Planning Grasps for a Robot Hand in the Presence of Obstacles

Nancy S. Pollard
The MIT Artificial Intelligence Laboratory
545 Technology Square
Cambridge, MA 02139

Abstract

In grasp planning for a robot hand, the impact of the environment geometry on grasp choice is often of paramount importance. This paper presents a partial solution to the problem of fitting a hand to an object in a given environment. Information provided by a set of fingertip contact regions or by a grasp prototype is exploited to efficiently compute the space of wrist configurations from which the grasp specifications can be met. The size and shape of this solution space reflect the constraints imposed by the environment geometry; they can be used to compare a grasp to others or to resolve grasp symmetries (e.g. in a wrap grasp of a cylinder). A quality metric computed over the solution space can be used to optimize a grasp. The paper describes a general method for computing this solution space, and presents a parallel implementation. Examples demonstrate grasp comparison and symmetry resolution.

1 Introduction

Grasp planning is plagued by two major computational problems: the search space is very large, and the constraints that must be satisfied are many and varied. The size of the search space is due to the large number of joints in a robot arm and hand. Performing a comprehensive or poorly guided search through this space for a good grasp is simply not feasible. Constraints such as the geometry and location of the target object, specifications on the task to be performed, and the placement of obstacles in the environment can help to focus the search, but these different constraints seem difficult to combine in any satisfactory way.

Methods developed to make the grasp planning problem more tractable generally focus on a subset of the constraints available. There are two main approaches: to focus on object geometry and contact force optimization; or to focus on hand kinematics and the fit of hand to object.

Work focusing on object geometry and contact force optimization includes that of Ferrari and Canny [8], Li and Sastry [14], and many others [17] [18] [19] [12]). Task specifications determine the optimization criteria, and may be general as in [8] or specific to the task as in [14].

Work on hand kinematics begins with the parallel jaw gripper, which has only a single degree-of-freedom (Lozano-Pérez et al. [16], Pertin-Troccaz [21], and others [13] [11] [20]). The problem of reducing uncertainty in part orientation (as in part feeder design) is similar; the kinematic space of the problem is small and can be searched completely (Goldberg and Mason [9], Brost [3], Christiansen et al. [5], and Erdmann et al. [7]).

When a more complex hand is considered, the problem of fitting hand to object becomes much more difficult. One approach makes use of a small set of predetermined grasps or grasping behaviors (see Bard and Troccaz [1], Cutkosky and Howe [6], Iberall and MacKenzie [10]). Once a grasp or behavior has been chosen, the hand can be treated as a simple device. Sensor feedback can be used to help deal with variations and uncertainties in the environment (as in Brock [2] and Chammas [4]), but the remaining problem is very similar to planning grasps for a parallel jaw gripper.

The configuration of the world is generally ignored in the work mentioned above. But unless the environment is under the control of the designer, any grasp planner must eventually deal with obstacles. What little work has addressed this problem has either used a projection of obstacles into the plane of motion of a parallel jaw gripper (see Lozano-Pérez et al. [16] and Pertin-Troccaz [21]) or a potential-field local search for a hand (see Pollard [22]). Neither method is entirely satisfactory, because neither produces global information about how the environment affects the space of grasps that are possible.

The goal of this paper is to demonstrate that a global description of the effect of environment geometry on grasp choice can be very useful. Even if contact placement information is available, or even if we are working from a prototype grasp, it is helpful to have this information about the environment; we want to know how well the hand fits into the space around the target object or how a prototype grasp might be adjusted to allow for a better fit. The sections below outline a method for obtaining a certain class of global information without searching the entire hand/arm workspace, and provide some data to show an application of the method to two different grasping problems: finding a grasp when contact points are given, and resolving symmetries in a prototype grasp.

2 Problem Definition

The problem addressed in this paper is to find a collision-free grasp of a target object in a given environment. The implementation and examples in the paper are based on the Salisbury hand, which has three fingers, each with three joints (Figure 1 shows a simple model). Fifteen parameters are required to fully specify the hand configura-

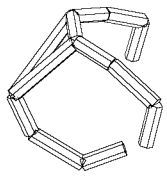

Figure 1: *A simple model of the Salisbury hand. The triangle represents the wrist.*

tion: nine for the joint angles and six for the wrist configuration (position and orientation).

A brute force approach to the problem of fitting this hand to a target object would be to build the entire 15-dimensional hand configuration space and evaluate each configuration according to some measure of grasp quality: Is the grasp free of collisions? Is the hand in contact with the object? How well does the force/torque distribution that can be achieved from these contacts fit the force/torque distribution expected from the task?

Unfortunately, we are far from having the time or computational power to build even a significant fraction of such a space. Instead, we will compute a 6-dimensional projection: the space of wrist configurations from which the hand can achieve a certain grasp or type of grasp without colliding with obstacles in the environment.

The sections below describe how this solution space can be efficiently computed. Collisions involving the robot arm are ignored; they could be added in a straightforward manner by computing wrist configurations that are free from arm collisions (as in Lozano-Pérez et al. [15]) and intersecting them with wrist configurations from which the hand can achieve a collision-free grasp. In addition, the output of this process is a collision-free end grasp, not an approach or a path or a grasping behavior. There is no guarantee that a path exists for the hand to reach this end grasp.

3 Building the Solution Space

The solution space for this problem is the 6-dimensional space of wrist configurations from which a specified type of collision-free grasp can be achieved. This section defines a class of grasp planning problems for which solutions can be constructed within a 6-dimensional space. It outlines the dynamic programming technique used to construct these solutions, and describes a parallel implementation.

3.1 Iterative Optimization

A solution can be easily constructed within a 6-dimensional space if the contribution of each link configuration can be evaluated separately. This condition allows us to construct globally optimal chains of links by iteratively

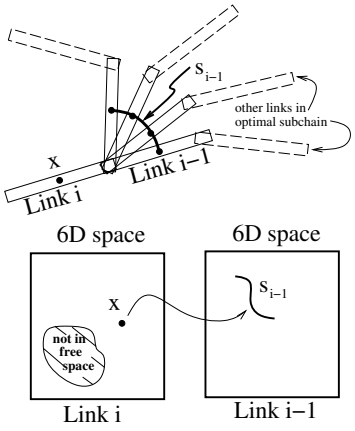

Figure 2: *An optimal chain of links can be constructed by repeated scanning for an optimal subchain.*

adding links from the distal to the proximal end of each finger. Each iteration (Figure 2) involves a scan operation over the range of joint angles possible between links. The optimal subchain identified by the scan is attached to the current link, and the process repeats.

As an example, suppose that we can specify the type of contact we would like to achieve at each link of the finger. We want to disqualify a chain immediately if any link does not lie in free space, and we may have some additional constraints on the configuration of each link to ensure that it matches a given class of hand shapes. Then, if $f_i(x_i)$ is the value computed for the single link i at configuration x_i, $F_i(x_i)$ is the maximal value for the subchain ending at link i, configuration x_i, and $s_{i-1}(x_i, \theta_{i-1})$ is the mapping from configuration x_i and joint angle θ_{i-1} to configuration x_{i-1}, we can write:

$$\begin{aligned} f_i(x_i) &= 1 \text{ for (proper contact type)} \\ &\textit{else} \quad 0 \end{aligned}$$

$$\begin{aligned} F_0(x_0) &= 0 \text{ for (illegal } x_0\text{)} \\ &\textit{else} \quad f_0(x_0) \end{aligned}$$

$$\begin{aligned} F_i(x_i) &= 0 \text{ for (illegal } x_i\text{)} \\ &\textit{else} \quad f_i(x_i) \wedge \bigvee_{\theta=\theta_{min,i}}^{\theta_{max,i}} F_{i-1}(s_{i-1}(x_i, \theta_{i-1})) \end{aligned}$$

Where "illegal x_i" is defined as follows:

$$\begin{aligned} \text{(illegal } x_i\text{)} &= (x_i \text{ outside free space}) \vee \\ &\quad (x_i \text{ outside expected range}) \end{aligned}$$

If the finger has n links, we calculate $F_{n-1}(x_{n-1})$. Any configuration x_{n-1} for which $F_{n-1}(x_{n-1}) = 1$ satisfies the requirements given. Any sequence of θ's by which it was reached provides us with a complete, legal solution for that finger. Any wrist configuration for which the requirements of all fingers and the palm can be satisfied is a legal wrist configuration.

3.2 General Implementation Notes

The iterative optimization technique described in the previous section has been implemented using a parallel algorithm. If we had a machine large enough to hold a reasonably fine tessellation of the 6-dimensional link configuration space, this implementation would be a straightforward matter. We would build the separate configuration spaces for each link of the hand; construct optimal finger configurations using the iterative scan operation of the previous section; and combine the optimal fingers to obtain the optimal grasp at each point in the 6-dimensional wrist configuration space. We do not have enough memory to hold such a large workspace, however. The following paragraphs describe a representation using a 3-dimensional grid of processors.

Workspace Representation: In this implementation, the 3-dimensional space of wrist *positions* is tessellated and divided among a 32x32x32 cube of processors on a Thinking Machines CM2 Connection Machine, a massively parallel SIMD (Single Instruction, Multiple Data) computer. Each processor represents a cube 0.5 inches on a side. The remaining wrist configuration space, the space of wrist orientations, is represented as a map of 180 orientations, one map in each processor.

The hand workspace is explicitly sampled at each wrist orientation. In the current implementation, for the Salisbury hand, 3 to 6 samples are taken for each joint angle, producing 45 sample configurations of the top fingers and 90 sample configurations of the thumb at each wrist orientation. Each sample finger configuration is represented as a path through the processor grid from fingertip to wrist. Allowing a range of joint motion about a sample finger shape produces a set of paths that looks like a tree rooted at the wrist.

Collecting Solutions: We build the 6-dimensional solution space by looping over all wrist orientations; looping over all finger workspace samples; collecting the contribution of each finger by following the precomputed paths from fingertip to wrist; and combining the optimal fingers to obtain an optimal grasp. Because each tree of paths from fingertips to wrist is specified relative to the final wrist position, grasp information can be collected along these paths for all wrist positions in parallel.

The 3-dimensional workspace representation we are using here does not allow us to precompute the separate contributions of each link of the hand, so this information must be generated on the fly. For the examples in this paper, each link must be tested to see that it lies completely within free space and that it meets any given contact conditions. The process works as follows: when data enters a processor, the processor must decide whether it is sufficiently far from obstacles in the environment for a section of the link to safely exist at its point in space. If not, the processor blocks all information flow. Since each processor makes its decision based on distance to the nearest obstacle, the model of the link implicit in this implementation is a cylinder with a diameter that may vary along the link length.

A similar mechanism is used to check for contact with the target object. A processor computes its distance to the target object and determines the forces that could be generated on the object at that point. The processor can then tag incoming data if the desired type of contact can be achieved with a section of the link at that point.

Computation time for code written in C and C* (parallel C), running on a Sun and a CM2 is approximately one minute for the examples that follow.

4 Fixed-Contact Grasps

Geometric analysis of a target object can generate sets of tip contact points that would give a grasp a good force/torque distribution. A representation of the wrist configurations from which each set of contact points can be achieved could be useful for comparing sets of grasps to one another. It could also be used to compare wrist configurations that are feasible when obstacles are present to those expected without obstacles. Such comparisons could facilitate the choice between grasps or allow for an objective measure of the effect of the environment on a chosen grasp.

4.1 Implementation

Since contact locations are given, we need only collect a yes or no answer for each wrist configuration: can each finger reach the assigned tip contacts while avoiding collisions? The objective function F_i defined in Section 3 can be used here.

For simplicity, we assume that nothing is known about wrist orientation, and so the 180 sample orientations are distributed evenly in the three-dimensional wrist orientation space.

4.2 Results

Results are presented for a grasp of a cylindrical object. Two sets of contact points are tested. For each set we can calculate the free wrist configuration space and measure the size and shape of the largest free region.

The left half of Figure 3 shows the shape of free wrist *position* space for selected slices of Cartesian space as represented by the grid of processors on the CM2. The slices are perpendicular to the cylindrical target object, which is in the center of each slice. Read left to right, top to

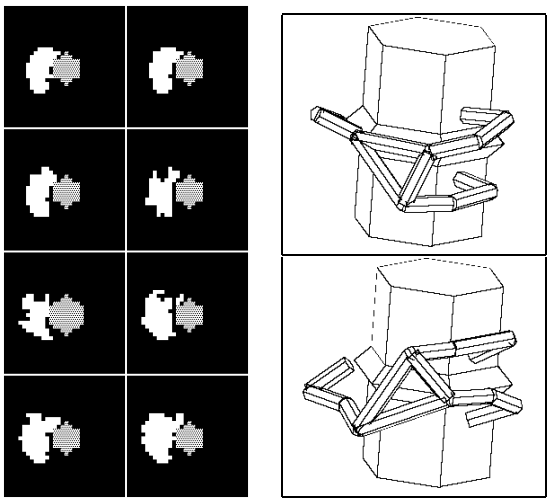

Figure 3: *(L) Cartesian space slices. Obstacles are gray; legal wrist positions are white. (R) Two central solutions.*

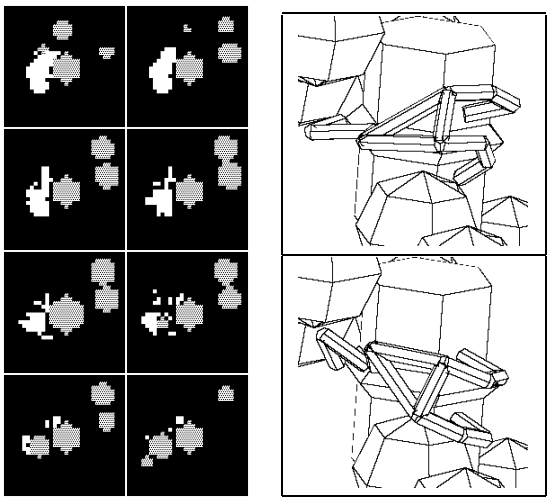

Figure 4: *(L) Cartesian space slices. Obstacles are gray; legal wrist positions are white. (R) Two central solutions.*

bottom, they move down the axis of the cylinder. A wrist position is highlighted if there is some orientation from which the designated grasp can be achieved at that position.

The right half of Figure 3 shows two solutions from a central wrist position, but from two different wrist orientations. Figure 4 shows results for a situation where obstacles have been added.

Figure 5 shows slices from environments with and without obstacles when the given contact points force the hand

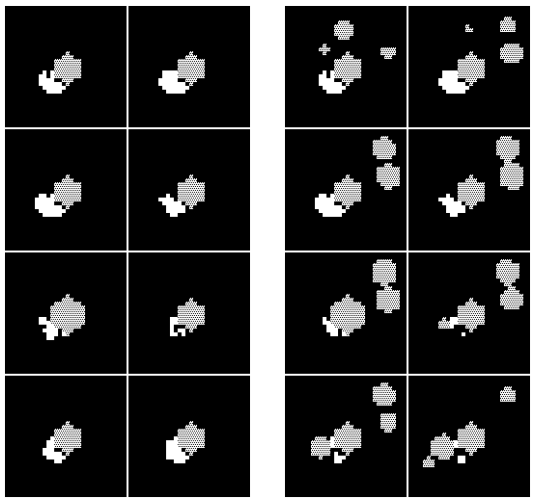

Figure 5: *Legal wrist position space. The figure shows a tighter grasp with and without obstacles.*

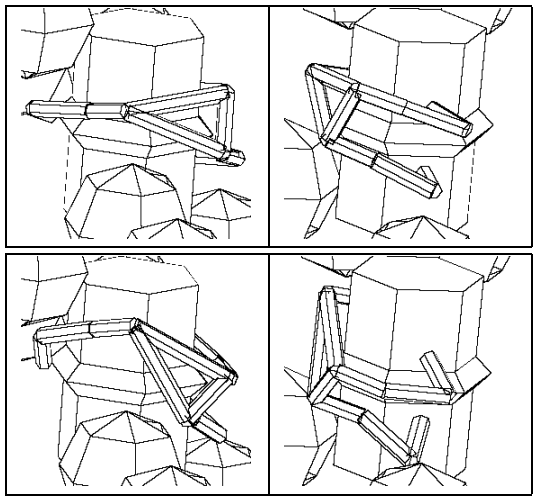

Figure 6: *Two solutions from a central wrist point. Each row shows a single grasp from two different views.*

to wrap more tightly around the object. Figure 6 shows two views each of two solutions from a central free wrist position.

From these data, we can make the following observations:

- The looser grasp is affected more severely by the presence of obstacles. The tighter grasp is affected less, in part because the added constraints on the grasp keep the hand near the target object.
- Both grasps must be adjusted upward to avoid the sphere interfering with the right finger.

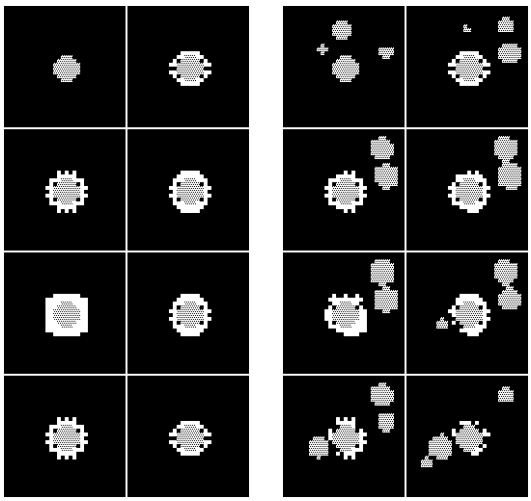

Figure 7: *Legal wrist position space for a wrap grasp with and without obstacles. Obstacles are gray; legal wrist positions are white.*

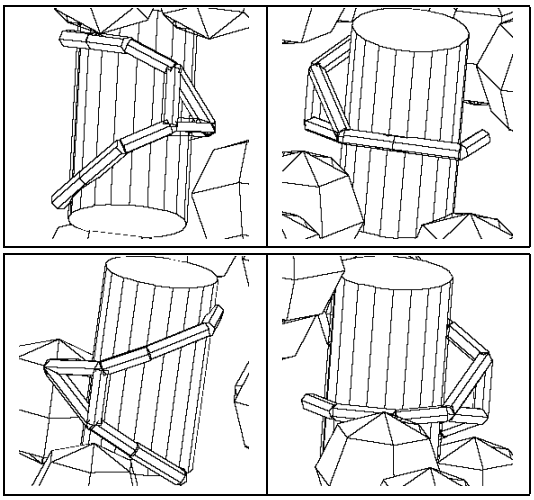

Figure 8: *Two different cylinder wrap grasps. Each row is a single grasp from two views.*

Given enough time and patience, we could use this technique to zero in on a good set of fingertip contact points, but this is not an effective way to resolve the symmetries in this problem. The next section shows a cleaner solution using essentially the same technique.

5 A Prototype Wrap Grasp

Although fingertip contact regions can be useful, a different sort of information may come from a grasp library. A grasp chosen from a library will match the shape of the hand to the shape of the target object. In some cases, however, such a grasp may have symmetries that need to be resolved. A representation of the wrist configurations from which the grasp can be reached without collision can help to resolve these symmetries. Results are presented here for a cylinder wrap grasp.

5.1 Implementation

If we assume that the hand shape of a prototype grasp is fixed (i.e. not parameterized), a wrap grasp of a cylinder has two degrees of freedom: one in position, one in orientation. We do not know exactly where the fingertips will contact the object, but we do know the expected hand shape and the expected types of contact at each of the links of the hand. With this information, we can also use the objective function F_i defined in Section 3.

In this example we can make more effective use of the ability to concentrate resources around grasp expectations. The 180 wrist orientations are concentrated around the single orientation degree-of-freedom in the grasp, and the sampling of the hand workspace can be concentrated around expected joint angles. If it were not for the coarse 32x32x32 position space grid, the sampling for this problem would be extremely accurate.

5.2 Results

Figure 7 shows selected slices of the free wrist position space in environments both with and without obstacles. Notice how the free region is broken near the bottom of the right-hand figure.

Figure 8 shows two solutions generated from different points in the region of free wrist configurations. The solution on the bottom row shows the thumb nearly in contact with the sphere causing the largest disturbance in the bottom slices of Figure 7.

6 Summary and Discussion

The goal of this paper was to illustrate that we can calculate useful global information about the effects of environment geometry on grasp choice. A method was illustrated for generating the 6-dimensional wrist configuration space from which a certain grasp or type of grasp can be achieved in a given environment. It was shown that this 6-dimensional solution can be calculated within a 6-dimensional space if enough information about the grasp is available to allow separate calculation of the contribution of each link configuration. An implementation was outlined using a parallel representation of position space and a serial representation of orientation space.

Two example solutions were shown, one where fingertip contact regions were known, and one where a prototype

grasp was given. These examples demonstrated that this method can be used to compare grasps on the basis of the room available for the wrist to maneuver. They also showed how a prototype grasp can be tuned to achieve the desired types of contacts at the links. This second property could be used to solve problems that are much less structured than those shown here. It is discussed further in the paragraphs below.

6.1 The Next Step

The method described in this paper is not particularly effective when the grasp given fits the object perfectly. A wrap grasp of a cylinder, for example, has only two degrees of freedom, so computing a 6-dimensional solution space is probably not the most efficient way to resolve these symmetries in the grasp.

This method will be more effective when the target object is irregular and when it is not clear how the grasp will have to be adjusted to fit the object. In this case, we will want to loosen the constraints on hand shape and make some measurement that tells us more about the match between each contact and the corresponding contact on the prototype grasp. A more formal definition of the contribution each contact makes to the stability or quality of the grasp as a whole will allow a more careful ranking of grasps based on their *functional* fit to a prototype. This is the subject of current work.

Acknowledgments

Many thanks to David Siegel, Maja Mataric, and Tomás Lozano-Pérez for helpful comments on this paper. I would also like to thank the anonymous reviewers, whose comments were very detailed and constructive.

This paper describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the University Research Initiatives under Office of Naval Research contract N00014-86-K-0685 and in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-91-J-4038.

References

[1] C. Bard and J. Troccaz. Automatic preshaping for a dextrous hand from a simple description of objects. In *Proc. IEEE Intl. Workshop on Intelligent Robots and Systems*, Tsuchiura, Ibaraki, Japan, July 1990.

[2] D. L. Brock. Implementation of behavioral control on a robot hand/arm system. In *Proc. 2nd International Symposium on Experimental Robotics*, Toulouse, France, June 1991.

[3] R. C. Brost. Automatic grasp planning in the presence of uncertainty. In *Proc. IEEE ICRA*, San Francisco, April 1986.

[4] C. Z. Chammas. Analysis and implementation of robust grasping behaviors. Technical Report AI-TR-1237, MIT, May 1990.

[5] A. Christiansen, M. Mason, and T. Mitchell. Learning reliable manipulation strategies without initial physical models. In *Proc. IEEE ICRA*, Cincinnati, Ohio, May 1990.

[6] M. R. Cutkosky and R. D. Howe. Human grasp choice and robot grasp analysis. In S.T. Venkataraman and T. Iberall, editors, *Dextrous Robot Hands*. Springer-Verlag, New York, 1990.

[7] M. Erdmann, M. Mason, and G. Vanecek, Jr. Mechanical parts orienting: The case of a polyhedron on a table. In *Proc. IEEE ICRA*, Sacremento, California, 1991.

[8] C. Ferrari and J. Canny. Planning optimal grasps. In *Proc. IEEE ICRA*, Nice, France, May 1992.

[9] K. Goldberg and M. Mason. Bayesian grasping. In *Proc. IEEE ICRA*, Cincinnati, Ohio, May 1990.

[10] T. Iberall and C. L. MacKenzie. Opposition space and human prehension. In S.T. Venkataraman and T. Iberall, editors, *Dextrous Robot Hands*. Springer-Verlag, New York, 1990.

[11] J. L. Jones and T. Lozano-Pérez. Planning two-fingered grasps for pick-and-place operations on polyhedra. In *Proc. IEEE ICRA*, Cincinnati, Ohio, May 1990.

[12] J. Kerr and B. Roth. Analysis of multifingered hands. *IJRR*, 4(4), Winter 1986.

[13] C. Laugier, A. Ijel, and J. Troccaz. Combining vision based information and partial geometric models in automatic grasping. In *Proc. IEEE ICRA*, Cincinnati, Ohio, May 1990.

[14] Z. Li and S. Sastry. Optimal grasping by multifingered robot hands. In *Proc. IEEE ICRA*, Raleigh, North Carolina, 1987.

[15] T. Lozano-Pérez, J. L. Jones, E. Mazer, and P. A. O'Donnell. *HANDEY A Robot Task Planner*. MIT Press, Cambridge, MA, 1992.

[16] T. Lozano-Pérez, J. L. Jones, E. Mazer, P. A. O'Donnell, W. E. L. Grimson, P Tournassoud, and A. Lanusse. Handey: A task-level robot system. In *4th Intl. Symp. on Robotics Research*, Santa Cruz, CA, 1987.

[17] X. Markenscoff and C. H. Papadimitriou. Optimum grip of a polygon. *IJRR*, 8(2), April 1989.

[18] B. Mishra, J. T. Schwartz, and M. Sharir. On the existence and synthesis of multifinger positive grips. *Algorithmica*, 2(4), 1987.

[19] V. Nguyen. The synthesis of stable force-closure grasps. Technical Report AI-TR-905, MIT, July 1986.

[20] I. Mazon P. Violero and M. Taix. Automatic planning of a grasp for a pick and place action. In *Proc. IEEE ICRA*, Cincinnati, Ohio, May 1990.

[21] J. Pertin-Troccaz. On-line automatic programming: A case study in grasping. In *Proc. IEEE ICRA*, Raleigh, North Carolina, 1987.

[22] N. S. Pollard. The grasping problem: Toward task-level programming for an articulated hand. Master's thesis, MIT, May 1989.