A Switching Control Approach to Haptic Exploration for Quality Grasps

Di Wang, Brian T. Watson, Andrew H. Fagg
Symbiotic Computing Laboratory
University of Oklahoma
Norman, OK 73072

***Abstract*— Robotic grasping is traditionally approached as a pure planning problem that assumes *a priori* knowledge of the target object's geometry. However, we would like our robot to be able to robustly grasp objects with which it has no prior experience. Our approach is to use haptic information to drive a search process for appropriate finger contact locations. Given a cost function that is based on the total force and moment applied to the object by the set of contacts, and simple assumptions about the local surface geometry, this search process can be formulated as one of gradient descent on the cost function. Prior work in this area has assumed that the surface of the object local to the contact is either flat or convex. However, when the surface is concave, the search process, in fact, ascends the cost function. Here, we propose a *switching controller* approach that estimates the local curvature of the object over multiple contacts. This information is then used to switch between one of two methods of estimating the gradient of the cost function. While this new approach shows comparable performance to the original when faced with objects containing only flat or convex surfaces, the new algorithm performs substantially better when objects contain concave surfaces.**

I. Introduction

Robotic grasping approaches have traditionally relied on *a priori* knowledge of the geometry of the object being grasped or on estimates of the geometry based on visual or range inputs (e.g., Bekey *et al.*[1], and Borst *et al.*[3], Miller *et al.*[12]). In either case, the modeled geometry is used to assess the quality of many potential grasps before one is selected for execution. In an alternative approach, one could make minimal *a priori* assumptions about the geometry of the object being grasped, and instead rely on haptic feedback to direct the tactile exploration of an object until a suitable grasp is found. Teichmann and Mishra [20], and subsequently Coelho and Grupen [4], introduced methods in which the local surface normal for each of several contacts was first estimated. Based on this information, contact displacements were computed that followed the negative gradient of a cost function. The cost functions were such that their minima corresponded to a quality grasp of the object. In the former case, this cost function was based on the area of the triangle formed by three contact points. In the latter case, two cost functions were used: one that described the net force applied to the object by the set of contacts, and another that described the net moment applied by the same contacts.

Di Wang and Andrew H. Fagg: School of Computer Science, [di|fagg]@cs.ou.edu
Brian T. Watson: School of Mechanical Engineering, btw@ou.edu

In the case of Coelho and Grupen [4], and in subsequent work by Platt *et al.* [15], [14], the gradients of the cost functions were estimated by making simple assumptions about the object's surface properties, including local geometry. First of all, the contacts were assumed to be frictionless. In the case of the force cost function, the surface was also assumed to be locally convex, and in particular, that it was a unit sphere. In the case of the moment cost function, the surface was assumed to be planar. The *force controller* and *moment controller* were defined such that each contact was displaced so that it followed the negative gradient of the corresponding cost function. Platt *et al.* combined the actions of the two controllers through a nullspace operation that favored the actions of the force controller over those of the moment controller.

This work has showed promise in enabling a grasping system to interact with objects of unmodeled geometries [14], [17]. However, the current approach suffers when the object's surface differs substantially from assumption of local convexity. Specifically, when the surface is concave, the force controller will drive the contact in a direction that serves to increase the net force rather than decrease it. This behavior substantially limits the class of objects that can be addressed by this grasp search approach.

The challenge, therefore, is how to appropriately address objects that contain concavities. Park and Starr [13] describe a grasp synthesis method for polygons of known shape that considers as potential contact locations convex and concave vertices. Funahashi *et al.* [6] analyzed the stability of grasps involving fingers with controllable stiffness and either concave object surfaces or fingers. However, they do not address the grasp synthesis step. In this paper, we address the concavity issue by introducing a second force controller that makes the assumption of local concavity. A meta-level controller then switches, on a per-contact basis, between the convex and concave control actions as a function of the estimated curvature of the object. A set of simulation experiments are used to demonstrate the utility of the hybrid convex/concave controller over the convex-only controller.

II. Methods

The goal is to find a set of contact locations on an object such that a set of non-zero normal forces applied at the contacts result in a net force and moment applied to the object of zero. In the presence of soft contacts and friction,

a set of contacts that satisfy these criteria ensures wrench closure, allowing the object to be squeezed at the contacts without accelerating it [18], [16].

Our approach to finding this set of contact locations involves a gradient descent search that requires the fingers to haptically explore the object. At each step of the search, contact is made with the object by each of the finger tips. Given an estimate of each of the contact locations and surface normals, the net force and moment applied to the object is estimated (assuming unit force applied at each contact). Given assumptions about the local geometry of each surface, contact displacements are estimated that are intended to reduce the net force and moment applied to the object. In the subsequent sections, we describe the process of computing these contact displacements.

A. Force Controller

The task of the *force controller* is to reduce the total force applied to the object by the set of contacts. Following Coelho and Grupen [4], the error function for this controller is:

$$\epsilon_f = \frac{1}{2}{\mathbf{f}_n}^T\mathbf{f}_n, \tag{1}$$

where $\mathbf{f}_n$ is a column vector describing the net force applied by the set of contacts. This is computed as follows:

$$\mathbf{f}_n = \mathbf{f} + \mathbf{f}_e, \tag{2}$$

where $\mathbf{f}$ is the unit force applied by the contact of interest, and is normal to the object surface at the contact point, and $\mathbf{f}_e$ is the sum of the remaining forces, including the other contacts and any modeled (but uncontrolled) external forces. Note that one can assume a force vector of arbitrary length to express the differential roles played by various contacts in a grasp. However, we do not exploit this flexibility for the purposes of this work.

Let $\mathbf{x}$ denote the Cartesian location of the contact in R^3. Following each probe of the object, the contact is displaced along a direction that reduces ϵ_f. This direction is computed by first estimating the gradient of the error function:

$$\frac{\partial\epsilon_f}{\partial\mathbf{x}} = \frac{\partial\epsilon_f}{\partial\mathbf{f}}\frac{\partial\mathbf{f}}{\partial\mathbf{x}}, \tag{3}$$

where:

$$\frac{\partial\epsilon_f}{\partial\mathbf{f}} = {\mathbf{f}_n}^T. \tag{4}$$

The gradient of the force with respect to contact location depends on our assumptions about the local surface geometry. For the *convex controller*, we follow the spherical assumption of Coelho and Grupen [4]. This scenario is illustrated in Fig. 1. The current contact produces force $\mathbf{f}_1$. Given that the other contacts produce force $\mathbf{f}_e$, the total force applied to the object is shown as $\mathbf{f}_n$. The finger tip is then displaced along $\Delta\mathbf{x}$ (the tangent to the sphere at the original contact), and then translated toward the origin of the sphere. For a small $\Delta\mathbf{x}$, this latter translation is very small. Therefore:

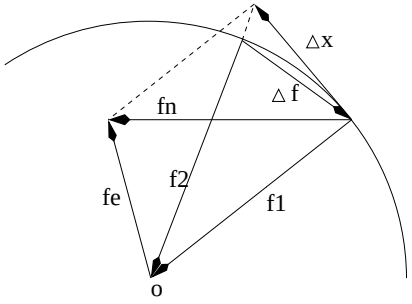


Fig. 1. Force control for convex object. $\mathbf{f}_1$ is the current force exerted by the finger and $\mathbf{f}_2$ is the future force exerted by the finger. $\mathbf{f}_e$ is the external force on the object.

$$\mathbf{f}_2 \approx \mathbf{f}_1 - \beta\Delta\mathbf{x}, \tag{5}$$

where $\beta > 0$ is a stiffness coefficient with units of N/m. The second partial derivative of equation 3 can be approximated as follows:

$$\frac{\partial\mathbf{f}}{\partial\mathbf{x}} \approx \frac{\mathbf{f}_2 - \mathbf{f}_1}{\Delta\mathbf{x}} \approx -\beta\mathbf{I}, \tag{6}$$

where $\mathbf{I}$ is a 3 by 3 identity matrix. Combining equations 3, 4, and 6, we have:

$$\frac{\partial\epsilon_f}{\partial\mathbf{x}} \approx -\beta{\mathbf{f}_n}^T. \tag{7}$$

Finger displacement will follow the negative gradient of the error function. Therefore, the displacement will be in the same direction as $\mathbf{f}_n$, subject to maintaining contact with the object's surface.

The alternative to assuming a locally convex surface is to assume a concave one. For the *concave controller*, we assume that the contact is located on the inside of a unit sphere. This scenario is illustrated in Fig. 2. As with the previous case, we assume a small displacement in the direction of $\Delta\mathbf{x}$ (the surface tangent), and then a small translation back to the surface, toward the origin of the sphere. In this case, the following holds:

$$-\mathbf{f}_2 \approx -\mathbf{f}_1 - \beta\Delta\mathbf{x}. \tag{8}$$

Therefore:

$$\frac{\partial\mathbf{f}}{\partial\mathbf{x}} \approx \frac{\mathbf{f}_2 - \mathbf{f}_1}{\Delta\mathbf{x}} \approx \beta\mathbf{I}. \tag{9}$$

Combining equations 3, 4 and 9, we have:

$$\frac{\partial\epsilon_f}{\partial\mathbf{x}} \approx \beta{\mathbf{f}_n}^T. \tag{10}$$

The result is that the contact should be displaced along the direction opposite to $\mathbf{f}_n$. Note that this displacement should be done subject to maintaining some ideal force against the surface of the object (such that the object itself is not displaced). Note also that the direction of displacement given the concave assumption is in the opposite direction as the convex assumption.

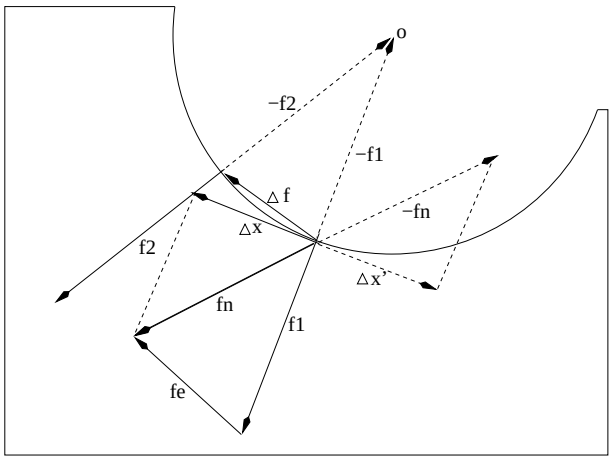


Fig. 2. Force control for concave object. $\mathbf{f}_1$ is the current force exerted by the finger and $\mathbf{f}_2$ is the future force exerted by the finger. $\mathbf{f}_e$ is the external force on the object.

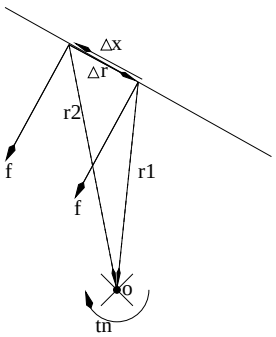


Fig. 3. Moment control. $\mathbf{f}$ is the force exerted by the finger and $\mathbf{f}_e$ is the external force on the object. "o" is the origin about which torque is measured. $\mathbf{r}_1$ and $\mathbf{r}_2$ denote the vector from current and future contact point to the origin, respectively. In current configuration, the net torque points inside the paper.

B. Moment Controller

The role of the *moment controller* is to reduce the total moment applied to the object. The moment control error function is defined as:

$$\epsilon_m = \frac{1}{2}\mathbf{m}_n{}^T\mathbf{m}_n, \tag{11}$$

where $\mathbf{m}_n$ is the net moment applied to the object. Here, net moment is composed of the moment due to the contact of interest ($\mathbf{m}$), and the moment due to any other modeled forces external to the contact of interest ($\mathbf{m}_e$):

$$\mathbf{m}_n = \mathbf{m} + \mathbf{m}_e. \tag{12}$$

The gradient of the error function with respect to contact displacement is as follows:

$$\frac{\partial \epsilon_m}{\partial \mathbf{x}} = \frac{\partial \epsilon_m}{\partial \mathbf{m}}\frac{\partial \mathbf{m}}{\partial \mathbf{x}}, \tag{13}$$

where:

$$\frac{\partial \epsilon_m}{\partial \mathbf{m}} = \mathbf{m}_n{}^T. \tag{14}$$

One can estimate the derivative of the contact-imposed moment with respect to contact displacement by following the planar surface assumption of Coelho and Grupen [4]. This situation is illustrated in Fig. 3. The current contact imposes force $\mathbf{f}$ at a position from which the moment reference point (o) is displaced by $\mathbf{r}_1$. Given a small contact displacement along the surface tangent of $\Delta\mathbf{x}$:

$$\begin{aligned}
\frac{\partial \mathbf{m}}{\partial \mathbf{x}} &\approx \frac{\mathbf{m}_2 - \mathbf{m}_1}{\Delta\mathbf{x}}, \\
&= \frac{\mathbf{f}\times\mathbf{r}_2 - \mathbf{f}\times\mathbf{r}_1}{\Delta\mathbf{x}}, \\
&= \frac{\mathbf{f}\times(\mathbf{r}_2 - \mathbf{r}_1)}{\Delta\mathbf{x}}, \\
&= -\frac{\mathbf{f}\times\Delta\mathbf{x}}{\Delta\mathbf{x}}, \\
&= -\begin{bmatrix} 0 & -f_z & f_y \\ f_z & 0 & -f_x \\ -f_y & f_x & 0 \end{bmatrix}.
\end{aligned} \tag{15}$$

Therefore:

$$\begin{aligned}
\frac{\partial \epsilon_m}{\partial \mathbf{x}} &= -\mathbf{m}_n{}^T\begin{bmatrix} 0 & -f_z & f_y \\ f_z & 0 & -f_x \\ -f_y & f_x & 0 \end{bmatrix} \\
&= -\mathbf{f}\times\mathbf{m}_n.
\end{aligned} \tag{16}$$

The result is that the moment controller recommends contact displacements in the direction of $\mathbf{f}\times\mathbf{m}_n$. Note that in the general case, the choice of the moment reference point can affect the magnitude and sign of this recommended displacement. However, when the net force due to the set of contacts is zero, they form a *couple*. Under this condition, the net moment (and hence, the recommended displacement) becomes the same for any choice of reference point [11].

C. Concavity Detection

Key to making use of the appropriate force control law (the convex or the concave form) is estimating the local surface curvature. One possible approach is to recover local curvature using a tactile or image array (e.g., Jiar *et al.* [9] and Lee & Nicholls [10]). In our case, we assume that a 6-axis load cell is embedded within the finger tip, from which one can infer the surface normal but not the local curvature [2]. Hence, it becomes necessary to integrate information over multiple probes of the object. At any given time, for the purposes of the force controller, a surface is assumed to be either concave or convex (with surfaces initially assumed to

be convex). Given two subsequent probes of an object, the assumed curvature is determined as follows:

$$\begin{aligned} \Delta\mathbf{f}^T\Delta\mathbf{x} < -\alpha_1 \quad & \text{assume convex} \\ \Delta\mathbf{f}^T\Delta\mathbf{x} > \alpha_2 \quad & \text{assume concave} \\ otherwise \quad & \text{no change} \end{aligned}$$

where $\Delta\mathbf{f}$ is the change the force exerted by the contact, $\Delta\mathbf{x}$ is the displacement of the contact, and $\alpha_1 > 0$ and $\alpha_2 > 0$ are switching parameters. These parameters are empirically selected so that the switching controller is sensitive enough to discover the concavities of the test objects, but is not distracted substantially by uncertainties in the force estimation process (both are set to $0.5\ Nm$).

D. Combining Force and Moment Control Actions

Following Platt *et al.*, the displacement recommendations of both the force controller and the moment controller are combined into a single displacement [15], [14]. This combination gives the moment controller the opportunity to influence the motion of the contact as long as it does not interfere with the actions of the force controller [8], [7]. Specifically, the combined control action is computed as follows:

$$\phi \equiv \phi_f + N(\phi_f)\phi_m, \tag{17}$$

where ϕ is the composite displacement, ϕ_f and ϕ_m are the recommended displacements by the force and moment controllers, respectively, and $N(.)$ is a null space projection matrix.

E. Switching Controller

The sequential behavior required for the haptic search process is implemented using a finite state machine, as shown in Fig. 4. Each finger is controlled using one such FSM. The object is initially assumed to be located such that flexing of the the fingers toward the palm will bring each finger into contact with the object. At each control step, the surface normal at the contact is used to estimate the local curvature of the surface and to recommend a displacement of the contact. The latter is computed using the combined force and moment controller recommendations (as described above), where the force controller is either of the type *convex* or *concave*. Displacement of the contact is implemented by drawing the finger away from the surface along the normal, translating the finger along the recommended displacement, and then placing the finger back onto the surface along the original surface normal. Under most conditions, the finger will again contact the surface, and the process will repeat from either the concave or convex states.

The search process terminates under two conditions. First, if the net force and moment fall below a critical threshold, the search is considered to have terminated successfully (the *Done* state). Second, if the search is still ongoing after a specified period of time, the controller is considered to have terminated in an *Error* state (we say that the controller has "timed out"). One scenario in which this can happen is when

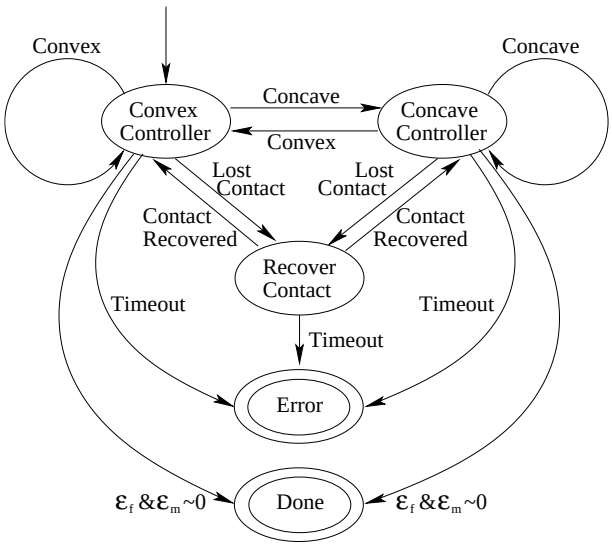


Fig. 4. Finite state machine for the controller with concavity detection

the controller has achieved an equilibrium state or is in a cycle in which the force or moment error does not satisfy the critical threshold.

There are also a number of cases in which the finger does not successfully return to surface of the object after displacement is executed. This situation can happen if the local surface is not smooth with respect to the size of the contact displacement. For example, this can happen when the contact is displaced off the corner of an object. This situation can also happen when there is an error in the estimation of the surface normal. Here, the recommended displacement can take the finger far away from the surface. In either case, the *recover contact* state is responsible for taking several heuristic approaches to bringing the finger back into contact with the object. Both problems are sensitive to the magnitude of the contact displacement for a single control step. Therefore, contact displacement magnitude is considered a parameter that must be selected appropriately (this is addressed in the *Results* section).

F. Simulation Implementation

All experiments were conducted within a custom simulation environment that is based on the Visualization Toolkit (VTK) package [19]. This toolkit allows for the modeling of the surfaces of the robotic fingers and of the objects of interest. The surfaces are modeled as meshes; collision between two surfaces is detected as a collision between two mesh triangles. The force direction (and hence surface normal) is then determined by the orientation of the colliding triangles. Due to this approach, estimation of applied force exhibits a certain degree of variation, depending on small changes in orientation of the colliding triangles.

The simulated robot used for the experiments described in this paper consists of two fingers. Each finger is four degrees of freedom: one adduction/abduction and three flexion. The fingers are mounted relative to one another such that they are able to oppose one-another. For these experiments, the

hand is fixed in space. Hence, any objects are placed near the palm of the hand such that both fingers can initially touch the object through a simple flexion motion.

III. Experimental Results & Evaluation

The proposed controller is explicitly designed to properly explore objects with both convex and concave surfaces. Our experimental hypothesis is that when presented with an object containing a concave surface, the proposed controller should perform better than one controller that only "expects" convex surfaces. However, when an object is composed of non-concave surfaces (convex or flat), the two controllers should perform identically.

In addition to the choice of control algorithm, two other factors can significantly affect the performance of a controller: the contact displacement magnitude and orientation of the object relative to the fingers. The latter factor affects which of the object's surfaces are initially exposed to the fingers. We have therefore implemented (for most of the following experiments) a 3-factor analysis of algorithm, displacement magnitude, and object orientation. For each combination of factors, a sample size of $N = 20$ is used. Although object orientation is a controlled factor, a small, random orientation is added to the object prior to initiation of the exploration process (uniform distribution: ± 5 degrees). Two measures of performance were used: the total number of successful grasps (achieving the *done* state), and the running time of the controller (a controller "times out" when a probe occurs 400 seconds into the search).

The implementation of the convex-only controller is the same as the switching controller (as described in Fig. 4). The only difference is that the *concave controller* state has been removed (and consequently, concavity detection is not performed).

A. Non-Concave Objects

Two non-concave objects were used in this study: a sphere and a cube. Because the sphere is symmetric about all rotations, we used a two-factor analysis (controllers ($\times 2$) and contact displacement magnitude ($\times 16$)). Fig. 5 shows the percent of successful grasps for each algorithm and contact displacement magnitude ("step size"). Both controllers performed well when step size was less than 10, but performance degraded with larger step sizes. This degradation is due to the fact that the larger step sizes tended to lead to instabilities around the equilibrium point. In addition, the larger step sizes could result in the finger "dropping off" of the object all together.

Across all step sizes, the mean success for both controllers was 80%. Furthermore, a two-way ANOVA analysis on the total running time of the controller indicated a significant effect of step size ($F = 40.56$; $p < 10^{-4}$), but no significant effect on choice of controller ($F = .17$; $p > .68$). There was, however, a significant interaction effect ($F = 6.79$; $p < 10^{-4}$). This effect highlights the fact that the switching controller performed slightly worse for small step sizes. This effect is due to the controller's inability to consistently

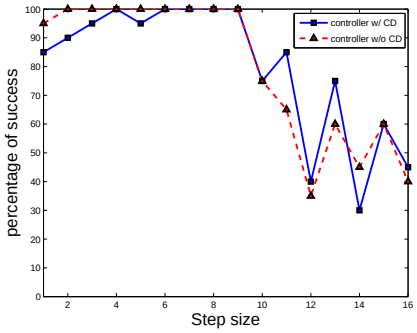


Fig. 5. Successful grasps by the switching controller and the convex-only controller when grasping the sphere.

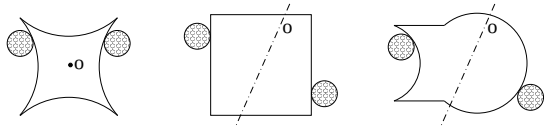


Fig. 6. Cross-section of three of the four objects: "concave cube", cube, and capped cylinder. (O is the rotation axis used in the experiments)

estimate the curvature of the object when displacements are small. Specifically, the "noise" in the contact normal estimation can sometimes dominate the curvature estimate in these cases.

The second non-concave object was a cube (Fig. 6). Across all step sizes and object orientations, the convex-only controller was successful 76.6% of the time, while the switching controller was successful for only 73.6% of the trials. This difference was not significant according to a Fisher Exact Test ($p > 0.09$). A three-way ANOVA test on running time showed a significant effect on the average running time of step size ($F = 29.4$; $p < 10^{-4}$) and on choice of controller ($F = 6.78$; $p < .01$). Mean running time for the convex-only controller was $204.75sec$, while the switching controller required a mean of $219.32sec$. The small additional cost in running time of the proposed controller was due to occasional misclassification of the surface curvature. This effect typically results in one or two control steps moving in the incorrect direction before the surface is correctly classified.

B. Concave Objects

Two objects containing concavities were used: a "concave cube," and a "capped cylinder" (see Fig. 6). The concave cube was constructed by subtracting hemispheres from each side of a cube. The result was a that a majority of its surface was concave, with only areas around the corners maintaining a flat surface. Most of the surface of the capped cylinder is convex, but one small region is concave. Depending on the orientation of the cylinder relative to the fingers, the initial contacts may "see" either both convex surfaces or one concave and one convex surface.

Fig. 7 shows the performance of each controller given the step size and the object orientation. Orientations of 75 and 90

degrees are such that both fingers are initially presented with a convex surface. For orientations of 0 and 15 degrees, one finger falls well within the concave region. An orientation of 45 degrees results in one finger landing on the edge between the concave and convex surface (with 30 and 60 degrees falling to either side of this sharp edge).

Both controllers exhibited high levels of success for orientations of 60, 75, and 90 degrees (Fig. 7a,b). However, the success of the convex-only controller suffers dramatically when presented with the concave surface (orientations of 0, 15, and 30 degrees). In contrast, the switching controller (7b) performed well for these orientations, most dramatically for 0 and 15 degrees. However, this performance dropped off as the step size increased beyond 8.

Across all step sizes and object orientations, the switching controller was successful 75.5% of the time, while the convex-only controller was successful only 49.5%. This difference was significant according to a Fisher Exact Test ($p < 10^{-4}$). A three-way ANOVA test on the controller running time showed a significant effect on the average running time of step size ($F = 15.3$; $p < 10^{-4}$) and of choice of controller ($F = 538$; $p < 10^{-4}$). Mean running time for the convex-only controller was $258.7sec$, while the switching controller required a mean of $201.6sec$.

In the concave cube case, across all step sizes and object orientations, the switching controller was successful 56.6% of the time, while the convex-only controller was successful only 8.67% of the time. This difference was significant according to a Fisher Exact Test ($p < 10^{-4}$). The step size also had a significant influence on the success rate, according to a Chi-squared test (for the switching controller, χ^2=342.4, $p < 10^{-4}$; for the convex-only controller, χ^2=29.1,p<.016).

C. Varying Object Size

One critical question to ask is the degree to which object size influences the outcome of the search process. To address this issue, we presented concave cubes that were 50% larger and 50% smaller than the original size reported above. For the large cube, across all step sizes and object orientations, the switching controller was successful 53.8% of the time, while the convex-only controller was successful only 6.41% of the time. This difference was significant according to a Fisher Exact Test ($p < 10^{-4}$). For the smaller cube, the switching controller was successful 52.7% of the time, while the convex-only controller was successful 31.6% (the difference also significant according to a Fisher Exact Test; $p < 10^{-4}$). The latter controller demonstrated a marked improvement in performance over the other object sizes. This improvement was due in large part to a fortuitous alignment between the edge of the cube and the initial positions of the fingers. The alignment enabled the convex-only search process to discover a grasp solution in which the flat regions around the corners of the cube were used in the final solution, rather than the concavity.

D. Randomized Object Poses

In the above experiments, each object was positioned in a fixed location and a fixed set of rotations were used (about a single axis). In order to examine the robustness of our controller over a wider range of initial conditions, we also performed a set of experiments in which the pose of the object was chosen randomly. We chose the capped cylinder and the concave cube with a range of controller step sizes (6,7,8, and 9). The object position is selected from a normal distribution (standard deviation: 15% of finger time radius) and the orientation is selected uniformly from the possible 3D orientations.

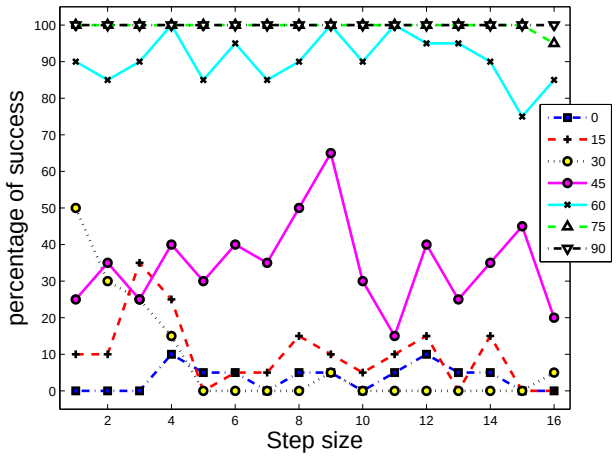
For each of the two objects, both the convex-only and the switching controllers were employed. A total of 100 samples were taken for each of the four conditions. For the cylinder, the switching controller achieved 86% success, where the convex-only controller achieved 75%. For the concave cube, the switching controller achieved a success rate of 49.3%, while the convex-only controller achieved a rate of 9.5%. For this latter object, of the failures to grasp, more than 70% were cases in which an equilibrium was successfully reached, but either the net force or moment (or both) were above the acceptable threshold (e.g., grasping a cube using two fingers on the same corner).

IV. Conclusions and Future Work

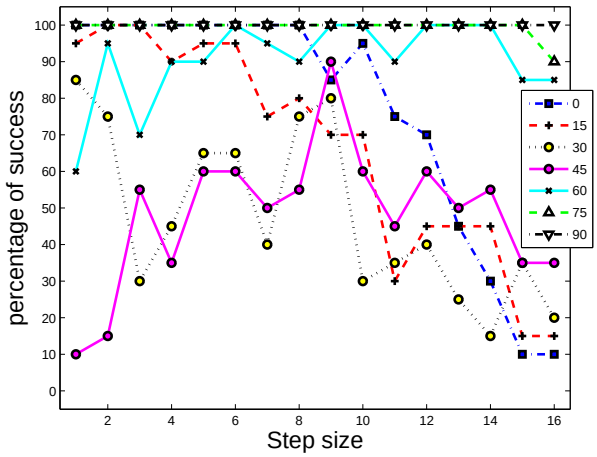
When grasping and manipulating objects in open environments where uncertainties exist in their geometry and pose, robust behavior requires the on-line integration of sensory data. In particular, one can make use of haptic information to guide the search for contact locations that will comprise a stable grasp. This search process can be formulated as one of gradient descent of a wrench-based error function. However, estimation of the gradient of this error function with respect to contact movement relies on knowledge of the curvature of the surface around the contact point.

In this paper, we have proposed an approach in which the local curvature is estimated to be either convex or concave using a pair of spatially distributed probes over the object. Given this estimate, the gradient following algorithm can appropriately assess the direction in which to displace the contact in order to further reduce error. For objects that include concave surfaces, our experimental results show a substantial performance increase by the proposed algorithm over an algorithm in which the surface is always assumed to be convex. For convex objects, although the proposed algorithm pays a small performance cost, it is not substantial. These results hold for a range of parameter settings, including the magnitude of the search step size.

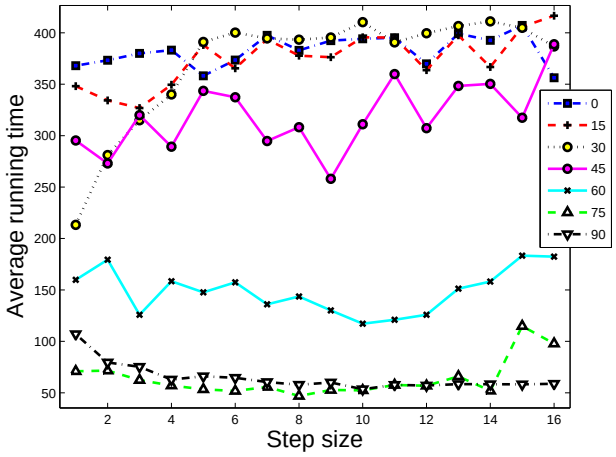
Because the search for an appropriate grasp configuration follows the gradient of the cost function, the approach suffers from the existence of local minima. This was particularly clear in case of the concave cube, in which the proposed controller failed on 43.4% of the trials, 70% of which were cases in which the two fingers were left attempting to grasp a single corner of the cube. In general, the existence of these local minima is an interaction of the geometry of the



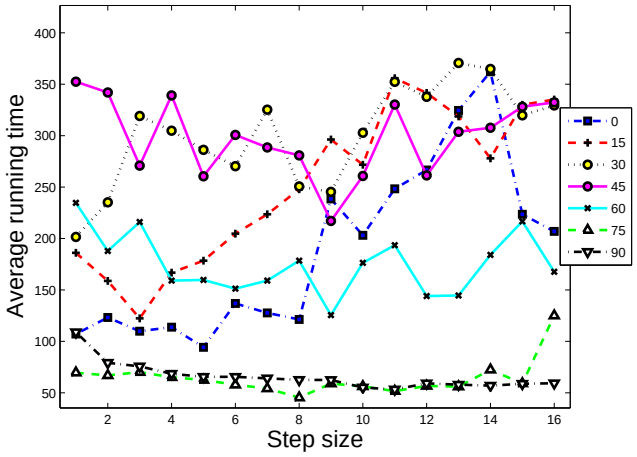
(a) Percentage of Success for the Convex-only Controller



(b) Percentage of Success for the Switching Controller



(c) Average Running Time for the Convex-only Controller



(d) Average Running Time for the Switching Controller

Fig. 7. Performance of the convex-only and the switching controllers on the capped cylinder.

object, and the number of contacts involved in the grasping process. One can address this issue simply by detecting that a local minima has been found, and then restarting the search process from a new configuration. In this restart step, one could also remove or add fingers, and hence contacts.

More generally, it is possible to introduce vision-based techniques to bias the selection of the initial configuration of the fingers. Specifically, we would want such a configuration to drop the contacts into a well of the error function whose minimum corresponds to an acceptable grasp. Along these lines, Coelho, Piater, and Grupen [5] have developed an approach in which visual representations are explicitly acquired that robustly predict the outcome of a grasp search process. Once the models have been learned, it becomes possible to use the visual representations in a novel situation to place the hand and fingers in a configuration that is often near to an acceptable grasp. We are currently working to extend this approach to allow grasping of three dimensional objects.

V. Acknowledgments

This material is based upon work supported in part by the National Science Foundation under Grant No. 0453545, and by the University of Oklahoma.

References

[1] G. A. Bekey, H. Liu, R. Tomovic, and W. J Karplus. Knowledge-based control of grasping in robot hands using heuristics from human motor skills. *IEEE Transactions on Robotics and Automation*, 9(6):709–722, 1993.

[2] A. Bicchi, J. Salisbury, and D. Brock. Contact sensing from force measurements. *International Journal of Robotics Research*, 12(3):249–262, 1993.

[3] Ch. Borst, M. Fischer, and G. Hirzinger. A fast and robust grasp planner for arbitrary 3d objects. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 1890–1896, Detroit, MI, USA, May 1999.

[4] J. A. Coelho, Jr. and R. A. Grupen. A control basis for learning multifingered grasps. *Journal of Robotic Systems*, 14(7):545–557, 1997.

[5] J. A. Coelho, Jr., J. Piater, and R. A. Grupen. Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot. *Robotics and Autonomous Systems Journal, special issue on Humanoid Robots*, 37(2–3):195–219, November 2000.

[6] Y. Funahashi, T. Yamada, M. Tate, and Y. Suzuki. Grasp stability analysis considering the curvatures at contact points. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3040–3046. IEEE, April 1996.

[7] M. Huber. *A Hybrid Control Architecture for Adaptive Robot Control*. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, 2000.

[8] M. Huber and R. A. Grupen. A hybrid discrete event dynamic systems approach to robot control. Technical Report 96-43, University of Massachusetts Amherst, Department of Computer Science, October 1996.

[9] Y. Jiar, X. Wu, Z. Zhao, and K. Li. A sensory robotic grasping system for object recognition and shape recovery. In *Proceedings of the IEEE International Conference on Industrial Technology*, pages 333–337, Guangzhou, China, December 1994. IEEE.

[10] M. Lee and H. Nicholls. Tactile sensing for mechatronics: A state of the art survey. *Mechatronics*, 9:1–31, Jan 1999.

[11] M. T. Mason. *Mechanics of Robotic Manipulation*. MIT Press, 2001.

[12] A.T. Miller, S. Knoop, H.I. Christensen, and P.K. Allen. Automatic grasp planning using shape primitives. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1824–2829, 2003.

[13] Y. C. Park and G. P. Starr. Grasp synthesis of polygonal objects. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1574–1580. IEEE, May 1990.

[14] R. Platt, Jr. *Learning and Generalizing Control-Based Grasping and Manipulation Skills*. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, 2006.

[15] R. Platt, Jr., A. H. Fagg, and R. A. Grupen. Nullspace composition of control laws for grasping. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS'02)*, 2002.

[16] R. Platt, Jr., A. H. Fagg, and R. A. Grupen. Manipulation gaits: Sequences of grasp control tasks. In *Proceedings of the International Conference on Robotics and Automation (ICRA'04)*, pages 801–806, April 2004.

[17] R. Platt, Jr., R. A. Grupen, and A. H. Fagg. Learning grasp context distinctions that generalize. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 2006.

[18] J. Ponce, S. Sullivan, A. Sudsang, J.-D. Boissonnat, and J.-P. Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *International Journal of Robotics Research*, 16(1):11–35, 1997.

[19] W. Schroeder, K. Martin, and B. Lorensen. *Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Kitware, Inc., 4 edition, 2006.

[20] M. Teichmann and B. Mishra. Reactive algorithms for 2 and 3 finger grasping. In *Proceedings of the 1994 International Workshop on Intelligent Robots and Systems*, 1994.