

Extending Fingertip Grasping to Whole Body Grasping

Robert Platt Jr.

Andrew H. Fagg

Roderic A. Grupen

Laboratory for Perceptual Robotics
Department of Computer Science
University of Massachusetts, Amherst
{rplatt, fagg, grupen}@cs.umass.edu

Abstract

Although it is mechanically possible for a robot manipulator to grasp using non-fingertip contacts, there are few examples of this. We call non-fingertip grasping such as grasping with proximal finger phalanges or grasping with the sides of arms “whole body grasping.” While robotic demonstrations are rare, humans commonly use whole body grasps to interact with the world. One of the distinctive features of whole body grasping is the kinematic coupling among potential contacts. This kinematic coupling introduces extra constraints into the grasp synthesis problem. In this paper, we extend recent grasp control techniques to whole body grasping. We show how the grasp control may be parameterized with a set of contact points on the surface of the robot manipulator that enables the grasp search to handle the extra kinematic coupling constraints and find whole body grasps.

1 Introduction

One of the fundamental problems in robotics is that of grasping and manipulating an object. While robotic grasping research typically assumes that fingertips alone will be used to grasp objects, non-fingertip contacts are also possible. For example, potential contact points may exist along the entire surface of a finger - not just at the fingertip. We use the term “whole body grasp” to refer to grasps that depend on non-fingertip contacts on arbitrary hand and body surfaces.

Whole body grasping is a critical generalization of fingertip grasping. When using a hammer for example, a human does not rely on fingertip contacts alone for a secure hold. Instead, a human is likely to wrap his thumb and all four fingers around the handle. This grasp not only puts as many hand surfaces in contact with the handle as possible, but it also makes stronger and more proximal hand surfaces part of the grasp.

Unfortunately, synthesizing whole body grasps can

be difficult because kinematic coupling among non-fingertip contacts may prevent desired contact points from being made. Therefore, manipulator kinematics must be considered when synthesizing grasps. In this paper, we show that this may be accomplished using virtual fingers, opposition spaces, and a proximal to distal sequencing of grasp resources.

Proximal contacts are those located on manipulator surfaces close to the root of the kinematic tree. Conversely, distal contacts are located far away from the root of a kinematic tree. On a humanoid robot, for example, the fingertips are distal to the palm and the palm is proximal to the fingertips. We show that an effective way to find whole body grasps is to precede grasp synthesis for distal contacts (fingertips) with grasp synthesis for proximal contacts (palm). This approach is based on the fact that when a contact is made, it is easier to freeze the joints proximal to the contacts than to calculate self motion of the manipulator with respect to the contacts. Acquiring contacts in proximal to distal order allows for an ordered commitment of degrees of freedom.

We first give an overview of relevant robotics literature on the subject of whole body grasps. Section 3 reviews some of our previous work on grasp controllers and describes how arbitrary hand contacts can be controlled. Section 4 describes how virtual fingers and opposition spaces can parameterize a grasp controller, and why a proximal to distal ordering of the grasp planning process yields robust grasp solutions. Finally, we describe experiments demonstrating the effectiveness of our approach.

2 Previous Work

Grasp synthesis has been an active subject of research for more than two decades. A large portion of this work has focused on how to place disembodied contacts on an object so as to accomplish either force closure or form closure, one of two sufficient condi-

tions for being able to firmly grasp an object. For example, Faverjon [3] and Nguyen [5] have developed algorithms for placing contacts on objects of known geometry. Their methods are based on a detailed exploration of geometrically sufficient conditions for developing a secure grasp on an object. Work in this category typically does not directly use hand kinematic information in grasp planning.

Some researchers have also studied ways of using “whole hand” contacts to construct grasps. Pollard has described a way to synthesize enveloping grasps from prototypes [7]. This approach selects a whole hand grasp prototype from a collection of possible grasps and adapts it to a particular situation. Reznik has leveraged the ability of highly redundant fingers to follow the contours of an object [9].

The present work is based on previous work by Coelho and Grupen [1] and Platt *et al.* [6]. That previous work explores a grasp controller that displaces contacts on the surface of an object of unknown geometry so as to reduce error functions related to necessary conditions for force closure. The grasp controller does not take the location of the contact in the kinematic tree into account. The current work addresses this issue. We show how grasp controllers can be effective in cases where manipulator kinematics information must be used in the grasp planning process.

3 A Contact Displacement Controller

Our grasp controller functions by executing a series of regrasps. On each regrasp, the fingers close until they lightly make contact with the object. The grasp controller uses tactile sensor information to compute controller error and error gradient with respect to contact position. After making light contact with the object, the controller lifts the contacts off the object and displaces them by a small amount in the direction of the negative error gradient.

The grasp controller descends an error function that minimizes net wrench. Several researchers, Ponce [8] for example, have shown that for a given coefficient of coulomb friction, net zero wrench with at least three contacts is a sufficient condition for force closure in an environment with a strictly greater coefficient of friction.

3.1 Force and Moment Contact Control Laws

The grasp controller reduces net wrench error by separately minimizing net force and moment. These two objectives are combined using a nullspace control composition. The primary objective of the controller is to reduce net force. The controller also simultaneously

minimizes net moment as long as this is consistent with the net force objective. These two objectives are the basis of two control laws: the force-based contact position control law and the moment-based contact position control law.

The force-based contact position control law ϕ_{force} is a potential function that has equilibria in configurations where the contacts exert the reference net force. Without loss of generality, we hereafter assume this reference to be zero. Let \vec{f} be the net force vector applied by the set of contacts (each contact is assumed to apply a unit force that is tangential to the sensed surface normal). The contact configuration error is defined as:

$$\epsilon_f = \vec{f}^T \vec{f}. \quad (1)$$

The gradient of ϵ_f with respect to contact location is $\frac{\partial \epsilon_f}{\partial \vec{x}}$. ϕ_{force} follows $-\frac{\partial \epsilon_f}{\partial \vec{x}}$ by repositioning the contacts, \vec{x} , on the surface of the object.

The moment-based contact position control law ϕ_{moment} functions similarly to ϕ_{force} . It has equilibria in configurations where the contacts exert zero net moment. Let \vec{m} be the net moment vector. Contact configuration error is defined as:

$$\epsilon_m = \vec{m}^T \vec{m}. \quad (2)$$

As with the force-based control law, ϕ_{moment} follows the negative of the gradient $\frac{\partial \epsilon_m}{\partial \vec{x}}$. More detail on both of these control laws can be found in Platt *et al.* [6], and Coelho and Grupen [1].

3.2 Combining Grasping Control Laws

As we have described, net force and net moment are minimized separately. We combine these control laws using the “subject to” constraint as follows:

$$\Phi_{moment} \triangleleft \Phi_{force}.$$

The above expression uses \triangleleft to express the “subject to” relationship. This expression should read: Φ_{moment} subject to Φ_{force} . The controller written above will reconfigure the manipulator to try to minimize net force as a first priority. If possible, it will also try to minimize net moment.

The “subject to” constraint is shorthand for a projection of one control law into the nullspace of another. For ϕ_{moment} contact displacements not to disrupt ϕ_{force} , they must not affect net force. Therefore, they must be projected into the nullspace of $\frac{\partial \vec{f}}{\partial \vec{x}}^\#$. In this expression, $(\cdot)^\#$ denotes the pseudoinverse and \vec{x} is the generalized position. The following projects the moment control law into the nullspace of the force control law:

$$\frac{\partial \epsilon}{\partial \vec{x}} = \frac{\partial \epsilon_f}{\partial \vec{f}} \frac{\partial \vec{f}}{\partial \vec{x}} + \frac{\partial \epsilon_m}{\partial \vec{m}} \frac{\partial \vec{m}}{\partial \vec{x}} \left(I - \frac{\partial \vec{f}}{\partial \vec{x}} \frac{\partial \vec{f}}{\partial \vec{x}}^\# \right). \quad (3)$$

The combined controller displaces contacts in the direction of $-\frac{\partial \epsilon}{\partial \vec{x}}$ so as to reduce the combined error ϵ .

3.3 Displacing Contacts

Contact displacement is accomplished using resolved motion rate control. The displacement is decomposed into three motions: a lift-off from the object, a displacement, and a reacquiring of contact. Each motion is the result of a resolved motion rate control loop which iterates on an augmented Jacobian:

$$\frac{\partial \vec{x}}{\partial \vec{q}} = \left(\frac{\partial \vec{x}_1^T}{\partial \vec{q}} \quad \cdots \quad \frac{\partial \vec{x}_k^T}{\partial \vec{q}} \right)^T. \quad (4)$$

In this equation, \vec{q} denotes the manipulator joint space and $\frac{\partial \vec{x}_i}{\partial \vec{q}}$ denotes the manipulator Jacobian for the i^{th} finger. Combining each of the $\frac{\partial \vec{x}_i}{\partial \vec{q}}$ s into a single Jacobian enables the system to attempt simultaneously to accomplish all contact displacements.

4 Contacts on Arbitrary Body Surfaces

The grasp controller described in the last section searches for optimal configurations of disembodied contacts. In order to use this technique on a robot, however, the contacts must be acquired by touching the object with surfaces of the hand or arm. For many robot manipulators, this means that some configurations are better for grasping than others. Figures 1a and 1b illustrate two possible grasps of a cylinder. Figure 1a shows a humanoid hand grasping a cylinder by opposing the thumb with the fingers. Figure 1b shows the hand grasping a cylinder by attempting to oppose the thumb and index finger with the ring finger. This illustrates how the morphology of the human hand is better suited to the grasp in Figure 1a than to the grasp in Figure 1b.

Manipulator kinematics must be accounted for in the grasp search process. Suppose that the grasp controller described in Section 3 were configured to use contacts on both the fingertips and palm. This controller would attempt to improve the grasp in Figure 1b by moving the ring and index fingers further around the object in opposition to the palm. Moving the index finger toward the ring finger so as to oppose the thumb would increase grasp error before the grasp began to improve. The tight kinematic coupling between the palmar contact and the finger contacts prevents the index finger from opposing the thumb. This illustrates a local minima in grasp configuration imposed by the kinematic structure of the hand.

This section will introduce three ideas that build on each other to solve this problem. Each idea provides a way to inject manipulator morphology information

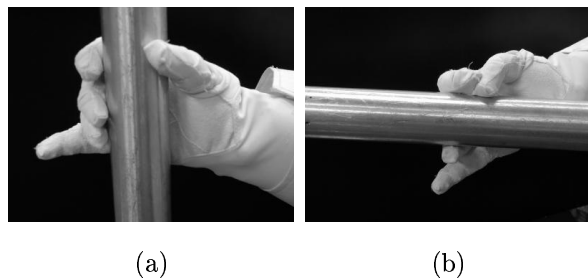


Figure 1: (a) Robonaut hand illustrating a grasp that takes advantage of hand kinematics by opposing thumb and fingers. (b) Robonaut hand illustrating a grasp where the index finger and thumb oppose the ring finger. The kinematics of the hand have caused the grasp controller to get stuck in a local minima.

into the grasp search process. The first idea is that of the “virtual finger.” A virtual finger is a way to abstract a group of contacts into a single contact. The second idea is the “opposition space.” An opposition space is two or more contacts or virtual fingers which oppose each other. The third idea is to structure the grasp search process by giving proximal contacts higher priority than distal contacts.

4.1 Virtual Fingers

One way manipulator kinematics can inform the grasp search process is through the use of “virtual fingers.” A virtual finger is a set of fingers or other hand/arm surfaces which provide a single oppositional force [4]. For example, humans often use their four fingers as a single virtual finger in opposition to the thumb. Figure 2 illustrates this on a humanoid robot hand. The small black dots are contacts and the large ellipses represent virtual fingers. Contacts which comprise one of the virtual fingers all act to provide a single oppositional force.

We implement a virtual finger on the robot by grouping multiple physical contacts together to function as a single contact. Position and normal information from each of the contacts in the virtual finger is averaged:

$$\vec{x}_{VF} = \sum_{i \in VF} \vec{x}_i / |VF| \quad (5)$$

$$\vec{n}_{VF} = \sum_{i \in VF} \vec{n}_i / \left| \sum_{i \in VF} \vec{n}_i \right| \quad (6)$$

In these equations, \vec{x}_i and \vec{n}_i are the location and normal respectively for the i^{th} contact in the virtual finger.

This single averaged contact is used by the grasp controller to compute force and moment error. After the controller calculates a regrasp displacement for the virtual finger, each of the member contacts executes the same displacement.

4.2 Opposition Spaces

Another idea which can enable manipulator kinematic information to inform the grasp search process is the “opposition space.” This is an idea introduced by Iberall [4] that links the form of a human grasp with its function. An opposition space is a set of at least two contacts or virtual fingers that can oppose each other [4]. The contacts in the opposition space need not be fingers or even parts of the hand. For example, humans often hold bulky objects using the sides of their arms. The opposition space is important to this discussion because it links specific contact regions on the hand surface with the role they play in the grasp. We use this mapping to parameterize the grasp controller described in Section 3. The two different opposition spaces we experimented with are illustrated in Figures 2a and 2b.

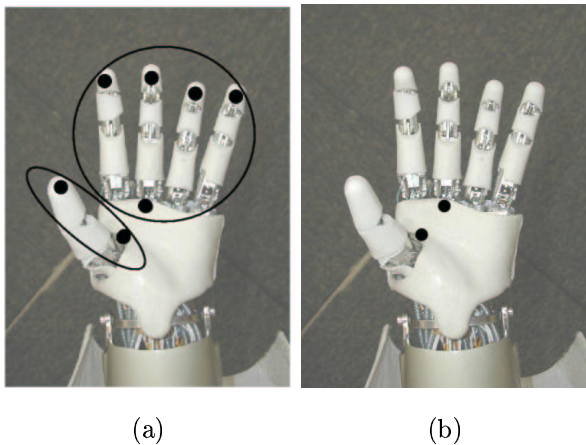


Figure 2: (a) Robonaut hand illustrating Thumb/Fingers opposition. (b) Robonaut hand illustrating palmar opposition. In both diagrams, black dots mark contact surfaces and circles denote virtual fingers.

4.3 Proximal to Distal Opposition Spaces

Grasping using non-fingertip contacts is more difficult than grasping using fingertips for at least two reasons. First, the kinematic coupling of contacts makes independent contact displacement difficult. On the humanoid hand used in our experiments, it was difficult to independently move the palmar and fingertip contacts because of their close proximity in the kinematic chain. Second, the morphology of the hand makes some contact opposition strategies much more effective than others. For example, we have already seen how the ring finger/index finger opposition illustrated in Figure 1b is not very effective.

These two difficulties can be addressed by committing grasp resources in proximal to distal order. The set of contacts and virtual fingers is sorted from proxi-

mal to distal. Moving distally from the proximal extreme, contacts/virtual fingers at approximately the same distance from the kinematic root are organized into opposition spaces. This results in a set of opposition spaces where the opposition is among contacts at approximately the same distance from the root. During search, the most proximal opposition space is used to grasp first. When an optimal grasp is found for this opposition space, the process is repeated for the next most proximal opposition space. This process continues until the most distal contacts are part of the grasp.

As an example of proximal-to-distal grasping, consider a humanoid robot reaching toward a box. Assume that there are three relevant opposition spaces: opposition between palms, opposition between thumb and fingers on each hand, and opposition among the fingers on a single hand. In this case, the grasp processes might work as follows: The robot first grasps the object with two arms, each arm terminated with a single virtual finger. Once the two arms are placed on the object in a satisfactory way, the thumb and fingers are recognised as two separate virtual fingers. These contacts are displaced on the object surface without significantly moving the arms until they too are optimally placed. Lastly, the fingers may be optimized with respect to each other without significantly displacing the hand.

In the next section we report on experiments which demonstrate the effectiveness of a proximal to distal prioritization of opposition spaces composed of virtual fingers.

5 Experiments

We implemented our grasp algorithms on a simulation of the NASA JSC Robonaut [2]. Robonaut is a dextrous humanoid robot designed and built at NASA Johnson Space Center. It is intended to assist humans with extra vehicular activity (EVA) in space. One of the most impressive features of Robonaut is its 12 degree-of-freedom (DOF) hand. The hand, shown in Figures 2a and 2b, closely resembles the human hand and is mounted on a seven DOF arm. We simulated the kinematics of both the Robonaut hand and arm as a single mechanism. In simulating contact with graspable objects, we assume that position and surface normal information is available at each contact point.

One of the main goals of the Robonaut project is to build a humanoid which can perform useful tasks in space. These tasks typically include holding wrenches, drills, or handrails. In each of these cases, an enveloping grasp is more appropriate than a fingertip grasp.

In our experiments, the Robonaut hand executes en-

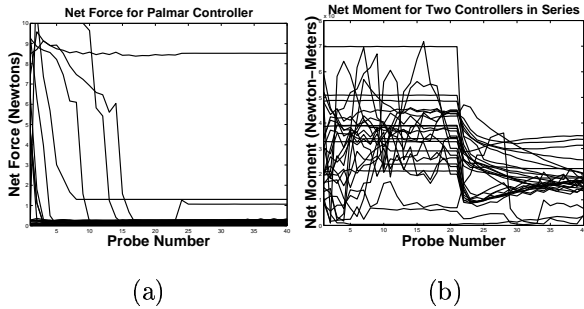


Figure 3: Net force and moment as a function of probe number for 40 trials. Each trial starts with the hand in a random orientation above a 4cm radius cylinder. (a) illustrates the net force behavior of the palmar (proximal) controller. (b) shows net moment for trials where the palmar controller (proximal) runs for the first 20 trials and the thumb-fingers (distal) controller takes over after that.

veloping grasps on cylinders presented in random orientations. In each experiment, results are reported for 40 trials composed of 40 tactile probes each. In order to provide a consistent measure of grasp progress for different grasp controller parameterizations, the results show net force and moment applied by all contacts as a function of probe number.

Three experiments were conducted to evaluate the effectiveness of the proximal to distal approach using opposition spaces composed of virtual fingers. The first experiment uses a proximal opposition space composed of the palmar contacts illustrated in Figure 2b to parameterize the grasp controller in the cylinder grasping task. The second experiment uses the distal opposition space composed of virtual fingers illustrated in Figure 2a for the same cylinder grasping task. The third experiment uses both of these approaches in serial to explore a proximal to distal prioritization of opposition spaces.

5.1 Palmar Opposition

The first experiment investigates the effects of running the grasp controller for the palmar opposition space. The opposition space used includes only two contacts on the palm, as shown in Figure 2b. Since there is no mechanical freedom between the two contacts, the controller can move the contacts only by reorienting the entire hand. This is exactly what happens when the controller is run with palmar opposition. The palm reorients so that the palmar contacts oppose each other as much as kinematically possible. As Figure 3a illustrates, this has the effect of reducing net overall force. By the twentieth probe, the net force for all trials but one has fallen below 1.5 Newtons. This indicates that on all trials but one, the grasp controller with palmar opposition placed the fingers and thumb on opposite sides of the cylinder.

However, the palmar opposition space is less successful in actually opposing thumb and fingers. The first twenty probes shown in Figure 3b illustrate the net moment trajectory for the palmar opposition space. As the first twenty probes show, the palmar opposition space does not significantly reduce net moment.

5.2 Thumb-Fingers Opposition

In order to further reduce net force and moment applied at palmar and distal contacts, the controller must engage the distal contacts in the grasp search process. This is possible when the grasp controller is parameterized with a thumb/fingers opposition space. As Figure 2a illustrates, the contact at the tip of the thumb is grouped with the palmar contact at the base of the thumb to form one virtual finger. The other virtual finger is composed of the palmar contact at the base of the fingers combined with the set of contacts at the fingertips.

The grasp controller parameterized with the thumb-fingers opposition space takes the force and moment applied by the fingers into account when searching for a grasp. This enables the controller to find lower net force and net moment solutions than the palmar controller does. Figure 4b shows net moment for the thumb-fingers (distal) controller for 40 trials. The average net moment converges to a value less than two Newton-Meters. This is a significant improvement over the performance of the palmar (proximal) controller that remains at an average value of four Newton-Meters.

Since the distal controller takes force and moment applied by the fingers into account, we might expect this controller to reduce average net force to a low value as well. Figure 4a shows that this is not the case. There are several trajectories for which net force never falls below two Newtons. Although the dis-

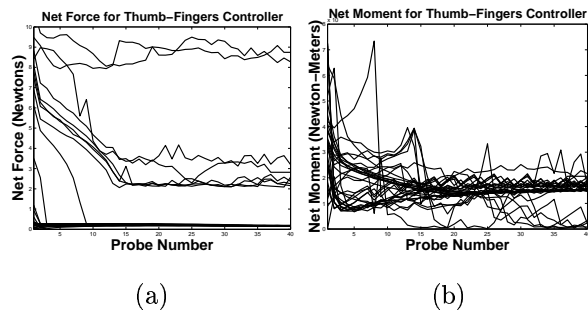


Figure 4: Net force and moment as a function of probe number for 40 trials. Each trial starts with the hand in a random orientation above a 4cm radius cylinder. (a) illustrates the net force for the thumb-fingers (distal) controller. (b) shows net moment for the thumb-fingers controller.

tal controller parameterization is able to find a lower net moment solution, it can fail with a very high error when started with certain incompatible initial configurations.

These failing trajectories are the result of the thumb-fingers controller getting stuck in non-optimal kinematic configurations such as the one shown in Figure 1b. In these situations, the kinematics of the hand are not used in the most effective way. However, because the distal (thumb-fingers) controller follows a greedy gradient toward low error solutions, it is unable to escape these minima.

5.3 Prioritized Combination of Proximal and Distal Opposition

The results of the preceding two experiments indicate that running the grasp controller in proximal and distal opposition spaces have different force and moment results. The proximal controller more effectively reduces net force while the distal controller better reduces net moment. This suggests that some combination of the controllers is needed.

This final experiment runs the proximal controller for the first 20 probes and the thumb-fingers (distal) controller for the last 20 probes. Figure 3b shows the net moment for this experiment. There is an obvious transition from proximal controller to distal controller. The proximal controller does not seem to have a significant effect on moment while the distal controller immediately drives net moment lower. Although not shown, the proximal to distal combination of controllers has a similar effect on net force as that shown for the proximal controller alone in Figure 3a.

The results show that combining the proximal and distal controllers is better than using either controller on its own. The proximal controller serves to “funnel” the state of the system into a region where the distal controller can be effective. Since proximal and distal controllers are executed serially, there is no extra computational cost in the combined approach.

6 Conclusions and Future Work

In this paper, we have shown how grasp controller techniques for grasping with disembodied contacts can be extended to the whole body grasping problem. Whole body grasping greatly increases the number of contact points used to restrain the grasped object and can be more powerful and stable than fingertip grasps. We show that whole body grasping fits neatly into the grasp controller paradigm when grasp controllers are parameterized with virtual fingers and opposition spaces. We show that giving proximal opposi-

tion spaces priority over more distal ones can eliminate undesirable minima in the grasp error function.

This paper studies whole body grasping in the context of a humanoid hand. The next logical step is to verify these techniques with two-armed grasping experiments. We would like to explore further the proximal-to-distal methodology with proximal contact points on humanoid arms and chest.

Acknowledgments

The authors wish to thank Oliver Brock for valuable comments and the members of the Dexterous Robotics Laboratory at Johnson Space Center for their enthusiastic support. This work was supported by the National Science Foundation under grants CISE/CDA-9703217, and IRI-9704530, DARPA MARS DABT63-99-1-0004, NASA/RICIS (University of Houston #215), and NASA GSRP Fellowship.

References

- [1] J. Coelho and R. Grupen. A control basis for learning multifingered grasps. *Journal of Robotic Systems*, 1997.
- [2] M. Diftler, C. Culbert, R. Ambrose, R. Platt, and W. Buethmann. Evolution of the nasa/darpa robonaut control system. In *IEEE Int'l Conf. Robotics Automation*, 2003.
- [3] B. Faverjon and J. Ponce. On computing two-finger force-closure grasps of curved 2d objects. In *IEEE Int'l Conf. Robotics Automation*, 1991.
- [4] C. MacKenzie and T. Iberall. *The Grasping Hand*. North-Holland, 1994.
- [5] V. Nguyen. Constructing stable grasps. In *Int'l Journal of Robotics Research*, 1989.
- [6] R. Platt, A. H. Fagg, and R. A. Grupen. Nullspace composition of control laws for grasping. In *IEEE Int'l Conf. on Intelligent Robots and Systems*, 2002.
- [7] N. Pollard. Parallel algorithms for synthesis of whole-hand grasps. In *IEEE Int'l Conf. Robotics Automation*, 1997.
- [8] J. Ponce, S. Sullivan, A. Sudsang, J. Boissonnat, and J. Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *Int. J. Rob. Res.*, 1996.
- [9] D. Reznik and V. Lumelsky. Multi-finger hugging: A robust approach to sensor-based grasp planning. In *IEEE Int'l Conf. Robotics Automation*, 1994.