  `for` $j = 1$ `to` N_i `do`:
    Let $C_{j,i}$ be the jth i-clique in G.
      `if` $i = 1$ or 2
        Do the needful [Trivial Cases]
      `else do`: [$i > 2$]
        1. $w \leftarrow |C_{j,i}|$; $h(C_{j,i}) \leftarrow w$.
        2. Consider all good partitions of $C_{j,i}$ in turn.
          2.1 Let $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ be one such good partition.
            2.2 Let $x \in R(E(T_1, T_2, \ldots, T_l))$.
            2.3 `if` $(1 + h(T_1, T_2, \ldots, T_l) < w)$ `do`:
              2.3.1 $w \leftarrow 1 + h(T_1, T_2, \ldots, T_l)$; $h(C_{j,i}) \leftarrow w$.
              2.3.2 Let $\{D_1, \ldots, D_l\} \leftarrow \{\ d_v(,\ _v(x + \phi_v)) \mid v \in C_{j,i}\ \}$
              2.3.3 For $1 \leq r \leq l$, $T_r = \{\ v \in C_{j,i} \mid d_v(,\ _v(x + \phi_v)) = D_r\ \}$.
              2.3.4 Remove all edges out the tree node $X(C_{j,i})$.
              2.3.5 Associate node $X(C_{j,i})$ by grasp angle x.
              2.3.6 Create l edges out of node $X(C_{j,i})$ with labels $D_1, \ldots, D_l$.
              2.3.7 Let the rth edge, $1 \leq r \leq l$, point to $X(T_r)$.

□

Analysis: Assume that at any stage of the computation, given a clique T, the information associated with it, namely, $h(T)$ and a pointer to $\rho(T)$, can be obtained by random access in $O(1)$ time. Also assume that we have precomputed the good partitions of every maximal clique and stored them in tabular form (as described in the proof Lemma 8 and Fig. 11). Steps 2 and 2.1 can be implemented by loading in the next row (partition) from the table. The test for "goodness" of this partition with respect to $C_{j,i}$ consists of checking whether the partition loaded in has at least two distinct labels for the columns corresponding to vertices in $C_{j,i}$. This check can be done in $O(i)$ time. Assume that the partition also brings with it its generator which may be used as x in Step 2.2 (without computation of $E(T_1, \ldots, T_l)$ and $R(E(T_1, \ldots, T_l))$. At this time subcliques $T_1, \ldots, T_l$ have been processed and so $h(T_1), \ldots, h(T_l)$ may be accessed and their maximum, $h(T_1, \ldots, T_l)$ computed in $O(l) = O(i)$ time for Step 2.3. A single iteration of steps 2.3.1–2.3.7 can be implemented in $O(l) = O(i)$ time. Let $T(p)$ represent the preprocessing time complexity to resolve a p-clique. Since there are at most $O(n^3)$ good partitions (Lemma 8), it is easy to see that

$$T(p) = \sum_{i=1}^{i=p-1} \left(\begin{array}{c} p \\ i \end{array} \right) O(in^3). \tag{1}$$

This gives $T(p) = O(n^4 2^p)$. Total preprocessing complexity is the sum of complexities for each maximal clique, which is upper bounded by $\frac{n}{q} n^4 2^q$, where q is the size of the maximum clique in G. The on-line grasp plan still runs in $O(\log n)$ per grasp. However, because to the representation of grasps directly as trees, its implementation is straightforward and is not described.

This exponential time complexity isn't as bad as it sounds since the size of the largest clique in the graphs we consider is likely to be small even if we have large number of parts. The correctness follows from the discussion before presentation of the planning algorithm and the fact that, at the end of the processing for $C_{j,i}$, the best partition for it would have been determined correctly (Step 2.3).

Let $\{T_j\}, 1 \leq j \leq l$ be the partition of T induced by the diameter values. The edges out of the root are labelled by the diameter values $D_1, \ldots, D_l$ and the children of the root are labelled by the vertices of the individual T_j and will be associated by an appropriate grasp angle.[17] The leaves of this tree consist of vertices that all belong to the same part or a clique of vertices such that every edge e between two vertices associated with distinct parts has $R(e) = \emptyset$. Let the subtree whose root $\rho(T)$ is labelled by the vertices of clique T be denoted as TREE(T). Additionally, let $h(T)$ denote the depth of TREE(T). It is easy to see that, at the end of the computation, this definition of $h(T)$ and that given in the previous paragraph are the same. Initially, the tree for every clique (including the non-maximal ones) consists of a single unconnected node labelled by the vertices of that clique.

Let N_i be the number of i-cliques in G. q is size of maximum clique in G.

[17] Each internal node is just like for the root – it stores the grasp strategy for the T_j it is labelled by.

e. Now consider an open interval $A_i = (\alpha_i, \alpha_{i+1})$. It is easy to see that, for every edge e, either $A_i \subset R(e)$ or $A_i \cap R(e) = \emptyset$.

By "a (good) partition of T generated by x", let us understand the following. Let $\{D_1, \ldots, D_l\}$ be the set of diameter values (ignoring duplicates) in $\{d_v(, {}_v(x + \phi_v)) | v \in T\}$. Then, the partition of T generated by x is $T_1 \uplus T_2 \uplus \ldots \uplus T_l$, where for $1 \leq i \leq l$, $T_i = \{v \in T | d_v(, {}_v(x + \phi_v)) = D_i\}$. The major claim is that all good partitions $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ in T are included among those generated by some $x \in A$. To verify this claim observe that the end-points of every interval of $R(E(T_1, T_2, \ldots, T_l))$ are present in A and also that *any* point[16] in $R(E(T_1, T_2, \ldots, T_l))$ suffices to generate the partition $T_1 \uplus T_2 \uplus \ldots \uplus T_l$. This proves the first statement of the lemma. The second statement follows from the simple fact that $R(E(T'_1, T'_2, \ldots, T'_{l'}))$, for the partition $T'_1 \uplus T'_2 \uplus \ldots \uplus T'_{l'}$ of any sub-clique T', also has all its interval end-points in A (due to simple inclusion).

This gives us a simple procedure to list out all the n^3 partitions of G, namely, form the set $A = \{\alpha_1, \alpha_2, \ldots, \alpha_t\}$ (can be done in $O(n^3 \log n)$ time by sorting once the $R(e)$ have been compute) and determine the partitions generated by every α_i, $1 \leq i \leq t$. Each such partition would require $O(n)$ time to compute, and $O(n \log n)$ time to sort by D_i value. Therefore, all the good partitions can be computed in $O(n^4 \log n)$ time. □

Corollary 1 *A partition $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ of T is good if and only if it is generated by some $x \in A$, where A and "generation of a partition" are as defined in the proof of Lemma 8.* □

Let us assume that all the (at most $t = O(n^3)$) good partitions of the cliques in G are stored in a table as shown in Fig. 11. The columns correspond to the n vertices of G. The rows give the various good partitions. For considering the good partitions of subset s of vertices, simply restrict your vision to the columns corresponding to those vertices (second statement of Lemma 8). Each good partition also stores its generator alongside. For example, x_1 generates the good partition $\{v_1, v_2, v_5\} \uplus \{v_3\} \uplus \{v_4\}$ for the clique $\{v_1, v_2, v_3, v_4, v_5\}$. More than one generator might exist for the same partition: notice that x_4 also generates the same partition as x_1 for $\{v_1, v_2, v_3, v_4, v_5\}$. A partition which was good for a clique may not be good for its subcliques: x_1 and x_4 do not generate a partition for $\{v_1, v_2, v_5\}$.

Our approach towards the optimal algorithm is to consider every clique in G, beginning with 1-cliques, then 2-cliques, working upwards until the maximal clique. When considering an i-clique T the aim is to compute its optimal grasp plan. This is done by considering every good partition $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ of T. The optimal grasp strategies for each of $T_1, T_2, \ldots, T_l$ have already been computed by this time because we have worked bottom-up. Let $h(T_i)$ be the grasp length of the optimal grasp plan for T_i. Define the *grasp length of the good partition* $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ to be

$$h(T_1, T_2, \ldots, T_l) = \max_{i=1}^{l}(h(T_i)).$$

Define the *best partition* of T to be the good partition of shortest grasp length. The optimal grasp plan for T can be computed from the optimal grasp plans of $T_1, T_2, \ldots, T_l$, where $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ is the best partition for T. $h(T)$ will be 1 plus the grasp length of its best partition.

Here, we store optimal grasp plans as a rooted tree, each node of the tree alongside a subclique (rather than lists alongside vertices as done previously). The root, $\rho(T)$, of the subtree storing the so-far-optimal grasp strategy for clique T is labelled by the vertices of T. The root is also associated with some angle x which is the grasp angle that generates the so-far-best partition of T. Let $\{D_1, \ldots, D_l\}$ be the set of diameter values (ignoring duplicates) in $\{ d_v(, {}_v(x + \phi_v)) \mid v \in T \}$.

[16] including the end-points

repeat:

0. Let v be any vertex in C_i,
1. Rotate the gripper by angle $X = x_{i,v}$.
2. Squeeze and measure the diameter D.
3. $C_{i+1} = \{v \in C_i | d_{i,v} = D\}$.
4. $i \leftarrow i + 1$.

until $|C_{i+1}| = |C_i|$.

$\nu \leftarrow i \Leftrightarrow 1$. Let $C_\nu = \{v_1, \ldots, v_z\}$ be of heterogeneity w.
Let $\{P_1, P_2, \ldots, P_w\}$ be the parts associated with the vertices $\{v_1, \ldots, v_z\}$.
Declare that the part P is one of $P_1, P_2, \ldots, P_w$.

□

Analysis: The subclique selection step, Step 3, is the dominant operation in every grasp and can be implemented in $O(\log n)$ time per grasp. The grasp length of the plan, μ, is bounded above by the size of the A_v, D_v lists, which is at most $q' = |C_1| \leq q \leq n$. (q is the size of the maximum clique in G).

3.5.2 Optimal grasp strategies

The reason why the Planning Algorithm SUBOPTIMAL of section 3.5.1 does not produce the shortest grasp plan is that the partitions $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ (Step 6) were chosen on basis of an *arbitrary* cutting edge e and *arbitrary* cutting orientation x_e (Steps 2,3).

Let $E(T_1, T_2, \ldots, T_l)$ denote the set of edges with one end point in one of the T_i and the other end-point in one of the other T_j. That is,

$$E(T_1, T_2, \ldots, T_l) = \{e = (u, v) \mid u \in T_i, v \in T_j, 1 \leq i, j \leq l, i \neq j\}.$$

To get the optimal number of grasps, one approach is to consider every partition of every subclique. However, even the number of 2-partitions of an i-clique is very large ($\Theta(2^i)$). Therefore this naive approach would lead to an algorithm that requires at least

$$\sum_{i=1}^{i=q-1} \left(\begin{array}{c} q \\ i \end{array} \right) \Omega(2^i) = \Omega(3^q)$$

operations. However, notice that we only need to consider "good" partitions $T_1 \uplus T_2 \uplus \ldots \uplus T_l$, *i.e.* those satisfying $R(E(T_1, T_2, \ldots, T_l)) \neq \emptyset$. The following lemma gives a polynomial bound on the number of good partitions.

Lemma 8 *There are only $O(n^3)$ good partitions $T_1 \uplus T_2 \uplus \ldots \uplus T_l$ of any maximal clique T in G. Furthermore, the only good partitions $T'_1 \uplus T'_2 \uplus \ldots \uplus T'_{l'}$ of a subclique $T' \subset T$ are the aforementioned $O(n^3)$ partitions of T restricted to the vertices of T'. Finally, these $O(n^3)$ partitions of maximal clique T can be computed in $O(n^4 \log n)$ time.*

Proof: Lemma 5 implies that every $R(e)$ has a complexity of $O(n)$. This implies that there exists a constant c so that each $R(e)$ has no more than cn intervals[15] over $[0, \pi)$. Consider the set of left end points and right end points of all the $R(e)$ as a sorted (in increasing order) set of points $A = \{\alpha_1, \alpha_2, \ldots, \alpha_t\}$. Note that t is at most $cn\frac{n(n-1)}{2} = O(n^3)$ because there are $\frac{n(n-1)}{2}$ edges

[15] We may assume all closed intervals.

PUSH clique S_i onto the queue Q.
while $(Q \neq \emptyset)$ **do**:

1. POP a clique T from Q.
2. Pick arbitrary edge $e = (u, w)$ in T satisfying $R(e) \neq \emptyset$. If no such e, go to Step 1.
3. Pick an arbitrary orientation $x_e \in R(e)$.
4. For every vertex $v \in T$,
   4.1 (compute new orientation) $\phi_v \leftarrow , _v(\phi_v + x_e)$; Let $d'(v) \leftarrow d_v(\phi_v)$.
   4.2 (update associated lists) $A_v \bowtie x_e$; $D_v \bowtie d'(v)$.
5. Sort the vertices v of T by their d' value. (T is no longer a clique because of Step 4.1)
6. Let the partition induced on T by d_v be $T_1 \uplus T_2 \uplus \ldots \uplus T_l$.
7. PUSH each sub-clique $T_j, 1 \leq j \leq l$, with $|T_j| > 1$, onto Q.

□

Analysis: Because of the choice of x_e and Lemma 6, there will be at least two distinct $d'(v)$ values arising from the vertices of the clique T (Step 4.1). That is, $l \geq 2$ (Step 6) and $|T_j| \geq 1, 1 \leq j \leq l$. Step 4 is assumed to be done in parallel for each vertex. Computing the new orientation (Step 4.1) could be understood as a vertex migration: vertex $(\phi_v, P_v, d_v, , _v)$ before Step 4.1 behaves as vertex $(, (\phi_v + x_e), P_v, d_v, , _v)$ after this step. Therefore, after Step 4.1, notice that we could get duplicated vertices, *i.e.* more than one vertex corresponding to the same orientation of the same part. In such a case, we eliminate all but one of these vertices. The queue always consists of mutually disjoint cliques only. In Step 2, if there is no edge e with $R(e) \neq \emptyset$, then it implies that the clique T is unpartitionable, *i.e.* the set of parts $\{P_u | u \in T\}$ are pairwise σ-equivalent all at their witnessing orientations. However, Step 2 can be accomplished in linear time because:

Lemma 7 *Let v be some vertex in a p-clique T. If all the $p \Leftrightarrow 1$ edges e from T incident on v have $R(e) = \emptyset$, then every edge f in T has $R(f) = \emptyset$.* □

Proof:

Suppose there is an edge $f = (u, w)$ with $R(f) \neq \emptyset$. Let $x \in R(f)$. Then, by Lemma 6, $x \in R((u, v))$ or $x \in R((w, v))$ which contradicts $R(e) = \emptyset$ for every edge incident on v in T.

□

The sorting in Step 5 becomes the dominant operation in each iteration. Hence, the worst case time complexity to empty the queue starting with the p-clique T is $O(p \log p + (p \Leftrightarrow 1) \log(p \Leftrightarrow 1) + \ldots) = O(p^2 \log p)$ time. Repeating this process for every maximal clique S_i gives us the overall complexity of $O(n^2 \log n)$ time for this planning algorithm. Correctness essentially follows from the Lemma 6 that guarantees a partition of T by grasp action x_e.

The lists A_v, D_v associated with a vertex v of a clique C give the grasp plan to resolve the clique C. This on-line grasp plan is now presented.

Let $A_v = \{x_{1,v}, x_{2,v}, \ldots\}$ be the list of angles associated with vertex v. $D_v = \{d_{1,v}, d_{2,v}, \ldots\}$ is the corresponding list of associated diameter values.

(On-line) Grasp Strategy SUBOPTIMAL

INPUT: Planar part P lying flat on table.
Grasp P and measure the diameter D. Hash on to the appropriate (maximal) clique C_1.
i.e. $C_1 = \{v \in G | d_v(\phi_v) = D\}$, $|C_1| = q'$.
Set $i \leftarrow 1$.

Resolving edges. To resolve a heterogeneous 2-clique $\{u, v\}$ connected by the edge e, where $R(e) \neq \emptyset$, simply rotate the gripper by any angle[12] in $R(e)$, squeeze the part and measure the diameter. By the definition of $R(e)$, one can now identify the part based on this last value measured.

If $R(e) = \emptyset$, the two parts are σ-equivalent and no grasp can distinguish between them. σ-equivalent parts include parts with identical diameter function.

Resolving cliques with more than 2 vertices. First, extend R, the resolving set of orientations, to an induced subgraph or any set of edges Z by defining[13]

$$R(Z) = \bigcap_{e \in Z} R(e).$$

If $R(C) \neq \emptyset$, for a clique C, do as before, *i.e.* rotate the gripper by any[14] orientation in $R(C)$ and squeeze. The measured diameter, by the definition of $R(C)$, can identify the part. However, if $R(C) = \emptyset$, a single additional grasp will not suffice.

Resolving cliques C with $R(C) = \emptyset$ is the subject of the rest of this paper.

3.5.1 An efficient planning algorithm resulting in a short but suboptimal strategy

We first present an planning algorithm that requires an off-line $O(n^2 \log n)$ preprocessing time and produces a grasp plan requiring no more than q' grasps, where $q' \leq n$ is the size of the maximal clique in G that the part 'hashes' onto after the first grasp.

The off-line planning algorithm computing the grasp plan is presented now. Let two lists A_v, D_v, both initialized to $\emptyset$, be associated with every vertex $v \in G$. Let Q be a queue of cliques initialized to $\emptyset$. Operations on Q are "PUSH", inserting a clique at the tail of the queue; and "POP", removing a clique from the head of the queue. Let $A \bowtie a$ denote that element a being inserted at the tail of list A. Let disjoint union of two sets T_1, T_2 be denoted as $T_1 \uplus T_2$.

Lemma 6 *If the vertices $\{a, b, c\}$ form a 3-clique (possibly non-maximal) with $R((a, b)) \neq \emptyset$. Then, for every orientation $x \in R((a, b))$, x exists in one or both of $R((a, c)), R((b, c))$.*

Proof: Let diameter value $D_a = d_a(, {}_a(x + \phi_a))$. D_b, D_c are similarly defined. Since $x \in R((a, b))$, $D_a \neq D_b$. This implies that D_c is unequal to at least one of D_a and D_b, say $D_c \neq D_a$. Then, $x \in R((a, c))$ by definition of $R((a, c))$. □

(Off-line) Planning Algorithm SUBOPTIMAL

Let G be decomposed into s maximal cliques $S_1, S_2, ..., S_s$.
for $i = 1$ **to** s **do**:
  if $(|S_i| \neq 1)$ **do**:

[12] or by a "best" orientation $\phi_{\max}$ maximizing the absolute diameter difference, $|d_u(, {}_u(\phi_u + \phi)) - d_v(, {}_v(\phi_v + \phi))|$, over all $\phi \in S^1$. This would allow for maximum sensor error and is easy to determine during the computation of $R(e)$ given in the proof of Lemma 5 in section 3.4. However, if we want to allow for maximal error in gripper rotation, we can choose to take the middle orientation in the largest continuous interval of orientations in $R(e)$.

[13] If Z has heterogeneity $h < |Z|$, it is sufficient to consider the subset of edges $Z' = \{(u, v) | u, v \in Z, P_u \neq P_v\}$ and define $R(Z) = \bigcap_{e \in Z'} R(e)$. The given definition assumes the worst case of $h = |Z|$.

[14] or "best" orientation – but in this case best is harder to compute. In fact it reduces to a maximin problem : find $\phi \in R(C)$ that maximizes

$$\min_{u,v \in C} |d_u(, {}_u(\phi_u + \phi)) - d_v(, {}_v(\phi_v + \phi))|.$$

However, if we want to allow for maximal error in gripper rotation, we can choose to take the middle orientation of the largest continuous interval of orientations in $R(C)$.

Attributes on edges Unequality up to ϵ, $\overset{\epsilon}{\neq}$, is defined so that $a \overset{\epsilon}{\neq} b$ if $|a \Leftrightarrow b| > \epsilon$. Let $e = (u, v)$ be an edge in G. Then, the attribute associated with edge e is a subset of S^1, $R(e)$, defined by

$$R(e) = \{\phi \mid d_u(,\ {}_u(\phi_u + \phi)) \overset{\epsilon}{\neq} d_v(,\ {}_v(\phi_v + \phi))\}.$$

$R(e)$ may be termed the *set of resolving orientations* for edge $e = (u, v)$ for the following reason. Suppose we have one of the two part-orientation pairs: P_u in orientation ϕ_u or part P_v in orientation ϕ_v. These have the same diameter up to ϵ (because of the existence of edge e). However, applying a grasp action taken from $R(e)$ will result in different (up to ϵ) diameters identifying the part uniquely. The next lemma gives the complexity of computing $R(e)$. Let $|,|$ be the number of steps in transfer function , .

Lemma 5 *$R(e)$, where $e = (u, v) \in E$, can be constructed in $O(|,\ {}_u| + |,\ {}_v|)$ time.*

Proof Outline: $R(e)$, where $e = (u, v)$, can be seen to be

$$R(e) = \{\ \phi \in S^1 \mid \sigma_u^{(\phi_u)}(\phi) \neq \sigma_v^{(\phi_v)}(\phi)\ \}.$$

Therefore, first construct functions $\sigma_u^{(\phi_u)}$ and $\sigma_v^{(\phi_v)}$. Scan them from from left to right to determine $R(e)$. □

Theorem 6 *Given a set of planar polygonal parts $P_1, \ldots, P_k$ with N faces in total (and $n \leq N$ stable orientations), its internal representation, i.e. the graph of stable diameters, G, consists of n vertices and can be computed in $O(N + n^3)$ time.*

Proof Outline: In $O(N)$ time it is possible to compute n, the number of stable orientations of the parts. This determines the vertex set of G. The associated information for each vertex v can be computed in $O(n \log n)$ time. Summing over all vertices, we get $O(n^2 \log n)$ time. The edge set can now be computed in $O(n^2)$ time. Computation of each $R(e)$ takes $O(n)$ time and therefore, computation of all the $R(e)$ takes $O(n^3)$ time. □

Testing σ-equivalence If $R(e) = \emptyset$, for some edge $e = (u, v) \in G$, then the parts (P_u, P_v) are σ-equivalent at ϕ_u, ϕ_v, and vice versa. This gives a test for σ-equivalence between parts.

3.5 Planning algorithms and grasp strategies

For an induced subgraph $G' \subset G$, the cardinality of $\{P_v | v \in G'\}$ is called its heterogeneity. If this number is one, then the subgraph is termed homogeneous, and heterogeneous otherwise.

The first grasp on the part, which is randomly applied, and returns a diameter value which is "hashed" into one of the maximal cliques $C \subset G$. If this clique C is homogeneous, the part is identified. Otherwise additional grasping is required to resolve the heterogeneous clique. The next grasp should be a useful grasp giving us new information that will help identifying homogeneous sub-cliques. To determine useful grasps we need the attributes on the edges defined before.

3.3 Problem statement

The shape identification problem is defined below.

> Given a set of k polygonal parts, $\{P_1, P_2 \ldots, P_k\}$, no pair of which are σ-equivalent, with a total of N faces and n stable equilibrium orientations, $n \leq N$; find a sensing plan consisting of parallel-jaw gripper grasp actions for identifying each part such that, if X_i is the maximum length of a sequence of grasp actions to identify part P_i, then $\max_i X_i$ is minimized.

We call a tree of grasp actions as a *grasp plan* or grasp strategy. $\max_i X_i$ is called the *grasp length* of the grasp plan. The problem as stated above asks for the *optimal grasp plan*, a grasp plan of shortest grasp length. See Fig. 1 for an example in which each $X_i = 2$. Our results are two planning algorithms: one that constructs an optimal grasp plan (Section 3.5.2) in time $O(n^4 2^n)$, and the other constructs a suboptimal sensing plan (Section 3.5.1) in $O(n^2 \log n)$ time. Neither plan requires more than n on-line diameter measurements. Both the planning algorithms operate on an internal representation of the stable equilibrium orientations which is computable in $O(N + n^3)$ time (Section 3.4).

The problem as stated above requires no two parts in $\{P_1, P_2 \ldots, P_k\}$ to be σ-equivalent. The parts are identified uniquely in a known final orientation. While it is extremely unlikely that a given set of factory parts will contain σ-equivalent parts, detecting σ-equivalent pairs is straightforward after construction of our graph representation of the parts (Section 3.4).

3.4 Representation of stable orientations

Let the error in the linear position sensor be $\epsilon > 0$. All of the analysis that follows is based on a representation of the k parts as a *graph of stable diameters*, G_ϵ, which we now define. $G_\epsilon = (V, E_\epsilon)$ is an undirected graph with its vertex set V in a 1-1 correspondence with the set of stable equilibrium orientations (within $[0, \pi)$) of all the k polygons. Assume that every vertex v in G has the following information associated with it: ϕ_v, the stable orientation it corresponds to; P_v, the associated part; d_v, the diameter function of P_v; and $,_v$, the transfer function of P_v. Define the operation, $\stackrel{\epsilon}{=}$ (equality up to ϵ) as $a \stackrel{\epsilon}{=} b$ if $|a \Leftrightarrow b| \leq \epsilon$. An edge between two vertices exists if and only if the corresponding two stable orientations have diameter values equal up to ϵ. That is

$$E_\epsilon = \{(u, v) \mid u, v \in V, \ d_u(\phi_u) \stackrel{\epsilon}{=} d_v(\phi_v)\}$$

The graph becomes more dense as ϵ increases. In future, we drop the subscript ϵ on G, V and E for convenience.

Let G have n vertices and m edges. Let $|T|$, where T is an induced subgraph of G, denote the number of vertices in T. If ϵ is sufficiently small,[11] then G partitions itself into disjoint maximal cliques (completely connected subset of vertices). Otherwise, every connected component of G need not be a clique. For simplicity, in future we assume that all connected components are cliques. Similar analysis applies to non-clique components.

The maximal cliques of G can be computed in $O(n+m)$ time by finding connected components. Each maximal clique C has an associated diameter value δ_C which could be chosen as the average of the elements of the set $\{d_v(\phi_v) | v \in C\}$.

[11] more precisely, if $\epsilon < X/n$, where X is the smallest gap among the set of distinct stable diameters

3.1.1 The transfer function ,

The transfer function of a part , : $S^1 \to S^1$ records the change in orientation of a part after it is grasped (refer to Assumptions 4-6 above). That is, if θ denotes an orientation of P with respect to G in the contact state, , (θ) denotes the orientation in the grasped state. The connection between the diameter and transfer functions is the following rule.

General principle of frictionless squeezing: The grasping process tends to (locally) minimize diameter.

This gives a simple procedure for computing the transfer function from diameter function. Let x, y be adjacent local maxima in d and let z be the unique local minima between them. Then all initial orientations in the region between x and y fall to z after the grasp. Each such region between two adjacent local maxima is called a *step*. In [Rao and Goldberg, 1992a] we show that, with a minor modification to the grasping process (replacing one grasp with three mini-grasps), all steps may assumed left-closed and right-open. Thus $\forall\, \phi \in [x, y)$, , $(\phi) = z$. For polygonal parts , is a step function with a fixed point[9] in the interior of each step. The points of discontinuity in , correspond to local maxima in d and fixed points correspond to local minima. See Fig. 9 for an example.

Orientations that are local minima in the diameter function, or equivalently, fixed points in the transfer function, are called stable (equilibrium) orientations. From Lemma 1, it is clear that an n-gon has at most n stable orientations.

3.2 The sensor function and σ-equivalence

Given a part P with known diameter function d and transfer function , , its *sensor function*, $\sigma : S^1 \to \Re_+$ is the composition of d and , , *i.e.* $\forall\theta,\ \sigma(\theta) = d(, (\theta))$. For polygonal parts, if $|, |$ denotes the number of steps in transfer function , , then σ is a step-function with at most $|, |$ steps.[10] Given P initially in orientation θ with respect to the gripper, a grasp action at α will return a measurement of $\sigma(\alpha + \theta)$. The new orientation, θ' of the part with respect to the gripper will be , $(\alpha + \theta)$.

Let f be some function whose domain is S^1. Then $f^{(x)}$ denotes the function f normalized with respect to orientation x, *i.e.*

$$\forall\theta \in S^1,\ f^{(x)}(\theta) = f(\theta + x \bmod 2\pi).$$

Two parts P_A, P_B with sensor functions σ_A, σ_B are σ-equivalent if there exist stable orientations β_A, β_B of P_A, P_B, respectively, such that

$$\forall\theta \in S^1,\ \sigma_A^{(\beta_A)}(\theta) = \sigma_B^{(\beta_B)}(\theta).$$

If two parts P_A, P_B are σ-equivalent, we refer to the orientations β_A, β_B in the above definition as the witnessing orientations for the σ-equivalence. Alternatively, we say that the parts P_A, P_B are σ-equivalent at orientations β_A, β_B. Replacing σ by d or , , we obtain definitions for parts being d-equivalent and , -equivalent, respectively. Notice that d-equivalence implies , -equivalence and also σ-equivalence. However, σ-equivalence does not imply , -equivalence. See Fig. 10.

[9] An orientation z such that , $(z) = z$.

[10] σ could have less than $|, |$ steps if two adjacent steps in , result in the same diameter.

of length at most n. The second planning algorithm (Section 3.5.2) runs in $O(2^n n^4)$ time and produces a grasp plan of optimal length. Both grasp strategies require $O(\log n)$ on-line time per grasp. This can be improved to constant time by using efficient hashing techniques.

3.1 Gripper and the grasping process

By a *parallel jaw gripper* (denoted by J) we understand a gripper consisting of two linear jaws arranged in parallel. Let L, H denote the lower jaw and upper jaw, respectively. Further assumptions:

1. The part P is a rigid planar polygonal object resting flat on a table.
2. The part's initial position is unconstrained as long as it wholly lies between the two jaws. The part remains between the jaws throughout grasping.
3. The motion of the gripper G is orthogonal to the jaws L, H.
4. All motion occurs in the plane and is slow enough that inertial forces are negligible. The scope of this *quasi-static* model is discussed in [Mason, 1986; Peshkin, 1986].
5. Both jaws make contact simultaneously (pure squeezing).
6. Once contact is made between a jaw and the part, the two surfaces remain in contact throughout the grasping motion. The action continues until further motion would deform the part. The part and the gripper are now said to be in the *grasped state*.
7. The gripper can be rotated about an axis orthogonal to the table.
8. There is zero friction between the part and the jaws. See [Goldberg, 1990] for a design validating this assumption. Essentially, this can be achieved by incorporating one of the jaws with a frictionless bearing.
9. In a grasped state, the distance between L, H is measurable by a linear position sensor up to an error ϵ.

Most of these assumptions are essentially the same as those made in [Goldberg, 1993; Rao and Goldberg, 1992b] (works dealing with orienting parts) except Assumption 9 which is new. In Assumption 7, it is not necessary that the gripper be rotatable precisely. Some margin in error of part orientation with respect to the gripper is allowed. This issue is studied in [Goldberg, 1993]. Also, looking ahead at the planning algorithms that follow in Section 3.5, we may note that we always choose orientations to rotate the gripper from among a set of intervals of orientations such that any orientation from this set is applicable. Thus, if we have an inaccurately rotating gripper, we may choose the middle most orientation in the largest interval to allow for maximal error in rotation. Assumption 5 can be relaxed to *push-grasp* actions [Brost, 1988] in which one of the jaws, say L, first makes contact with the part and pushed the part against the other jaw before squeezing. For simplicity, of presentation, we retain this assumption of pure squeezing. See Fig. 8.

3 Positive results : recognizing polygonal parts from a known set

The previous sections discussed some negative results for shape recovery from diameter function. However, note that we assumed nothing about the shape of the part except for it being a polygon. In this section, we consider the following problem. We have a part P whose shape is unknown but for being one of k planar polygonal parts $Q_1, \ldots, Q_k$ of known shapes. Furthermore, P is known to lie flat on a table between the jaws of a frictionless parallel jaw gripper equipped with a linear position sensor capable of measuring, up to an error ϵ, the distance between the jaws. The gripper can be rotated about an axis perpendicular to the table on which the part rests. The problem is to determine the shape of P using a minimum number of diameter measurements on the part.

The determination of P is up to σ-equivalence between parts – a notion that will be defined formally in Section 3.2. For now we could understand it in light of the negative results – Theorems 3,4 – presented earlier. For example, parts that have the same diameter function and are indistinguishable by diameter measurements alone. Also, the sensor error ϵ can play a factor in reducing distinguishability between parts.

Section 3.4 gives a simple test to check whether two parts are σ-equivalent. This test may be applied *a priori* to the set of given parts and we replace each subset of σ-equivalent parts by a single representative part (any single part from the subset). Thereafter we may deal with the problem of recognizing parts from among a set of parts, no two of which are σ-equivalent.

We begin in section 3.1 with a brief description on the mechanics of the low-friction parallel jaw gripper[8] and by defining the notion of a *stable grasp* on the part by the gripper. The only diameter measurements allowed are those in which the part is in a stable grasp by the gripper. There are at most r stable grasps on an r-sided polygonal part. Section 3.4 discusses the representation of all stable grasps of the system (*i.e.* all stable grasps of the known parts $Q_1, \ldots, Q_k$) as a graph data structure. If N is the total number of faces in all the polygons, and $n \leq N$ the number of stable orientations, the computation of this data structure requires $O(N + n^3)$ time. We use the terms "planning algorithm" and "grasp strategy" to denote different entities. "Planning algorithm" (or "planner") refers to the off-line preprocessing done on the graph data structure. The output of this preprocessing is a "grasp strategy" (or "grasp plan" or just "plan") which is run on-line. The grasp strategy gives a sequence of angles at which grasps must be applied on the part in order to recognize it. In general, the next grasp angle will depend on the current diameter measured. The grasp strategy is "tree-like," every node of the tree being associated with some subgraph of G. The root is associated with the whole of G and children of a node are associated with disjoint portions of the subgraph of the parent. This partition of the subgraph associated with the parent is determined by the current diameter value measured. We begin at the root, and travel down along a path of the tree progressively refining the search by pruning the graph. At an intermediate node of the path, we choose the correct child to visit next depending on the diameter value measured. Search trees are popular in AI. For example see [Rich, 1983]. By the *length* of the grasp strategy (or plan), we understand the maximum depth (number of grasps) of any leaf in the tree representing the strategy.

There are several measures of complexity here : the preprocessing time (complexity of the "planning algorithm"); worst case number of required grasps of the grasp plan to identify the object ("height" of the tree representing the grasp plan); and amount of on-line work to be done per grasp. Section 3.5 discusses tradeoffs between preprocessing time and length of the plan. Given the data structure representing the stable orientations of the parts, we present two planning algorithms. One (Section 3.5.1) requires $O(n^2 \log n)$ time and produces a non-optimal grasp plan

[8] Further details, including the design of such a device, may be found in [Goldberg, 1990]

follows. Let multiset $I = I_0 = \{a_1, \ldots, a_n\}$ be an instance of *EQUI_PARTITION*. Let it have two identical elements A, A. Replace the pair A, A by four elements $A, M, A + 2M, 3M$, where $M = \sum_{i=1}^{n} a_i$, to form the instance I_1 containing $n + 2$ elements. Observe that I_1 has strictly less identical elements as I_0 and that I_1 is equi-partitionable if and only if I is (this is because the element $3M$ has to be in a different partition as the elements M and $A + 2M$). Continue this process creating the instances $I_2, I_3, \ldots$, containing progressively lesser identical elements until the instance I_k containing $n + 2k$ elements, $k < n$, is created containing no pair of identical elements. I_k is therefore a valid instance of *EQUI_SET_PARTITION*. Observe that each $I_j, j > 0$ so obtained satisfies the condition that I_j is equi-partitionable if and only if I_{j-1} is. Therefore, I_k is equi-set-partitionable if and only if $I = I_0$ is equi-partitionable. □

Now we define the following problem.

MINIMAL_POLYGON_FROM_DIAMETER_FUNCTION (MPFD):
Given an m, k-diameter function d, is there a minimal polygon P fully consistent with d?

Theorem 5 *MPFD is NP-Complete.*

Proof: We use the *NP*-completeness of *MPFS*. Let algorithm *MPFD(d)*, where d is an m, k-diameter function, return true (*resp.* false) according as whether there is (*resp.* is not) an $m + k, 0$ polygon P fully consistent with d.

Then we solve the *MPFS* problem using the following algorithm:
INPUT: description (orientations, lengths) of n planar segments, no two of which are parallel.
OUTPUT: true/false whether or not they form a convex polygon.

1. Let Φ be a circular list of the orientations of the input segments in sorted order. Let t be a list of the lengths of the segments sorted according to the order in Φ.

2. Use equations 6 (in Appendix A) to determine whether there is some (valid) diameter function d that has $\Phi(d) = \Phi$ and $t_{P'} = t$, where P' is some polygon fully consistent[7] with d.

   If there does not exist such a d (Equations 6 do not have a solution), return "FALSE" and exit.

3. We assume that there exists such a valid d. It is easy to compute d after solving equations 6. Now invoke *MPFD(d)*.

   Return "TRUE" if and only if *MPFD(d)* returned "true".

Complexity of the first step is clearly polynomial in n. The second steps involves forming and solving n linear equations in n unknowns, which basically involves inverting an $n \times n$ matrix which has polynomial complexity. Step 3 basically involves a call to *MPFD*. Therefore, if *MPFD* was polynomial time decidable, so would *MPFS*.

Correctness follows from the following. If the set of input segments forms a polygon, then the polygon would have to have a diameter function d which is easily computable given t, Φ, and d would have its minima and kinks exactly at orientations in Φ. If they did not form a polygon, then no diameter function would exist and this would be detected in Step 2. above. However, it could happen that a valid diameter function d exists for a polygon P' that has $\Phi_{P'} = \Phi, t_{P'} = t$, but P' could have parallel edges. To check for parallel edges, Step 3 calls *MPFD(d)*. If it returned true, then it implies the original set of segments form a convex polygon. And if the original segments form a polygon, Step 3 would be executed and the call to *MPFD(d)* would return "true". □

[7] Such a polygon exists since d is valid. However, P' could have parallel edges.

fixed length and orientation, does there exist an arrangement of them forming a convex polygon?

Lemma 3 *PFS is NP-complete.*

Proof Outline: *PFS* is clearly in *NP*. So it is sufficient to show that *PFS* is *NP*-hard. We do this by polynomial transformation from *PARTITION*, the following well-known *NP*-complete problem.

Given a multiset of n positive numbers $a_1, \ldots, a_n$, *is there a set* $S \subset \{1, \ldots, n\}$ *such that*

$$\sum_{i \in S} a_i = \sum_{i \in \{1,\ldots,n\}-S} a_i \ ?$$

Given an instance I of *PARTITION* having n elements, consider the following set I' of $n+2$ segments (each segment is described as (l, ϕ), where $l \in \mathcal{R}_+$ is its length and $\phi \in [0, \pi)$ is angle made by it with x-axis):

$I' = \{(a_i, 0) | a_i \in \mathrm{I}\} \cup \{(1, \pi/2), (1, \pi/2)\}$.

It can be shown easily that I has a partition if and only if the segments from I' form a rectangle. □

Corollary to Lemma 3 CPFS is NP-Complete.

Lemma 4 *MPFS is NP-complete.*

Proof Outline:[6] Again it is easy to see that *MPFS* is in *NP*. So it is sufficient to show that *MPFS* is *NP*-hard. We do this by polynomial transformation from the following variant of *PARTITION* which we will show *NP*-hard later.

EQUI_SET_PARTITION: *Given a set of n positive numbers* $\{a_1, a_2, \ldots, a_n\}$ *(no two of which are equal), n even,*

does there exist a set $S \subset \{1, 2, \ldots, n\}$ *with exactly* $n/2$ *elements such that* $\sum_{i \in S} a_i = \sum_{i \in \{1,\ldots,n\}-S} a_i$?

Given an instance I of *EQUI_SET_PARTITION*, a set of n positive numbers $\{a_i | 1 \leq i \leq n\}$, n even, consider the following set I' of segments each described as (x, y), where x (*resp.* y) is the length of the projection of the segment on the x-axis (*resp.* y-axis). The slope of this segment is $\tan^{-1} y/x$. $I' = \{(1, a_1), (1, a_2), \ldots, (1, a_n)\}$. Notice that no two segments are parallel since no two of the a_i are equal making I' a valid instance of *MPFS*. We now show that I is equi-set-partitionable if and only if the segments of I' can be arranged to form a minimal polygon.

If I is equi-set-partitionable, let $S \subset \{1, \ldots, n\}$ be the set of $n/2$ indices of one of the partitions. Arrange the $n/2$ segments $\{(1, a_i) \mid i \in S\}$ end on end as a convex chain in increasing order of segment slope. This chain has a net x-projection of $n/2$ and a net y-projection of $\sum_{i \in S} a_i$. Similarly, arrange the remaining segments, $\{(1, a_i) \mid i \notin S\}$, as another convex chain but in decreasing order of segment slope. This second chain has the same net x, y-projections as the first. Identify corresponding ends of the two chains to form a minimal polygon. Conversely, if the segments I' can be formed into a minimal polygon, cut the polygon at the lowest (leftmost) and highest (rightmost) vertices (such vertices are unique because no segment in I' is purely vertical or horizontal) to form two convex chains. Each chain has the same net x, y-projections. Taking S as the set of $n/2$ indices corresponding to the segments of any one chain shows that I is equi-set-partitionable.

EQUI_SET_PARTITION can be shown *NP*-hard by polynomial transformation from *EQUI_PARTITION*, a known *NP*-complete problem [Garey and Johnson, 1979]. *EQUI_PARTITION* is *PARTITION* with the additional constraint that the cardinality of S be $n/2$. Briefly, this transformation is as

[6] This simpler proof is due to Govindan Rajeev of the University of Tennessee who pointed it out based upon a preliminary version of this paper [Rao and Goldberg, 1992c].

basically involved showing that a particular length $t(\phi)$ could be split, in infinitely many ways, into two segments (in the polygon), both of orientation ϕ whose lengths sum up to $t(\phi)$. Thus, most of these polygons would have parallel edges of varying lengths. This suggests that we might define a representative polygon as one without any parallel edges satisfying a given diameter function. Two obvious questions arise: does there always exist a representative polygon for a given diameter function; and if a representative polygon exists, is it always unique? We try and answer these questions in this section. The latter question is answered in the negative in Theorem 4 by constructing a counter-example, and the former question is shown *NP*-complete in Theorem 5. A major lemma in proving our *NP*-completeness result is showing that the problem of arranging a set of line segments, no two of which are parallel, into a convex polygon is *NP*-complete. This bears some resemblance to the result of [Rappaport, 1987] which shows that the problem of drawing (additional) line segments to connect a collection of given fixed line segments (by their end-points) into a simple circuit is *NP*-complete. In our case, we allow the segments to translate and we do not allow additional line connecting segments.

A *minimal polygon* is a (convex) polygon without any parallel edges, *i.e.* an n, p-polygon with $p = 0$.

Theorem 4 *Minimal polygons satisfying a given diameter function are not unique.*

Proof: See Fig. 7. □

The implication of this theorem is that even for minimal polygons, it is impossible to completely recover its shape from the diameter function.

Given an m, k-diameter function d, if there exists a minimal polygon P with n edges having d as its diameter function, then by Lemma 1, $n = m + k$. First notice that t_P, Φ_P are extractable from the diameter function d. We have seen above that there could be more than one minimal P. There can be at most a finite number (2^{n-1}) of possible minimal P since P is constrained to be convex. Now we tackle the question whether there always exists such a minimal P. We show that deciding this question is *NP*-complete in Theorem 5. Before we get there, we have to show some other problems *NP*-complete.

By *arranging* a set of n planar segments $S_0, \ldots, S_{n-1}$, we mean translating them in the plane so that two segments are either not intersecting, or if they intersect, they do so only at their end-points. All sets of segments in this section are multisets, *i.e.* they could contain more than one identical element. Consider the following problems.

POLYGON_FROM_SEGMENTS (PFS):
Given a set of n planar line segments, $S_0, \ldots, S_{n-1}$, each with a fixed length and orientation, can they be arranged so as to form a polygon (n-gon)?

Note that *forming a polygon* is equivalent to forming a simple polygon since the only intersections we allow between segments are at their end-points. Also note that we do not have any restriction on the orientations of the edges (any number of them could be parallel). Thus minimality of polygons is not addressed just as yet.

CONVEX_POLYGON_FROM_SEGMENTS (CPFS):
Given a set of n planar line segments, $S_0, \ldots, S_{n-1}$, each with a fixed length and orientation, can they be arranged so as to form a convex polygon (n-gon)?

MINIMAL_POLYGON_FROM_SEGMENTS (MPFS):
Given a set of n planar line segments, $S_0, \ldots, S_{n-1}$, no two of which are parallel, each with a

We solve these $3Z$ equations for the $3Z$ unknowns $(\alpha_j, l_j, d(\phi_j))$. We know that *at least* one solution for the $3Z$ equations exists since d, d' are valid. However, there could exist more than one solution giving different d, d'. We show in Appendix A that at most one solution for these $3Z$ equations exists. Thus, the diameter function is uniquely constructed. □

In a sense, Φ_P, t_P is the *maximal non-redundant (and invertible) information* of the geometry of a polygon P obtainable from its diameter function, Φ_P giving the orientations of the faces of P, and t_P the perimeter along each orientation. However, the two lists do not completely determine P because there could be up to two faces along an orientation $t(\phi)$ and the $t(\phi)$ constraint is *only a constraint on the sum of the length of the two faces.* In fact, as Theorem 3 shows, there are infinitely many polygons P having the same valid diameter function (and hence the same Φ_P, t_P).

Diameter functions of parallelograms (4, 2-gons) are termed *trivial.*

Theorem 3 *For every non-trivial valid diameter function d there exist infinitely many polygons having diameter function d.*

Proof: Fig. 4 show this is true for diameter functions of triangles and quadrilaterals. See also Fig. 5. Towards the generalization assume that P is a polygon having diameter function d. P exists since d is valid. Let A, C be two vertices of P touching l, h at a maxima orientation. Let this maxima orientation be the zero orientation, WLOG. Let D, B be the vertices adjacent to C in P (*i.e.* DC, BC are two faces of P). Likewise, let D^*, B^* be the vertices adjacent to A. D^* and D are on the same side of AC (as are B^* and B). D^* (*resp.* B^*) could be coincident with D (*resp.* B). For example, in Fig. 4: the quadrilateral case, $D = D^*, B = B^*$. Let $\phi_1, \pi \Leftrightarrow \phi_2, \phi_3, \pi \Leftrightarrow \phi_4$ be the orientations of faces CB, AB^*, AD^*, CD, respectively. Without loss of generality assume $\phi_2 < \phi_4, \phi_1 < \phi_3$. The other cases (including equality) are treated similarly. See Fig. 6. F', E' are can be arbitrarily chosen on CB, AB^*, respectively. G' is such that CG' is parallel and equal to B^*E'. Thus we have $t(\phi_2) = |AB^*| = |AE'| + |CG'|$. H' is determined similarly. It is defined so that AH' is parallel and equal to BF'. Now, $t(\phi_1) = |CB| = |CF'| + |AH'|$.

A line is drawn parallel to AD^* (*resp.* (DC)) through H' (*resp.* G'). Points $D^{*\prime}, D'$ are chosen on these two lines so that the distance between $D', D^{*\prime}$ is equal to that between D^*, D. Now the portion of the polygon P between D^*, D can be moved over to between $D^{*\prime}, D'$.

$B^{*\prime}, B'$ are defined in a similar manner. First a line is drawn parallel to AD^* (*resp.* DC) through F' (*resp.* E'). $B^{*\prime}, B'$ are chosen on these lines so that the distance between them equals that between B^*, B. The portion of P between B, B^* can be moved over to between $B', B^{*\prime}$. If this causes any problems of convexity, then take the faces $F'B'$, $E'B^{*\prime}$, and those originally between B, B^*, sort them by orientation, and arrange them between F' and E'.

Simple geometry can be applied to show that $|H'D^{*\prime}|+|B'F'| = |AD*| = t(\phi_3)$ and $|G'D'| + |E'B^{*\prime}| = |DC| = t(\phi_4)$. For example to show that $|H'D^{*\prime}|+|B'F'| = |AD*|$, draw a line through H' parallel to $D^{*\prime}D'$ intersecting AD^* at Z. Now note that triangle $F'B'B$ is congruent to triangle $H'ZA$ and so $|B'F'| = |AZ|$. Also note that $H'ZD^*D^{*\prime}$ is a parallelogram, and so $|H'D^{*\prime}| = |ZD^*|$.

Thus, the two polygons $P \stackrel{\text{def}}{=} A, D^*, \ldots, D, C, B, \ldots, B^*, A$ and $P' \stackrel{\text{def}}{=} A, H', D^{*\prime}, \ldots, D', G', C, F', B', \ldots, B^{*\prime}, E', A$ have the same diameter function by Theorem 2 since $t_P = t_{P'}$ and $\Phi_P = \Phi_{P'}$. Finally note that there are infinitely many P' since the choices of F', E' (along a line segment) were arbitrary. □

2.1 Minimal Polygons

Theorem 3 is a negative result for shape recovery from diameter: there exist infinitely many polygons consistent with a given measured diameter function. However, the proof of the theorem

where L, α are the parameters of the sinusoid in d between $(0, \phi_1)$, i.e. $L\cos(\alpha) = d(0), L\cos(\alpha + \phi_1) = d(\phi_1)$.

Proof Outline: ($\Leftarrow$) It is enough to show that P is consistent with d at ϕ_1 as well. That is, we need to show that $d(\phi_1)$ can be obtained from $t(\phi_1)$. Consider the equations $L\cos(\alpha) = d(0)$ and $L\cos(\phi_2 + \alpha) = d(\phi_2) \Leftrightarrow t(\phi_1) \cdot \sin(\phi_2 \Leftrightarrow \phi_1)$.

L, α can be determined (uniquely) from these equations (this is also shown in Appendix A) and $d(\phi_1)$ can be obtained as $L\cos(\alpha + \phi_1)$.

($\Rightarrow$) There are two geometric cases to consider: Orientation 0 is a kink and 0 is a local minimum. Let us consider the former case in Fig. 3. The latter case is proved similarly.

In the figure, AA', BB' are faces of P at orientation 0. AC, BD are faces at orientation ϕ_1 and CE, DF are faces at orientation ϕ_2. Some of these faces could be of length 0. Let BX be a line perpendicular to the x-axis. Now $|AB| = L$ and $\angle ABX = \alpha$, the parameters of the sinusoid between $0, \phi_1$.

$|AC| + |BD| = t(\phi_1)$. Extend BD to Q so that $|DQ| = |AC|$. Draw lines parallel to CE (*i.e.* lines making angle ϕ_2 with the positive x-axis) through A, B. Draw a line perpendicular to these lines through Q intersecting them at S, T as shown. Extend BT to point R so that triangle ARB is right angled at R.

Now $ASTR$ is a rectangle and so $|ST| = |AR| = L\cos(\alpha + \phi_2)$. Also, $|TQ| = |BQ|\sin(\phi_2 \Leftrightarrow \phi_1)$. Finally notice that $|BQ| = t(\phi_1)$ and $|QS| = |TQ| + |ST| = d(\phi_2)$. □

Theorem 2 *Two polygons P, Q have the same diameter function if and only if $\Phi_P = \Phi_Q$ and $t_P = t_Q$.*

Remark: From Cauchy's surface formula (See [Benson, 1966]), it follows that the integral of the diameter function of any planar part equals its perimeter. Thus, two parts having the same diameter function must have the same perimeter (but not necessarily vice versa). Our theorem above states something stronger for the class of polygonal parts – namely that the two parts must have the same set of partial perimeters.

Proof: *Only if*: Let P, Q both have diameter function d. Then $\Phi_P = \Phi_Q$ since each is equal to $\Phi(d) \bmod \pi$. Let ϕ_1, ϕ_2, ϕ_3 be an adjacent triplet of orientations[5] in $\Phi(d)$.

Assume WLOG that $\phi_2 \in [0, \pi)$. Then Lemma 2 gives a formula for $t(\phi_2)$ that must be satisfied by both polygons. A generalization of this shows that $t_P = t_Q$.

If: Let the two polygons have (valid) diameter functions d, d'. We prove $d = d'$ by showing that one can reconstruct a *unique* diameter function d, given these two lists Φ_P, t_P.

Let the given Φ_P have $Z = m + k$ orientations: $\Phi_P = \{\phi_0, \ldots, \phi_{Z-1}\}$. Let the diameter function between ϕ_j, ϕ_{j+1} (assume $j \pm 1$ are modulo Z) be the sinusoid $l_j\cos(\phi + \alpha_j)$. Thus we have the equations

$$l_j\cos(\phi_j + \alpha_j) = d(\phi_j), l_j\cos(\phi_{j+1} + \alpha_j) = d(\phi_{j+1}).$$

The unknowns are $d(\phi_j), \alpha_j, l_j$. If these are recovered, it is obvious that d is. This gives us $2Z$ equations in $3Z$ unknowns. The remaining Z equations are obtained from t information using Lemma 2:

$$t(\phi_j) = \frac{d(\phi_{j+1}) \Leftrightarrow l_{j-1}\cos(\phi_{j+1} + \alpha_{j-1})}{\sin(\phi_{j+1} \Leftrightarrow \phi_j)}.$$

[5] Recall from the definition of an gpsf that $\Phi(d)$ must have at least four transition orientations.

Consider orientations ϕ in which an face of P is flush with one of l, h. These are precisely the orientations in $\Phi(d)$. If the orientation is stable under jaw action (replacing l, h by a parallel jaw gripper nad squeezing, see also Section 3.1.1), it is a local minimum in d, and otherwise it is a kink orientation. Since there are n faces, we would expect $n = m + k$. This is true if P did not have any pairs of parallel faces. Let p be the number of pairs of parallel faces of P. Then a simple counting argument gives:

Lemma 1 $n \Leftrightarrow p = m + k$. □

Notice that the quantities on the left side n, p are the polygon's geometrical properties, while the quantities on the right m, k are properties of its diameter function. Let us refer to an n-gon having p pairs of parallel sides as an n, p-polygon. A diameter function having m minima and k kinks is an m, k-diameter function. Thus, it is $n \Leftrightarrow p$, rather than n alone that decides how "complex" the diameter function of the n, p-polygon is. Parallelograms (4,2-polygons) give the "simplest" diameter functions in the sense that they are the only polygons (among convex polygons) that have $n \Leftrightarrow p = 2$. Triangles (3,0-polygons), trapeziums (4,1-polygons), 5,2-pentagons, and 6,3-hexagons are the next simplest having $n \Leftrightarrow p = 3$. All other polygons have $n \Leftrightarrow p > 3$.

Corollary to Lemma 1 If an n_1, p_1-polygon and an n_2, p_2-polygon have the same diameter function, then $n_1 \Leftrightarrow p_1 = n_2 \Leftrightarrow p_2$. □

This is our first step towards shape recovery from diameter function.

2 Negative results

In this section we present our results implying that complete shape recovery of a planar part from its diameter function is impossible. We begin by investigating conditions for a polygon to be consistent with a given diameter function within a range of orientations (Lemma 2). Theorem 2 presents a necessary and sufficient condition for two polygons to have the same diameter function. Theorem 3 shows that there are infinitely many polygons, all satisfying the conditions of Theorem 2, and all having the same diameter function. From the proofs of these negative results for the general class of polygons, it becomes natural to seek shape recovery from diameter function for a special class of polygons: namely those without any pair of parallel faces. We call such polygons *minimal polygons* and consider the shape recovery problem restricted to these polygons in Section 2.1.

Let $t_P(\phi)$, $0 \leq \phi < \pi$, be zero if $\phi \notin \Phi_P$; otherwise it equals the sum of the lengths of all faces[4] of P that have orientation ϕ. $t_P(\phi)$ is called the *perimeter of P at orientation ϕ*. Let $t_P = \{t_P(\phi) | \phi \in \Phi_P\}$ sorted in the order of increasing ϕ. The subscripts P are dropped if we are discussing only one polygon.

A polygon P is said to be *consistent* with a valid diameter function d *between orientations* $[\phi_a, \phi_b]$ if the diameter function of P matches d between orientations $[\phi_a, \phi_b]$. This is written as $P \sim d[\phi_a, \phi_b]$.

Lemma 2 *Let P be a polygon and d some valid diameter function, not necessarily that of P. Further, let $0 < \phi_1 < \phi_2$ be an adjacent triplet of orientations in $\Phi(d)$ and also in Φ_P. Let P be consistent with d at orientations $0, \phi_2$. Then,*

$$P \sim d[0, \phi_2] \Leftrightarrow t_P(\phi_1) = \frac{d(\phi_2) \Leftrightarrow L\cos(\phi_2 + \alpha)}{\sin(\phi_2 \Leftrightarrow \phi_1)},$$

[4] There can be at most two such faces due to the convexity assumption on P.

Notice that a gpsf is continuous, single valued, and has a finite number of local maxima and local minima. A gpsf is differentiable at all but a finite number (at most Z) of orientations in S^1. For a gpsf f, let $MAX(f)$, $MIN(f)$ denote the finite set of local maxima, local minima orientations, respectively. For a gpsf f, let $Z(f)$, the *size* of f, denote the finite integer Z, and $\Phi(f)$, the *set of transition orientations* of f, denote $\{\phi_0, \ldots, \phi_{Z-1}\}$ from the definition of the gpsf f. If $|S|$ refers to the cardinality of a finite set S, then for a gpsf f, $|MAX(f)| = |MIN(f)|$ and $|\Phi(f)| = Z(f)$.

From now on all sets of orientations, such as $\Phi(f)$, $MIN(f)$, $MAX(f)$, and others to be defined later, will be treated as ordered (circular) lists of orientations, *i.e.* their elements will be assumed sorted in a circular list. Two orientations ϕ_a, ϕ_b are said to be *adjacent* with respect to some property if they are adjacent in the (circular) list of all orientations having that property. For example, ϕ_a, ϕ_b are adjacent local maxima in a gpsf f if they are adjacent in $MAX(f)$. From the definitions of local maxima, local minima, notice that if we merge the circular lists $MAX(f), MIN(f)$, we get a new list $MAXMIN(f)$ in which elements of $MAX(f), MIN(f)$ alternate. That is, between every two adjacent local maxima, lies a unique local minima (and vice versa).

Theorem 1 *A function f is a valid diameter function if and only if*

1. *f is a gpsf,*
2. *f has period π, and*
3. $\mathrm{MIN}(f) \subseteq \Phi(f)$, $\mathrm{MAX}(f) \cap \Phi(f) = \emptyset$ *(that is, parameters of the sinusoid change at every local minima and never change at any local maxima).*

Proof Outline: Showing that diameter functions of polygons have these properties is not difficult. For the other direction, assume we are given a function f satisfying these properties. We can construct the left half of a polygon (the right half is a reflection of the left half through the centroid) whose diameter function is exactly f. The idea is the following.

Let 0 and $\phi*$ be two adjacent local maxima in f, and let there exist y orientations $\phi_1, \phi_2, \ldots, \phi_y$ from $\Phi(f)$ between 0 and $\phi*$. The polygon we construct will have y consecutive edges flush with orientations $\phi_1, \phi_2, \ldots, \phi_y$. The lengths of these edges can be determined from the parameters of sinusoids in gpsf f. □

1.3 Further notation and an initial result

Let d denote a valid diameter function and P a polygon. Unless otherwise specified, d is the diameter function of P. Two circular lists of orientations are equal if they are equal after some fixed orientation (possibly zero) is added every element of one of them. From now on, maxima, minima stand for local maxima, local minima (in a diameter function), respectively.

Orientations in $k(d) = \Phi(d) \Leftrightarrow MIN(d)$ are called *kink orientations*, or more simply *kinks*. That is, kinks are the non-minima orientations at which the parameters of the sinusoid describing d change. Kinks and minima, *i.e.* orientations in $\Phi(d) = MIN(d) \cup k(d)$, are all and the only orientations at which an face of the polygon P is in contact with (at least) one of l, h. For example, an obtuse angled triangle has only one minima between $[0, \pi))$ when the largest side is in contact with one of l, h. When one of the other two sides is flush with the lines l, h, we get a kink orientation. Let $MINMAXKINK(d)$ denote a list of all maxima, minima, kinks in the diameter function d.

Let Φ_P denote the set of angles (module π) that the edges of polygon P make with the x-axis. Φ_P is $\Phi(d)$ restricted to the range $[0, \pi)$. Let m, k, respectively denote the number of minima, kinks in $[0, \pi)$, in the diameter function d of an n-gon P.

with object but also the direction of normal there. Our negative results (Section 2) therefore imply that for shape recovery, both the length of the projection, and the offset information (namely where the projection lies with respect to some fixed point on the sliding line) are necessary, and merely the length of the projection (diameter information) is not sufficient.

Now we refer to literature related to our work on grasp strategies for recognizing polygons. Wallack and Canny consider shape and pose recognition by scanning light beam projections that are orthogonal to the plane of the part [Wallack and Canny, 1991]. Only a few high resolution scan lines are used and the pose algorithm runs in $O(n)$ time (on the average) for an n-sided polygonal part. In [Wallack and Canny, 1993] they consider light beams that are in the plane of the part which can be used to measure diameter without affecting part orientation. In contrast, grasping actions with a parallel jaw gripper can be used to measure diameter *after rotating the part into a stable orientation* prior to the measurement. They also consider the problem of determining part orientation by planning an sequence of diameter measurements [Wallack and Canny, 1992]. Our's is the more general problem of recognizing a part from a known set. Also, we explore the computational complexity of planning an optimal sequence.

Our work on recognizing part shape also has some relation with robotic inspection [Spyridi and Requicha, 1990] and the work of Ellis [Ellis, 1987] concerned with tactile data for recognition. Chen and Ierardi [Chen and Ierardi, 1991] consider the problem of recognizing a set of polygonal parts from diameter measurements. However, they do not consider optimal plans. Kang and Goldberg [Kang and Goldberg, 1992] consider a similar problem where the diameter measurements are corrupted by Gaussian noise. They show that data from a sequence of random grasps can be processed using a Bayesian estimator. However random grasping has low average-case performance when part shapes are similar. The relation between planning manipulation strategies and searches through trees was observed in [Erdmann and Mason, 1986; Mason *et al.*, 1988; Taylor *et al.*, 1987]. The notion of *active vision*, servoing sensors so as to obtain information efficiently is popular in vision recognition strategies [Raviv, 1991]. Ours is also an active sensing strategy in that we seek a best next grasp for the part based on the previous ones.

Finally, our deployment of the parallel-jaw gripper and simple diameter sensing is in tune with Canny's and Goldberg's Reduced Intricacy in Sensing and Control, or RISC, robotics [Canny and Goldberg, 1993]. They observe that for certain industrial tasks, complex (high degree of freedom) manipulators and sensors are unnecessary and that simple, robust, and inexpensive hardware is called for. In this paper, we demonstrate a RISC approach to shape recognition.

1.2 Valid diameter functions

We call a diameter function *valid* if it is the diameter function of some polygon. Since the concept of a valid diameter function is so central to this paper, we now set about trying to characterize valid diameter functions.

A polygon P is specified by its n vertices in counter-clockwise order $v_0, v_1, \ldots, v_{n-1}$. Let $\oplus$ denote addition modulo n. The orientation of an face of P, $v_i, v_{i\oplus 1}$ is the angle made by it $\mathrm{mod}\pi$ *w.r.t* the positive x-axis. Let ϕ denotes an arbitrary orientation in S^1.

A continuous function $f : S^1 \rightarrow \mathcal{R}_+$ is said to be a "good" piecewise sinusoidal function (gpsf) if there exists a finite integer $Z \geq 4$, and a cyclic ordering of orientations

$$\phi_0 < \phi_1 < \ldots < \phi_{Z-1} < \phi_Z = \phi_0$$

such that $\forall j \in \{0, 1, \ldots, Z \Leftrightarrow 1\}$, and $\forall \phi \in [\phi_j, \phi_{j+1}] : f(\phi) = l_j \cos(\phi + \alpha_j)$, for some $l_j \in \mathcal{R}_+$, and $\alpha_j \in S^1$.

except that it is a polygon. Section 3 assumes that the given part has one among a known set of polygonal cross sections. We also assume a frictionless parallel jaw gripper [Goldberg, 1990] equipped with a linear position sensor for taking diameter measurements. We briefly discuss mechanics of the parallel jaw gripper grasping a polygonal object [Goldberg, 1990]. We only consider *stable* diameter measurements (*i.e.* on grasping, the part settles into a stable equilibrium orientation before the diameter at that orientation is measured). The aim is to compute a "grasp strategy", a sequence of stable diameter measurements, that would distinguish between the parts up to part distinguishability. We give algorithms that consider tradeoffs between preprocessing time and the length of the grasp strategy produced.

These results have been previously reported in technical report form [Rao and Goldberg, 1992c]. In interest of space and to increase readability, we have replaced proofs of some of the lemmas and theorems in this paper by proof outlines. The interested reader is referred to the technical report for complete proofs.

1.1 Related work

The concept of *diameter* of a set of points, the maximizing distance over all pairs of points, is well studied in computational geometry [Preparata and Shamos, 1985; Edelsbrunner, 1987]. Diameter functions, termed width functions in [Yaglom and Boltyanskii, 1951], were applied by Jameson [Jameson, 1985] to determine grasp stability for a part grasped in the jaws of a parallel-jaw gripper. Goldberg [Goldberg, 1993] used the diameter function to generate plans, in $O(n^2)$ time, to orient n-gonal parts. Rao and Goldberg [Rao and Goldberg, 1992b; Rao, 1992] extend these results to curved parts.

Our work has some relation to geometric probing which was introduced by Cole and Yap [Cole and Yap, 1987] and inspired by work in robotics and tactile sensing [Grimson and Lozano-Perez, 1987; Gaston and Lozano-Perez, 1984]. A complete description of these results appear in [Skiena, 1988]. Skiena [Skiena, 1989] provides a summary of the basic results and open problems in this area. However, some of the basic assumptions common in geometric probing are quite unrealistic in practice. Some examples of such assumptions are that the part is not disturbed by the probes, the probes are accurate (travel in straight lines and return error-free information), and probes are powerful (can recover information like normals from the contact face, *etc.*). Some of the more relevant papers in probing are discussed briefly.

Koutsou [Koutsou, 1988] investigates tactile exploration of objects using a parallel jaw gripper. He considers how a parallel-jaw gripper equipped with a tactile array can be used to measure physical properties of a part such as weight and stiffness. [Kolzow *et al.*, 1989] discusses an algorithm for the approximative reconstruction of a planar convex body from its projections in a finite number of directions. However, by "projections" of a body in a particular direction ϕ, they understand the length of the intersection of a line (with orientation orthogonal to ϕ) with the body as a function of the distance of the line from the origin. Thus, their idea of "projections" is quite different from "projection probes." *Projection probes* [Li, 1988] are a special type of *hyperplane probes* [Dobkin *et al.*, 1986]. A projection probe slides a straight line in a direction normal to itself over the plane until it hits the polygon. When it does, it returns the projection (or image) of the polygon on the line. From $3n \Leftrightarrow 2$ projection probings, [Li, 1988] shows that you can recover the complete shape of the polygon. If the sliding straight line makes an angle ϕ with the x-axis, then the *length* of the projection is the diameter at orientation $\phi \pm \pi/2$. Boissonat and Yvinec [Boissonnat and Yvinec, 1992] consider probing of non-convex objects – both polygons and polyhedra. They give algorithms that exactly recover shape using powerful projection probes that return not only point of contact

parts, for example a circle cannot be distinguished from a Reuleaux triangle (cf. the Wankel rotary engine) merely by grasping and measuring its diameter.

First, we present the negative result that shape cannot be uniquely recovered even for polygonal parts: for a given set of diameter measurements: there is an (uncountably) infinite set of polygonal shapes consistent with these measurements. Since most of the shapes in this set have parallel edges of varying lengths, we consider the related problem of identifying a representative polygon with no parallel edges. We show that given a diameter function, deciding whether such a polygon exists is *NP*-Complete.

These results motivate us to consider the problem of *recognizing* a part from a known (finite) set of parts. The challenge is to plan a sequence of grasp angles to efficiently recognize the part, taking into account the induced rotation of the part caused by the gripper. In other words, given a set of polygonal parts with a total of N faces, find the shortest sensing plan for disambiguating the parts using a parallel-jaw gripper. See Fig. 1 for an example.

Since diameters at only stable orientations can be measured, we build a graph representation, G, of the $n \leq N$ stable diameters in $O(N+n^3)$ time. Our positive results are two planning algorithms: one constructs an optimal sensing plan in time $O(n^4 2^n)$, the other constructs a suboptimal sensing plan in time $O(n^2 \log n)$. Neither plan requires more than n on-line measurements.

Our solution is an example of *active sensing* [Bajcsy, 1988], where the sensor is purposefully moved to acquire appropriate data. More specifically, it is an example of *sensor-based manipulation planning* [Taylor *et al.*, 1987], where the part is moved in the process of acquiring sensor data. Thus the planner requires a model of mechanics to predict part motion. In [Taylor *et al.*, 1987], a brute-force search was used to find optimal plans for the related problem of determining part orientation with a standard parallel-jaw gripper. Here, we consider a *frictionless* parallel-jaw gripper, modified with a linear bearing to insure a finite range of possible diameters [Goldberg and Furst, 1992]. We analyze the computational complexity of finding optimal and suboptimal plans for recognizing a part from a known set.

Let S^1 denote the space of planar orientations $[0, 2\pi)$ and $\mathcal{R}_+$ the set of positive reals. Given a fixed part P in an $x \Leftrightarrow y$ coordinate frame, the diameter function $d : S^1 \rightarrow \mathcal{R}_+$ of P can be formally defined as follows. Imagine two (infinite) parallel lines l, h (supporting lines) both making angle ϕ with the x-axis, just touching P so that P lies entirely in the region between the two lines. In such a case we say that the supporting lines l, h are at *orientation* ϕ with respect to the (fixed) polygon P. The diameter of P at orientation ϕ, $d(\phi)$, is the distance between the two lines that are at orientation ϕ. The diameter function is continuous and has period π. See Fig. 2. Also, the diameter function of a part is the diameter function of its convex hull. Therefore we can only seek shape recovery of the *convex hull of a part* from its diameter function [Goldberg, 1990].

Call a diameter function *valid* if it is the diameter function of a polygon. We begin, in Section 1.2 by characterizing valid diameter functions. In Section 2 we show that for every valid diameter function d there exist infinitely many polygons consistent with it. Thus, complete shape recovery of a polygon from its diameter function is impossible. However, we show that the orientation of every edge of the polygon and partial perimeters of the polygon along any orientation are recoverable from the diameter function. Looking at the proofs of these results, it becomes natural to consider shape recovery for the restricted class of polygons having no parallel edges (we call such polygons *Minimal polygons*). In section 2.1 we show that although the number of minimal polygons consistent with a given diameter function is finite, complete shape recovery still impossible. Also we show that deciding the question, "Given valid diameter function d, is there a minimal polygon consistent with d?" to be *NP*-complete.

These negative results for shape recovery assume that no *a priori* knowledge about the part

Shape from Diameter: Recognizing Polygonal Parts with a Parallel-Jaw Gripper*

Anil S. Rao† and Kenneth Y. Goldberg‡

November 5, 1993

Abstract

Our objective is to automatically recognize parts in a structured environment (such as a factory) using inexpensive and widely-available hardware. In this paper we consider the planar problem of determining the convex shape of a polygonal part from a sequence of projections. Projecting the part onto an axis in the plane of the part produces a scalar measure, the *diameter*, which is a function of the angle of projection. The diameter of a part at a particular angle can be measured using an instrumented parallel-jaw gripper.

First, we present the negative result that shape cannot be uniquely recovered: for a given set of diameter measurements, there is an (uncountably) infinite set of polygonal shapes consistent with these measurements. Since most of these shapes have parallel edges of varying lengths, we consider the related problem of identifying a representative polygon with no parallel edges. We show that given a diameter function, deciding whether such a polygon exists is *NP*-Complete.

These results motivate us to consider the problem of *recognizing* a part from a known (finite) set of parts. Given a set of polygonal parts with a total of N faces, can we find the shortest sensing plan for disambiguating the parts? Only diameters at $n \leq N$ stable faces can be measured. We construct an internal representation of these stable diameters in $O(N + n^3)$ time and then give two planning algorithms: one constructs an optimal sensing plan in time $O(n^4 2^n)$, the other constructs a suboptimal sensing plan in time $O(n^2 \log n)$. Neither plan requires more than n measurements.

1 Introduction

In automated manufacturing it is often useful to sort parts according to shape. A common approach is to use machine vision, which can be sensitive to lighting conditions and requires coordination with a programmable manipulator. In this paper we explore an alternative approach that uses an inexpensive modification to the parallel-jaw gripper. For the class of convex parts with constant polygonal cross section (2.5 D parts), we consider the following problem: recover the shape of a part's cross section by grasping the part with a parallel-jaw gripper and measuring the distance between the jaws: the *diameter* of the part at some angle. Can a sequence of such measurements be used to determine part shape? Note that the answer is clearly negative if we admit curved

*This material is based upon work supported by the National Science Foundation under Award No. IRI-9123747, ESPRIT Basic Research Action No. 6546 (project PROMotion), and by an equipment grant from Adept Technology, Inc.

†Department of Computer Science, Utrecht University, Padualaan 14, Postbus 80.089, 3508 TB Utrecht, Netherlands. anil@cs.ruu.nl, Tel: 31-30-533922, FAX: 31-30-513791.

‡Department of Computer Science, Institute of Robotics and Intelligent Systems, PHE 204, University of Southern California, Los Angeles, California 90089-0273. goldberg@iris.usc.edu, Tel: 1-213-740-9080, FAX: 1-213-740-7877.