Three Finger Optimal Planar Grasp

Bud Mishra, Marek Teichmann

Abstract— **In this paper, we study various algorithmic questions regarding the computation of an optimal three finger planar grasp. We present a novel $O(n^2 \log n)$-time algorithm to compute such an optimal grasp for an arbitrary simple n-gon. This algorithm can be used for finding "good" immobilizing sets. We also discuss several variations on the problem and many intriguing open questions in the area that remain unsolved.**

Keywords— **Dexterous Manipulation, Grasping, Fixturing, Immobility, Grasp Metrics, Algorithms.**

I. Introduction

In robotics and manufacturing, the problem of grasping, fixturing and workholding occupies a central position. While the question of analyzing and synthesizing *a* closure grasp or *a* fixture is fairly well-studied (see for instance [1], [8], [9], [10], [12], [13], [15], [20], [22]), the question of formalizing a satisfactory notion of a grasp metric, and devising efficient algorithm for synthesizing grasps of *good quality* has received relatively less attention. A systematic exploration in this direction was initiated in the work of Kirkpatrick, Mishra and Yap [5], where a deep interconnection between the problem of optimal grasp synthesis and certain quantitative versions of Helly-type theorems in combinatorial geometry (namely, quantitative Steinitz's theorem) plays a key role. However, the resulting algorithmic questions in the most general setting appear to be intractable. For a through discussion of these issues, consult [11], [20].

In order to better understand the underlying structure as well as to provide practical solutions in the simpler settings (as more common in manufacturing), we have directed our attention to the cases where we study lower-dimensional objects (2-D or $2\frac{1}{2}$-D) and of simpler geometry (polygonal objects) or simpler robot hands. In this paper, we explore this problem for two-dimensional polygonal objects with hands of relatively few fingers. Other works in a similar spirit include those of Ferrari and Canny [4], Li and Sastry [7], Brost and Goldberg [1], Teichmann [20] and Trinkle [21].

In this paper, we study various algorithmic questions regarding the computation of an optimal three finger planar grasp. We present a novel $O(n^2 \log n)$-time algorithm to compute such an optimal grasp for an arbitrary simple n-gon. We also discuss several variations on the problem and many intriguing open questions in the area that remain unsolved.

Supported by NSF Grant # IRI-94-14862.

Robotics Research Laboratory, Courant Institute, New York University, 251 Mercer Street, New York, N.Y.10012. E-mail: mishra@nyu.edu, teichman@cs.nyu.edu .

II. Preliminary

We wish to obtain the best three-finger grasp of a planar polygonal object assuming non-frictional contacts. Our algorithm is based on an idealized robot hand model. It consists of several independently movable force-sensing stiff fingers and grasps a rigid (planar) object K. The fingers are placed at points $\mathbf{p}$ of the boundary of K, which we shall denote by ∂K. Additionally, we make the following assumptions: (1) K is a full-bodied (i.e. no internal holes) compact subset of the Euclidean 2-space, and has a piecewise smooth boundary ∂K. (2) For each finger-contact on the body, we may associate a nominal point of contact, $\mathbf{p} \in \partial K$. We let ∂K^* denote the set of points $\mathbf{p} \in \partial K$ such that the direction $\mathbf{n}(\mathbf{p})$ normal to ∂K at $\mathbf{p}$ is well-defined; by convention, we pick $\mathbf{n}(\mathbf{p})$ to be the unit normal pointing into the interior of K.

In the case, where the contacts are frictionless, we call the corresponding grips 'positive grips.' The wrench system associated with each point is:

$$\Gamma(\mathbf{p}) = \{[\mathbf{n}(\mathbf{p}), \mathbf{p} \times \mathbf{n}(\mathbf{p})]\},$$

where the first part $\mathbf{n}(\mathbf{p})$ corresponds to the force component and the last part, the torque component. More general wrench systems can be used to reflect more complicated contact models (friction, soft contacts, etc.). Next we examine the concept of a *closure grasp*:

Definition 1: A set of gripping points on an object K to which corresponds a system of wrenches $\mathbf{w}_1, \ldots, \mathbf{w}_n$ is said to constitute a *force/torque closure grasp* (or simply, a *closure grasp*) if and only if any arbitrary external wrench can be generated by varying only the magnitudes of the wrenches, subject to the constraint that the magnitudes take nonnegative values.

A necessary and sufficient condition for a closure grasp in case of positive grips is

$$\mathbf{0} \in \text{int conv } (\mathbf{w}_1, \ldots, \mathbf{w}_n).$$

Another equivalent formulation would be to require that the wrenches $\mathbf{w}_1, \ldots, \mathbf{w}_n$ positively span the wrench space. It then follows that one requires four or more non-frictional grip points to establish a closure grasp on a planar object.

A simple linear time algorithm for finding *at least* one closure grasp for a polygonal object has been presented in [12]. However, it remains unclear whether it is possible to efficiently compute the "best" possible closure grasp for a fixed number (say, four or more) of fingers. For some related discussion of such questions, see Teichmann's thesis [20].

In this paper, we examine the grasping strategies for hands with three fingers. Note that in this case, since it is not possible to guarantee that the resulting grasp will have the force/torque closure properties, we are willing to sacrifice the condition requiring torque-closure. In other words, we wish only to achieve a three-finger grasp maximizing the smallest external force such a grasp can resist. If we ignore the torque component, then the condition for the force closure is given by

$$\mathbf{0} \in \text{rel int conv } (\Gamma(\mathbf{p}_1), \Gamma(\mathbf{p}_2), \Gamma(\mathbf{p}_3)).$$

More formally given a simple n-gon P, we wish to choose three distinct points $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$ on the interior of the edge segments of P such that the following properties hold:

1. The unit inner normals $\mathbf{n}(\mathbf{p}_1)$, $\mathbf{n}(\mathbf{p}_2)$ and $\mathbf{n}(\mathbf{p}_3)$ are concurrent. Assume that the object K is described with respect to a coordinate system whose origin is at this point of concurrency. In this case, note that
$$\Gamma(\mathbf{p}_i) = \mathbf{n}(\mathbf{p}_i), \quad i = 1, 2, 3.$$
2. The unit inner normals $\mathbf{n}(\mathbf{p}_1)$, $\mathbf{n}(\mathbf{p}_2)$ and $\mathbf{n}(\mathbf{p}_3)$ positively span the two-dimensional force space, i.e.,
$$(\forall \mathbf{w} \in \mathbb{R}^2)\ (\exists f_i \geq 0, 1 \leq i \leq 3)\ \mathbf{w} = \sum_{i=1}^{3} f_i \mathbf{n}(\mathbf{p}_i).$$
This condition follows from our discussion of closure grasps and the choice of the preceding coordinate system.
3. The unit normals are "well-balanced" in the sense that
$$\min\Big\{ |\mathbf{w}| : \mathbf{w} \in \mathbb{R}^2, \\ (\exists f_i \geq 0, 1 \leq i \leq 3)\ \chi(f_1, f_2, f_3) = 1 \\ \mathbf{w} = \sum_{i=1}^{3} f_i \mathbf{n}(\mathbf{p}_i) \Big\},$$
is as large as possible (among all choices of $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$). Here, $\chi(f_1, f_2, f_3)$ denotes a finger force constraint condition on the magnitude of the forces applied at the points of contact. For instance,
$$\chi_{con} : f_i \geq 0 \text{ and } \sum f_i \leq 1,$$
or
$$\chi_{max} : f_i \geq 0 \text{ and } \max f_i \leq 1.$$

Thus the first property denotes the trivial torque equilibrium condition; the second property denotes the force closure condition and the third property measures the goodness of the grasp. In English, the third property says: under the condition χ_{con}, we wish to maximize the radius of a disk, centered at origin and contained in the triangle formed by (convex hull of) the points (on the unit circle) corresponding to the vectors $\mathbf{n}(\mathbf{p}_1)$, $\mathbf{n}(\mathbf{p}_2)$ and $\mathbf{n}(\mathbf{p}_3)$. Similarly, under the condition χ_{max}, we wish to maximize the radius of a disk, centered at origin and contained in the Minkowski sum of the points (on the unit circle) corresponding to the vectors $\mathbf{n}(\mathbf{p}_1)$, $\mathbf{n}(\mathbf{p}_2)$ and $\mathbf{n}(\mathbf{p}_3)$—a convex hexagon.

Let the corresponding radii be denoted as $\rho_{con}(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ and $\rho_{max}(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$, respectively. Note that, if the

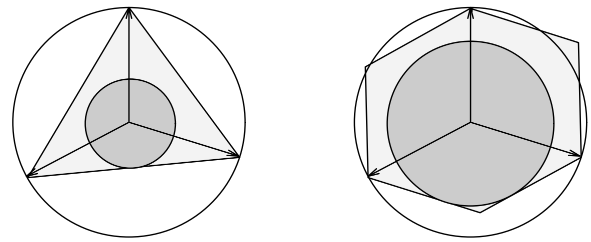

Fig. 1. Grasp metrics associated with χ_{con} and χ_{max}.

angle α_i's $(1 \leq i \leq 3)$ denote the angles between the inner normals then $\alpha_{max} = \max(\alpha_1, \alpha_2, \alpha_3) \geq 2\pi/3$ completely determines the radii

$$\rho_{con} = \cos(\alpha_{max}/2), \quad \text{and}$$
$$\rho_{max} = \sin \alpha_{max}.$$

Thus both these metrics are monotonically decreasing functions of $2\pi/3 \leq \alpha_{max} \leq \pi$, and it suffices to minimize α_{max}. However, for the sake of the ease of exposition, we will frequently use $\rho = \rho_{con}$, and refer to it as the "*residual radius*" of $\mathbf{n}(\mathbf{p}_1)$, $\mathbf{n}(\mathbf{p}_2)$ and $\mathbf{n}(\mathbf{p}_3)$. The optimal value of residual radius is denoted by ρ^*.

III. Immobility and Closure Grasps

We wish to note at this point that the placements for the three fingers as described in the earlier section also give rise to immobilizing sets. The notion of *immobility* is much more a geometric notion and is defined in terms of the lack of any freedom for finite movement of the immobilized object. (See Kuperberg [6]). An interesting survey of several open problems pertaining to this subject can be found in two recent computational geometry columns of O'Rourke (see [16], [17]).

Let K be a connected compact closed region in the plane $\mathbb{R}^2$; we say that K is (finitely) *immobilized* by a set of points $I \subseteq \mathbb{R}^2$ if any rigid motion of K in the plane causes at least one point of I to penetrate the interior of K. Since we could assume that I is a minimal set possessing such a property, we shall only consider, without loss of generality, the case where I belongs to the boundary of K $(I \subseteq \partial K)$. A planar object that can be immobilized by a set of no more than k points will be referred to as *k-immobilizable*. Thus, we are interested in computing the "best" immobilizing set for a 3-immobilizable simple polygon.

First, we claim that with the positions of the finger as described previously, we can immobilize the polygon. Our claim is a consequence of the following theorem of Czyzowicz, Stojmenovic and Urrutia [2].

Theorem 1: Three points immobilize a triangle if and only if the normals to the triangle at these points are concurrent.

The grasps as described satisfy the required concurrency condition, and the extension of the edges at which grasp points are placed form a triangle since their normals are positively spanning, again by assumption.

Furthermore, if any non zero coefficient of friction is allowed, these grasps are also good three finger closure grasps. This can be seen as follows. Let μ be this coefficient of friction, and assume μ is sufficiently small. Assume also that the origin lies at the concurrency point of the three forces. Let Γ be the "wrench map" mapping a unit force along the inward-directed normal at $\mathbf{p}_i$ to the corresponding force and torque in the 3-dimensional "wrench space." (See Mishra et al. [12].) Then $\Gamma(\mathbf{p}_i)$ lie on the xy plane in the wrench space $\mathbb{R}^3$. For a unit vector $\mathbf{n}$ in the xy plane in $\mathbb{R}^3$, let $\mathbf{n}^\perp$ be the unit vector perpendicular to $\mathbf{n}$ and rotated by $\pi/2$ counter-clockwise. Let also $\mathbf{n}_i = \mathbf{n}(\mathbf{p}_i)$. The two rays bounding the friction cone at $\mathbf{p}_i$ are $\sqrt{1-\mu^2}\mathbf{n}_i \pm \mu\mathbf{n}_i^\perp$, and their image by Γ is

$$\left[\sqrt{1-\mu^2}\mathbf{n}_i \pm \mu\mathbf{n}_i^\perp, \pm|\mathbf{p}_i|\frac{\mu}{\sqrt{1+\mu}}\right].$$

Letting α be the half angle of the friction cone, the last term follows from the formula of the cross product which involves the sine function and the fact that $\mu = \tan\alpha = \sin\alpha/\cos\alpha$.

Hence the top facet of $\mathrm{Conv}\{\Gamma_\mu(\mathbf{p}_i) : i = 1, 2, 3\}$ will be farther from the xy plane (and from the origin) by a distance of at least $\min_i\{|\mathbf{p}_i|\frac{\mu}{\sqrt{1+\mu}}\}$. Similarly for the bottom facet. Thus when μ is sufficiently small so that no other point of $\mathrm{Conv}\{\Gamma_\mu(\mathbf{p}_i) : i = 1, 2, 3\}$ is closer to the origin, this quantity will be a lower bound for the quality of the grasp.

IV. A Cubic Time Algorithm

Clearly, there is a trivial $O(n^3)$ time algorithm to find an optimal grasp of a simple n-gon, P, by exhaustively enumerating all edge triples of P and by examining each triple successively. Given an edge $e = ab$ of the polygon P, for every point $\mathbf{p} \in ab$, $\mathbf{n}(\mathbf{p})$ defines a unique point $q(e)$ on the unit circle in $\mathbb{R}^2$. Thus we may simply refer to this point on the unit circle by $q(e)$. Henceforth, let the edges of the n-gon be given as $E = \{e_1, e_2, \ldots, e_n\}$ and the corresponding points on the unit circle be $Q = \{q_1, q_2, \ldots, q_n\}$, where $q_i = q(e_i)$ $(1 \le i \le n)$. In order for an edge triple (e_i, e_j, e_k) to produce three necessary optimal contact points, it must be the case that (q_i, q_j, q_k) form a triangle with a positive residual radius of ρ^*—a condition that can be checked easily in $O(1)$ time. However, this is not sufficient—since we must check that there are three points $\mathbf{p}_i \in e_i$, $\mathbf{p}_j \in e_j$ and $\mathbf{p}_k \in e_k$ satisfying the torque equilibrium condition; namely, that $\mathbf{n}(\mathbf{p}_1)$, $\mathbf{n}(\mathbf{p}_2)$ and $\mathbf{n}(\mathbf{p}_3)$ are concurrent meeting at some point c.

This is not hard but requires some thought. We proceed as follows: Consider an edge ab of P. Let $HP(a, ab)$ be the open half plane containing ab and delimited by a line containing a and normal to ab and similarly, let $HP(b, ab)$ be the open half plane containing ab and delimited by a line containing b and normal to ab. Let

$$\mathrm{slab}(e) = HP(a, ab) \cap HP(b, ab),$$

where $e = ab$.

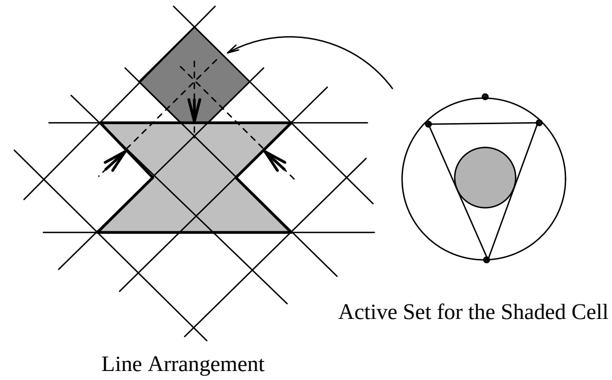

Fig. 2. The line arrangement associated with an object.

Then it is easy to see that for a triple of edges (e_i, e_j, e_k) to satisfy the torque equilibrium condition, it is necessary and sufficient that

$$\mathrm{slab}(e_i) \cap \mathrm{slab}(e_j) \cap \mathrm{slab}(e_k) = C \neq \emptyset.$$

The point of concurrency $c \in S$, and the contact points $\mathbf{p}_i$, $\mathbf{p}_j$ and $\mathbf{p}_k$ are determined by the normals from c onto the edges e_i, e_j and e_k.

Thus our previous arguments can be summarized to be saying that an edge triple (e_i, e_j, e_k) defines an optimal grasp if $\mathrm{slab}(e_i) \cap \mathrm{slab}(e_j) \cap \mathrm{slab}(e_k)$ is nonempty and that the triangle formed by the corresponding points on the unit circle has a positive residual radius of ρ^*, maximal among all choices of edge triples. These considerations yield an $O(n^3)$-time algorithm.

V. An Improved Subcubic Algorithm

Next, we ask if it is possible to improve upon the trivial $O(n^3)$-time algorithm. Here, we present an $O(n^2 \log n)$-time algorithm for finding the optimal three fingered planar grasp for an arbitrary simple polygon.

We first describe the algorithm assuming that the polygon P is nondegenerate (in the sense that will be made precise later) and then remark on how the nondegeneracy can be eliminated by a simple modification to the algorithm.

The algorithm can be described as follows: First we create the two-dimensional line arrangement formed by a collection of lines consisting of three lines per edge, where the triplet of lines associated with an edge ab are: (1) the line containing the edge ab, (2) the line normal to ab, containing a and (3) the line normal to ab, containing b. Now consider a nonempty cell C of this arrangement: we say a point $q = q(e)$ on the unit circle is *active* for this cell, if $\mathrm{slab}(e) \supseteq C$. The subset of points on the unit circle (among the points $q_1, q_2, \ldots, q_n$ of Q) that are active for this cell C, is called its active set and denoted by $\mathrm{active}(C) \subseteq Q$. Now, if we find three points q_i, q_j and $q_k \in \mathrm{active}(C)$, whose residual radius $\rho(C)$ is as large as possible (and positive), then it is seen that ρ^* is simply the maximum of all $\rho(C)$'s taken over all cells of the arrangement.

Note that there are at most $O(n^2)$ cells altogether and as we go from one cell C to its adjacent cell C' then the $\mathrm{active}(C')$ can be computed from the $\mathrm{active}(C)$ by adding or deleting a point on the unit circle, depending on the line containing the $C \cap C'$. Of course, here we have tacitly assumed that the polygon is nondegenerate, in the sense that all the lines on the arrangement are distinct, since otherwise $C \cap C'$ may belong to more than one line of the arrangement and thus require addition and deletion of more than one point of the set Q. Clearly, the active sets for all the cells can be computed in $O(n^2)$ time by visiting the cells of the arrangement, starting from a cell with an empty active set (such a cell exists sufficiently far away from the polygon P). However, computing the $\rho(C)$ for each cell may still take $O(n)$ time, thus forcing the entire procedure to take $O(n^3)$ time.

We circumvent this problem by the following simple trick: First of all we maintain the elements of each active(C) in a clockwise order in a dynamic balanced binary search tree. Since each update operation on this data structure takes $O(\log n)$ time, this increases the complexity of computing the active sets of all the cells to $O(n^2 \log n)$-time.

At any instant, we only remember $\tilde{\rho}$—the maximal residual radius seen so far. That is, $\tilde{\rho}$ is simply the maximum of those $\rho(C)$'s corresponding to only those cells C that have been visited so far. We also remember the edge triple associated with the radius value $\tilde{\rho}$. When we go from a visited cell C to an adjacent unvisited cell C', we do one of two things: If going to the next cell entails deletion of a point, q_i, on the unit circle, then we only have to update the active(C'); the maximal residual radius of C' cannot be larger than that of C and thus $\tilde{\rho}$ remains unchanged. If going to the next cell, on the other hand, entails addition of a point, q_i on the unit circle, then we have to both update the active(C') and check if $\tilde{\rho}$ can be improved. If the maximal residual radius of C', $\rho(C') > \tilde{\rho}$, then the associated triplet from active(C') must involve the new point q_i and two of the old points. How can we do this operation quickly?

First note that residual radii cannot take all possible values but only one of $\binom{n}{2}$ values, each value being determined by a pair of distinct points q_l and q_m and is equal to the radius of the circle that is centered at the origin and has the line containing q_l and q_m as tangent. All these radii can be sorted in $O(n^2 \log n)$ time and are denoted by

$$0 \leq \rho_1 \leq \rho_2 \leq \cdots \leq \rho_i \leq \cdots < 1$$

Suppose before visiting the cell C' the maximal residual radius seen so far is $\tilde{\rho} = \rho_i$. When we go to the cell C' (which requires adding the point q_i), we will successively test if it has a residual radius no smaller than ρ_{i+1}, ρ_{i+2}, etc. until we fail for some value ρ_j $(j > i)$. Each such test can be performed in $O(\log n)$ time as explained below.

Let $i < k \leq j$, and we wish to test if active(C') has three points involving q_i and of residual radius $\geq \rho_k$. Consider a circle $C(\rho_k)$ of radius ρ_k and centered at the origin. Two distinct points of active(C') are said to be *mutually visible* if the line segment connecting these two points do not intersect the interior of $C(\rho_k)$. Thus our test succeeds if we can find a pair of mutually visible distinct points among the active(C'), each of which is also mutually visible with q_i. Let the *leftmost partner* of q_i be the *last* mutually visible point of q_i encountered, visiting the points of active(C') in clockwise order starting from q_i. We call this point $LP(q_i)$. Similarly, we define the the *rightmost partner* of q_i by visiting the points of active(C') in anti-clockwise order, and call it $RP(q_i)$. Since the active points of C' are kept in their sorted order in a balanced search structure, both $LP(q_i)$ and $RP(q_i)$ can be computed in $O(\log n)$ time. Then it only remains to check that $LP(q_i)$ and $RP(q_i)$ are mutually visible, a step that can be accomplished in $O(1)$ time.

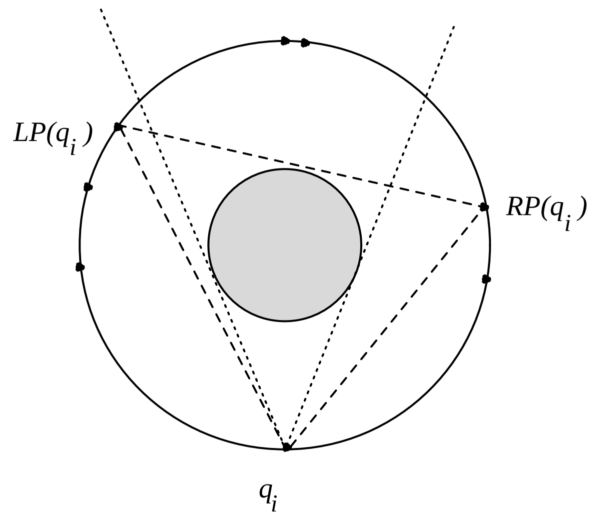

Fig. 3. Test involving q_i and a possible residual radius value of ρ_k.

Thus, we can keep track of $\tilde{\rho}$ by performing a sequence of tests per each new cell, each of which takes $O(\log n)$ time. Note that while there is no a priori bound on the number of tests we may need to perform for a new cell, it should be obvious that all but the last test succeeds and the last test fails. Thus there are at most one test per cell that fails, and the totality of all such failed tests incur a cost of $O(n^2 \log n)$. On the other hand, if we have a successful test involving a radius value ρ_k, then *we shall never perform another successful test involving ρ_k, subsequently.* Thus, the total number of successful tests are bounded by the number possible radii values ($\binom{n}{2}$ of those) and altogether they incur a cost of $O(n^2 \log n)$. Clearly, when we are done visiting all the cells, we have the global maximal residual radius ρ^* together with the edge triple, which readily give the three contact points, and we have spent $O(n^2 \log n)$ time.

If the polygon P is degenerate then the resulting arrangement may force us to add and delete many points of Q while going from a cell to its adjacent cell. If we enforce the discipline that all the deletions are performed before all the additions and each update is performed sequentially then the correctness of the algorithm still holds and the performance analysis goes through *mutatis mutandis*. In summary, we have

Theorem 2: Given an arbitrary simple n-gon P, we can compute a three finger optimal grasp of P in $O(n^2 \log n)$ time.

Proof: The proof is a simple consequence of the preceding discussion. ■

VI. Improvements and Open Questions

There are several open questions related to the problem of finding optimal planar grasps. We briefly discuss these problems.

A. Polygon with Forbidden Regions

Consider a variation on the above problem: Suppose we are given a simple polygon P with certain subset of ∂P designated as "forbidden" and its complement, "feasible." Assume that the feasible parts of the polygon consists of at most K segments (the edge segment ab being allowed to be a point a ($a = b$), in the degenerate case). We are asked to find an optimal three-finger grasp of the polygon with none of the fingers on a forbidden region. Using a small variation of the above algorithm, we can solve this problem in $O(k^2 \log k)$ time—only modify the line arrangement to consist of the following triple of lines per feasible edge segment $ab \subset e$, where e is an edge of P: (1) the line containing e, (2) the line normal to e and containing a, and (3) the line normal to e and containing b. If the edge segment is a point $a \in e$ then the above situation degenerates to two lines, one containing e and the other normal e at a.

B. Convex Polygon with Geometric Constraints

We do not know whether there is a better solution for the above problem with improved complexity. For instance, it is not even clear whether there are $O(n)$ time algorithms for objects with simpler geometry, e.g., convex objects. We have an $O(n)$-time solution only for what we shall refer to as *circular convex* polygonal objects. A convex polygon P will be called *circular* if there is a point c in its interior (its *center*) so that the line segment from c to a line containing an edge e and normal to e is entirely within P. For instance, the convex hull of a set of points on a circle defines a circular polygon (thus, the name). Note that in this case,

$$\bigcap_{e \in E} \mathrm{slab}(e) \neq \emptyset,$$

where E is the set of edges of P. Clearly, we can find a small neighborhood U of the center c such that active(U) is all of Q. In this case, the problem reduces to simply finding three points q_i, q_j and $q_k \in Q$ such that the residual radius of the resulting triangle is as large as possible.

We need to extend the notion of residual radius as follows: The residual radius of a triangle Δ is the *signed* radius of the largest disk centered at the origin that is either fully outside or fully inside Δ, the sign being positive or negative depending on whether the disk is inside or outside Δ respectively.

Assume that the points of Q are ordered in the anti-clockwise order as

$$q_1 > q_2 > \cdots > q_n.$$

for any point $q \in Q$ its successor, succ(q), is the point immediately following it in the clockwise order.

We start with three arbitrarily chosen distinct points, say $u_0 = q_1$, $u_1 = q_2$ and $u_2 = q_3$, for instance. At any instance, assume that, we have three points u_0, u_1 and u_2, at least two of which are distinct, and

$$u_0 \geq u_1 \geq u_2 \geq u_0$$

There are two cases to consider: (1) they are not all distinct, i.e., $u_i = u_{i+1}$ and (2) they are all distinct and the residual disk touches the edge $u_{i-1}u_i$.

In the first case, we advance the "forward point" u_i (i.e, replace u_i by succ(u_i)) with the hope of making the points distinct (this may not succeed and lead to further advancements of this kind). In the second case, we advance the "backward point" u_i (i.e, replace u_i by succ(u_i)) with the hope of releasing the "limiting edge" $u_{i-1}u_i$ and thus possibly (but not always) increasing the residual radius.

The algorithm keeps advancing the forward or the backward point (as the case may be) while recording the maximal residual radius seen so far until u_0 returns to its initial position, at which point it halts and outputs the edge triple corresponding to the maximal residual radius. Since the polygon is a circular convex polygon, one can easily determine the contact points by taking the normals from the center of the polygon to each edge of the edge triple. The correctness and the complexity analysis of the algorithm can be shown in a manner similar to the discussions in section 5 of the paper by Kirkpatrick et. al. [5] and is omitted here. Note, however, that the above technique fails for arbitrary convex polygons if we relax the condition of circularity.

Note also that the above technique can be easily adapted to the following problem: Given a simple polygon P and a center $c \in \mathbb{R}^2$, find a 3-finger optimal grasp of P such that the inner normals at the contact points go through c. This problem is solved by simply running the above algorithm starting with an active set, active(U) of a small open neighborhood of c. The resulting algorithm takes $O(n)$ time.

C. All Optimal Three Finger Grasps

Sometimes, we wish to determine not just one optimal three finger grasp but all of them. Then we may use any one of this class of optimal grasps, depending on the task at hand. Clearly, the brute force $O(n^3)$ time algorithm will succeed to do so. Note that the algorithm of the previous section cannot be easily modified into a two pass algorithm, since addition of a new point (in the process of going from one cell to an adjacent cell) may create an $O(n)$ edge triplets of residual radius ρ^*. Here, we describe an $O(n^2 \log n)$ algorithm for the special case when the object is convex.

Let P be a convex n-gon and let the possible residual radii (as in the preceding subsection) be given as

$$0 \leq \rho_1 \leq \rho_2 \leq \cdots \leq \rho_i \leq \cdots < 1.$$

We shall find the optimal residual radius ρ^* by performing a binary search on the sequence of possible residual radii. For a given value of ρ_i, we can enumerate all the edge triples that lead to a residual radius of ρ_i in $O(n^2)$ as follows: Corresponding to the possible radius value ρ_i, there are at most $O(n)$ edge pairs (e_i, e_j)'s such that the corresponding points q_i and $q_j \in Q$ on the unit circle satisfy the property that the line determined by $q_i q_j$ is tangent to a circle $C(\rho_i)$ centered at the origin and of radius ρ_i. Now for each such edge pair, we need to check in $O(n)$ time if there is another edge e_k such that $q_k \in Q \setminus \{q_i, q_j\}$ is mutually visible (with respect to $C(\rho_i)$) to both q_i and q_j and that

$$\text{slab}(e_i) \cap \text{slab}(e_j) \cap \text{slab}(e_k) \neq \emptyset.$$

We can thus enumerate all the e_k's that succeed this test. The binary search only considers $O(\log n)$ different values of ρ_i's and terminates with success with the largest possible value ρ^* and enumerating all edge triples corresponding to ρ^*. It is then trivial to describe all possible three finger optimal planar grasps. Thus the algorithm has a time complexity of $O(n^2 \log n)$.

However, the algorithm applied to a nonconvex polygon leads to an $O(n^3)$-time algorithm, as in a pathological case, there may be $O(n^2)$ edge pairs to be considered for a given value of ρ_i. It is noteworthy that this algorithm is rather simple to implement and may perform well in practice. For instance, if one performs binary search on the real interval $[0, 1]$ (instead of the possible radii values), then for a random polygon this algorithm can compute in $O(n \log n \log(1/\epsilon))$ all three finger grasps whose corresponding residual radii lie in the range $[\rho, \rho^*]$ of size $< \epsilon$, for sufficiently small positive ϵ.

D. Optimal Grasps with Four or More Fingers

We still do not know how to find optimal m-finger planar grasp ($m \geq 4$) in time better than what can be obtained by the brute force algorithm taking time $O(n^m)$. For instance, it is not even clear if there is an algorithm to compute such an optimal grasp in time $O(n^{m-1}\text{polylog } n)$. The complication arises by virtue of the torque components that one has to consider in the case when $m \geq 4$.

Some progress has been made, by modifying the problem to that of choosing an optimal set of m-finger contact points out of a preselected $O(n)$ points on the boundary of the polygon, ∂P. For instance, we have an $O(n^3 \log n)$ time algorithm to find such an optimal four finger grasp in this case. The technique employed for this case is a generalization of the preceding algorithm involving binary search. We suspect that the algorithm generalizes to m fingers ($m > 4$) and has a time complexity of $O(n^{m-1} \log n)$.

E. Parallel Jaw and Three-Jaw Grippers

In case of parallel jaw grippers and three-jaw grippers grasping an n-gon, one can compute optimal grasps in time $O(n)$. The algorithms in these cases involve simply going around the object and trying all possible grasps [4]. It is not clear, if these grippers are comparable to multi-fingered hands in terms of how well they optimize various grasp metrics.

F. Fixturing

Another problem of interest is to study the similar optimality problem in the case of "fixturing," where a polygonal object has to be fixtured by a set of toe-clamps that can be placed only at places designated by a set of toe-slots (which are usually arranged on a regular square grid) [1], [9], [10], [22]. The added difficulty arises because of the geometric constraints imposed by the toe-slots. It is easily seen that for a rectilinear object the optimal fixel (fixture element) placement can be determined in $O(n)$ time. However, the problem seems quite difficult even when we consider a convex polygonal object.

G. Reactive Robotics

Recently, we have been able to design "reactive hands" for grasping. These algorithms operate by determining a sensor-dependent binary vector and then actuating a small set of actuators by a simple table-lookup procedure [18], [19]. It remains an intriguing open question whether it is possible to design general multi-fingered reactive hands that always find an optimal grasp.

References

[1] R.C. Brost and K.Y. Goldberg, "A Complete Algorithm for Synthesizing Modular Fixtures for Polygonal Parts," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1994.

[2] J. Czyzowicz, I. Stojmenovic, and J. Urrutia, *Immobilizing a Shape*, Technical report TR-90-37, University of Ottawa, Canada, 1990.

[3] J. Czyzowicz, I. Stojmenovic, and T. Szymacha, *On a Problem of Immobilizing Polygons*, Technical report RR-92-05-6, Université du Québec à Hull, Québec, Canada, 1992.

[4] C. Ferrari and J. Canny, "Planning Optimal Grasps," *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, Nice (France), pp. 2290–2295, 1992.

[5] D. Kirkpatrick, B. Mishra and C.-K. Yap, "Quantitative Steinitz's Theorem with Applications to Multifingered Grasping," *Discrete & Computational Geometry*, Springer-Verlag, New York, **7**(3):295–318, 1992.

[6] W. Kuperberg, *DIMACS workshop on Polytopes*, DIMAC Center, Rutgers University, NJ, 1990.

[7] Z-X. Li and S. Sastry, "Task Oriented Optimal Grasping by Multifingered Robot Hands," *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*, pp. 389–394, 1987.

[8] X. Markenscoff, L. Ni and C.H. Papadimitriou, "The Geometry of Grasping," *International Journal of Robotics Research*, **9**(1):61–74, 1990.

[9] B. Mishra, "Workholding—Analysis and Planning," *Proceedings: IEEE/RSJ International Workshop on Intelligent Robots and Systems*: IROS'91, **1**:53–57, Osaka, Japan, 1991.

[10] B. Mishra, "An Algorithmic Approach to Fixturing," *1994 NSF Design and Manufacturing Grantees Conference*, M.I.T., Cambridge, MA, 1994.

[11] B. Mishra, "Grasp Metrics: Optimality and Complexity," **Algorithmic Foundations of Robotics**, (Edited by K. Goldberg, D. Halperin, J.-C. Latombe and R. Wilson), pp. 137–166, A.K. Peters, Wellesley, Massachusetts, 1995.

[12] B. Mishra, J.T. Schwartz and M. Sharir, "On the Existence and Synthesis of Multifinger Positive Grips," Special Issue: Robotics, *Algorithmica*, **2**(4):541–558, 1987.

[13] B. Mishra and N. Silver, "Some Discussion of Static Gripping and Its Stability," *IEEE Transactions on Systems, Man and Cybernetics*, **19**(4):783–796, 1989.

[14] B. Mishra and M. Teichmann, "On Immobility," Special Issue: Robot Kinematics, *Laboratory Robotics and Automation*, **4**:145–153, 1992.

[15] V. Nguyen, "Constructing Force-Closure Grasps," *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, pp. 1368–1373, San Francisco, CA, 1986.

[16] J. O'Rourke, "Computational Geometry Column 9," *SIGACT News*,**21**(1):18–20, 1990.

[17] J. O'Rourke, "Computational Geometry Column 11," *International Journal of Computational Geometry & Applications*, **1**(1):93–98, 1991.

[18] M. Teichmann and B. Mishra, "Reactive Algorithms for Grasping Using a Modified Parallel Jaw Gripper," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*: ICRA'94, San Diego, CA, 1994.

[19] M. Teichmann and B. Mishra, "Reactive Algorithms for 2 and 3 Finger Grasping," *Proceedings of the 1994 IEEE/RSJ International Workshop on Intelligent Robots and Systems*: IRS'94, Grenoble, France, 1994.

[20] M. Teichmann, *Grasping and Fixturing: a Geometric Study and an Implementation*, Ph.D. Thesis, Courant Institute, New York University, NY, 1995.

[21] J.C. Trinkle, "A Quantitative Test for Form Closure Grasps," Department of Computer Science, Texas A&M University, College Station, TX, 1992.

[22] Y. Zhuang, K.Y. Goldberg and Y-C. Wong, "On the Existence of Modular Fixtures," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1994.

Bud Mishra is a Professor of Computer Science at the Courant Institute of New York University. He received his B.Tech. (1980) from the Indian Institute of Technology in Communication Engineering and M.S. (1982) and Ph.D. (1985) degrees in Computer Science from Carnegie Mellon University. Professor Mishra is interested in mathematical and algorithmic methods of computer science with applications to various real-life problems. His research interests span many areas of computer science (e.g. VLSI tools, compiler design, real-time, asynchronous and self-timed systems, design of discrete controllers, algorithms and data structures, complexity, computational algebra and geometry, computational biology, computational finance), mathematics (logic, combinatorics, algebra) and robotics. In robotics area, his primary interest has been in the area of dexterous manipulation, where he studies manipulation schemes with polydactyl robot hands. His most recent projects include *reactive robotics*, a new control paradigm for robotics, *DigiLand*, an object-oriented distributed virtual economy and a collaboration on the human Genome project. Professor Mishra has acted as a technical consultant for various computer and financial industries.

Marek Teichmann is a Post Doctoral Fellow in the Computer Science Department at the Courant Institute of New York University. He received his B.S. (1987) in Mathematics and Computer Science and M.S. (1989) in Computer Science from McGill University, Montréal. He received his Ph.D. (1995) degree from the Courant Institute, New York University. His research focuses on applications of discrete and computational geometric techniques to robotics, manufacturing and computer graphics. His thesis contains an in-depth geometric analysis of various issues of grasping, randomized approximation algorithms for optimal grasp and fixture synthesis as well as a formulation of reactive robotics schemes, resulting in an implementation of a robust and fast reactive gripper.