

Estimating Part Tolerance Bounds Based on Adaptive Cloud-Based Grasp Planning with Slip

Ben Kehoe¹ Dmitry Berenson² Ken Goldberg²

Abstract—We explore setting bounds on part tolerances based on an adaptive Cloud-based algorithm to estimate lower bounds on achieving force closure during grasping. We consider the most common robot gripper: a pair of thin parallel jaws, and a conservative class of push-grasps allowing slip that can enhance part alignment for parts that can be modeled as extruded polygons. The grasp analysis algorithm takes as input a set of candidate grasps and perturbations of a nominal part shape. We define a grasp quality metric based on a lower bound on the probability of achieving force closure. We present two extensions to our previous highly-parallelizable algorithm that adaptively reduce the number of grasp evaluations and improve the lower bound by including slip. We develop a procedure for finding the effect of increasing tolerance in vertices on grasp quality, which allows part tolerances to be bounded to ensure minimum grasp quality levels. We find that including slip improves grasp quality estimates by 16%, and our adaptive extension reduces grasp evaluations by 91.5% while maintaining 92.6% of grasp quality.

I. INTRODUCTION

One of the primary focuses of automation is ensuring quality and reliability through structured environments. Even in a structured environment, there are errors and tolerances. This paper describes a method for finding bounds on tolerances using a Monte Carlo method to analyze the effect of part tolerances on grasps generated by an adaptive grasp planning algorithm.

Once the effect of a given tolerance on grasping is computed, the process can be reversed to find the highest tolerance that can provide acceptable results. Computing tolerances based on sampling can be time-consuming, and in manufacturing, computations may be required to execute within a fixed time interval. A grasp algorithm can only be used in this situation if it works within that limit, and will perform better if it can balance the evaluation of more grasps against the detail of the grasp candidate analysis.

Cloud Computing is a powerful new paradigm that expands on networked robotics [1] for massively parallel computation and real-time sharing of vast data resources. The parallel aspect of the Cloud can be used to facilitate grasping parts with shape tolerances. A fundamental challenge, even with perfect recognition, is variation in part shape, because of manufacturing constraints, and mechanics, because of limits on sensing during grasping.

¹Department of Mechanical Engineering, University of California, Berkeley; Berkeley, CA 94720, USA; benk@berkeley.edu

²Department of Industrial Engineering and Operations Research and Department of Electrical Engineering and Computer Science; University of California, Berkeley; Berkeley, CA 94720, USA; berenson@eecs.berkeley.edu, goldberg@berkeley.edu

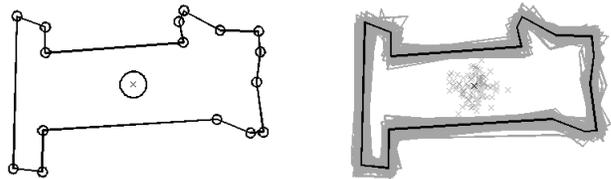


Fig. 1. Part tolerance model. On the left, circles with a radius of one standard deviation of a circular Gaussian distribution are drawn around each vertex and the center of mass. On the right, the nominal part is plotted over 100 sampled perturbations (shown in gray).

This paper describes new work building on [21] that introduced a method that leverages Cloud Computing to analyze grasps with part tolerances. We take a conservative approach: we use a statistical sample of part shape perturbations to find the value of a quality metric that estimates a lower bound on the probability of force closure for a class of grasps called *conservative-slip push grasps*, which can be rapidly evaluated without simulation. In [21], we restricted the class of grasps to *zero-slip push grasps*. In this paper, we extend that class to include a slip model. We then combine the results of the retained candidate grasps, weighting their success on a given part perturbation by the probability of that perturbation, to estimate a lower bound on the probability of achieving force closure.

We improve upon the grasp planning in [21] by adaptively reducing the candidate grasp set after testing a small number of part perturbations, reducing the overall number of grasp evaluations. Taking the best grasp from this method, we increase the tolerance on the part to find how the quality of the grasp decreases. Tolerance bounds are found where the quality falls below a threshold.

We analyze our extensions from previous work, finding that conserve-slip push grasps generate more robust and more intuitive results than zero-slip push grasps, with an average quality 16% higher, and that adaptive sampling can reduce the number of grasp evaluations performed by 71.5% without reducing grasp quality, or reduce grasp evaluation by 91.5% while maintaining 92.6% of grasp quality.

II. RELATED WORK

In “Algorithmic Automation” [16], abstractions can allow the functionality of automation to be designed independent of the underlying implementation and can provide the foundation for formal specification and analysis, algorithmic design, consistency checking and optimization. Algorithmic Automation thus facilitates integrity, reliability, interoperability, and maintainability and upgrading of automation.

Several studies use contact sensors to improve grasp quality in the presence of uncertain part geometry [14] [19]. However, many robotic grippers do not have contact sensing capability. Sensing is often implicitly assumed to be present, such as when pinch grasps are required, since the part must not be moved by contact with the gripper [9] [22] [34] [35].

Studies have explored properties of polygonal parts for grasping [7] [8] [12], but focus on point grasps, which ignores the complex interaction created by a gripper of nonzero width, as is the case with parallel-jaw grippers.

Push manipulation of parts has been extensively investigated by Mason [25] and others [2] [24]. Performing pushing operations with a gripper to reduce pose uncertainty has been demonstrated by Dogar and Srinivasa [13]. However, these methods, again, do not take into account part shape tolerance.

Similarly, many recent studies in robotic grasping focus on improving grasps on known parts [31] [32] [33] that do not take into account tolerances. The work in robotic grasping that addresses tolerance largely focuses on part pose [4] [13] [29]. Methods for sensorless part orientation [5] [17] [37] can also be used in the presence of uncertain part pose. However, these methods do not take into account tolerances for the geometry of the part.

While networked automation has a long history, only recently has research focused on networked robots sharing information to accomplish tasks widely separated in time and space [26] [36]. The introduction of Cloud Computing can allow computation to be offloaded from robots [3], as well as development of databases that allow robots to reuse previous computations in later tasks [11]. Grasping could benefit from this effort, since grasps computed for a part can be applied to similar parts encountered later [10] [15] [18]. This allows the construction of grasp databases that can be shared and referenced by multiple robots [18] [23].

An explicit part tolerance model for grasping was proposed by Christopoulos and Schrater [9] that approximates the part boundary with splines but does not account for motion induced by contact from the gripper. Models exist for tolerance [6] [20] that use worst-case bounds rather than probability distributions. Other work defines topological tolerance models but does not apply it to grasping [30].

III. PROBLEM STATEMENT

We consider a parallel-jaw gripper, gripping a part from above. We assume that we have a conservative estimate of the coefficient of friction between the gripper and the part, denoted μ .

We assume that the part can be modeled as an extruded polygon to be gripped on its edges, resting on a planar work surface, and that the part has an estimated nominal center of mass, which may not be at the centroid. The gripper-part interaction is assumed to be quasistatic, such that the inertia of the part is negligible [28].

A. Part Tolerance Model

Part shape tolerances are modeled as independent, Gaussian distributions on each vertex and center of mass, centered

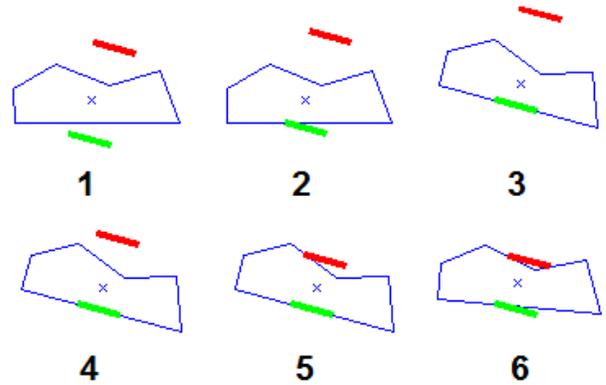


Fig. 2. Snapshots of the execution of a conservative-slip push grasp. The green jaw makes the first contact, and once a stable push is established in frame 3, the red jaw closes. After making contact in frame 5, the part rotates into slip closure in frame 6.

on their nominal values, as shown in Figure 1. The variance of the distributions is an input, which may be dictated by manufacturing constraints. One advantage of using probability distributions is that we can use a Monte Carlo approach to evaluate the effect of higher tolerances on candidate grasps.

B. Conservative-Slip Push Grasps with Force Closure

We consider a class of push grasps that enhance part alignment, *conservative-slip push grasps with force closure*. We define this as grasps in which the gripper pushes the part without slipping until it rotates into alignment with the first gripper jaw (a *zero-slip push*), or slips but is guaranteed to align with the edge (a *conservative-slip push*) and then completes force closure with the second gripper jaw, as seen in Figure 2. Under this conservative definition, we include slip of the second gripper jaw under limited conditions described in Section IV-A.3.

The input to the algorithm is a list of edges defining a non-intersecting polygon, denoted S_0 , and the variance of the Gaussian tolerance distributions for the vertices and center of mass.

C. Quality Measure

We define a quality measure $Q(g, S, \theta)$ as a lower bound on the probability that grasp g on part S will result in force closure based on parameter vector θ . The output of the algorithm is $\mathcal{Q} = \{Q(g, S, \theta) \mid g \in G(S)\}$, where $G(S)$ is the set of candidate grasps for part S . The best grasp and Q-value are:

$$g^* = \arg \max_{g \in G} Q(g, S, \theta)$$

$$Q^*(S, \theta) = Q(g^*, S, \theta)$$

The adaptive version of our algorithm may reduce the value of Q^* relative to the non-adaptive version. The value of Q^* as found by the non-adaptive algorithm is denoted Q_{max}^* , and the normalized value of Q^* for the adaptive version is $\hat{Q}^* = Q^*/Q_{max}^*$.

D. Candidate Grasp Configuration Space

The configuration space is defined by a starting position and orientation of the first gripper jaw, and a direction of motion from this position. We assume that orientation of the gripper jaw face is perpendicular to the direction of motion.

We reduce the configuration space from three dimensions to two using *nominal contact points* to eliminate some of the redundancies in grasp configurations. Candidate grasps (denoted g_{ij}) are defined as the ordered pair (\hat{p}_i, ϕ_j) , where \hat{p}_i is a *nominal contact point* and ϕ_j is an *approach angle*. A nominal contact point is the point on the boundary of the nominal part that contact by the first gripper jaw would occur. The set of nominal contact points is denoted \hat{P} . An approach angle is an angle from the normal direction of the part boundary (pointing into the part) at \hat{p}_i . The *approach line* is the line through \hat{p}_i along ϕ_j .

The actual initial contact point for a candidate grasp g_{ij} on a perturbed part may not be near the nominal contact point \hat{p}_i , as some part of the first gripper jaw may contact a different point on the part first. The actual contact configuration corresponding to candidate grasp g_{ij} on part S is $c_{ij,S} = (p, \psi)$, where p is the position of the gripper and ψ is the direction of the jaw relative to the x -axis. Different candidate grasps may have similar contact configurations; for example, if they approach the same edge of the part. Configurations are grouped into sets of similar configurations denoted $C_{q,\psi}$, where ψ is the same for all configurations in a set and q is a position that describes the set, e.g., one of the p values in the set. The set of similar configuration sets is denoted \mathcal{C} .

IV. GRASP ANALYSIS AND PLANNING ALGORITHMS

Our grasp analysis algorithm, shown in Algorithm 1, calculates the quality metric for a set of grasps and part perturbations. For each part perturbation, the candidate grasps are evaluated to estimate if they result in conservative-slip pushes (see Section IV-A.1). The successful conservative-slip pushes are grouped into sets of similar configurations (see Section IV-A.2), and conservative conditions for force closure are evaluated. Finally, the overall probability of achieving force closure for each candidate grasp is estimated.

Our grasp planning algorithm, shown in Algorithm 2, uses the analysis algorithm on a part using a Monte Carlo method: it generates a set of candidate grasps, and creates part perturbations drawn from the distribution. These grasps and perturbations are passed to the analysis algorithm.

The analysis algorithm uses a single parameter, the grasp elimination criterion R , which is used in adaptively reducing the candidate grasp set. The planning algorithm also uses several additional parameters, denoted as the vector $\theta = [d_C, \rho, \Phi, \mathcal{M}, R]$. The part tolerances for the vertices and center of mass described in Section III-A are also parameters. Three parameters are used for generation of candidate grasps. A filtering parameter d_C and a configuration density parameter ρ are used to determine the set of candidate grasp positions, and the set of candidate grasp orientations is a third parameter, denoted Φ . The algorithm iteratively tests part perturbations; the number of iterations and part

Algorithm 1: Grasp Analysis Algorithm. Highlighted line numbers indicate parallelizable steps.

```

Input: candidate grasp set  $G_1$ , part perturbations
 $S_1, S_2, \dots, S_M$ ;
1 for Part perturbation set  $\mathcal{S}_m = S_1, S_2, \dots, S_M$  do
2   for Part  $S_k = S_1, S_2, \dots, S_l \in \mathcal{S}_m$  do
3     for Candidate grasp  $g_{ij} \in G_m$  do
4       Estimate if  $g_{ij}$  results in conservative-slip
       push of  $S_k$ ;
5       end
6       Collect similar push configurations  $\mathcal{C}$ ;
7       for Similar configuration set  $C_{q,\psi} \in \mathcal{C}$  do
8         Estimate regions of force closure success on
9          $S_k$ ;
10        for Contact configuration  $c_{ij,S_k} \in C_{q,\psi}$  do
11          Predict force closure success
12           $s_{ijk} \in \{0, 1\}$  of  $g_{ij}$  for  $S_k$ ;
13        end
14      end
15    end
16  Produce grasp set  $G_{m+1}$  by removing low-quality
  grasps from  $G_m$  according to parameter  $R$ ;
17 end
18 for Candidate grasp  $g_{ij} \in G_1$  do
19   Compute grasp quality  $Q(g_{ij}, S_0, \theta)$ ;
20 end

```

Algorithm 2: Grasp Planning Algorithm. Highlighted line numbers indicate parallelizable steps.

```

1 Filter  $S_0$  into  $S_C$ ;
2 Determine nominal contact points  $\hat{P}$  on  $S_0$  using  $S_C$ ;
3 Create candidate grasp set  $G_1$  from  $\hat{P}$  and  $\Phi$ ;
4 Create part perturbations  $S_1, S_2, \dots, S_N$  of  $S_0$ ;
5 Compute quality of candidate grasps  $\mathcal{Q}$  using
  Algorithm 1;

```

perturbations in each iteration is set by the parameter \mathcal{M} , where $M = |\mathcal{M}|$ is the number of iterations, and \mathcal{M}_i is the number of perturbations tested in iteration i . The total number of part perturbations is $N = \sum_i \mathcal{M}_i$. The final parameter is the grasp elimination criterion for the grasp analysis. We describe these parameters and each step of our algorithms below.

A. Evaluating Part Perturbations

For each part perturbation in a part perturbation set, the candidate grasps are evaluated to estimate whether they achieve conservative-slip push grasps with force closure.

1) *Conservative-Slip Push Conditions*: The algorithm uses geometric properties of the part to determine all candidate grasps resulting in conservative-slip pushes aligned with a part edge for a given gripper width.

The conditions for success of a zero-slip push are as follows: the part purely rotates about the contact point without slipping, the part rotates towards stability with the

gripper jaw (that is, the edge rotates toward alignment with the gripper), and once the gripper has two points of contact, the center of mass must be between these points.

For a conservative-slip push, the gripper must be guaranteed to align with the initially-contacted edge. As shown by Mason [25], the motion of a part pushed at a given contact point is determined by the friction cone and the direction of pushing. The resulting constraint on candidate grasps is shown in Figure 3. In the conservative-slip regions, the motion of the gripper is guaranteed to be towards a 0 angle and the center point along the edge. Therefore, the configuration of the gripper as it slips must stay in the region or enter the zero-slip region, in which case a zero-slip push occurs. If the gripper becomes aligned without entering the zero-slip region, the gripper is guaranteed to cover the center of mass, so a successful push occurs. Because the slip analysis does not predict the exact aligned position of the gripper, the force closure tests for a slip push must succeed over all possible aligned positions of the gripper.

2) *Collecting Similar Conservative-Slip Push Configurations*: Before evaluating force closure on the candidate grasps that result in conservative-slip pushes, the conservative-slip push configurations for those candidate grasps are collected into sets of similar configurations. A similar configuration set often contains all the conservative-slip pushes for some edge of the part. Because our estimation of force closure for all positions on an edge can be determined analytically, the estimated closure success of all elements of a similar configuration set can be evaluated simultaneously, as shown in Section IV-A.3.

3) *Conditions for Force Closure*: Force closure on a part is achieved when the line between the contact points on each side lies inside the friction cones of both contact points [27]. If there are multiple contact points on a side, there need be only one successful contact point for force closure.

In our algorithm, force closure is considered to be achieved under three conditions. First, if the second gripper jaw contacts an edge and the contact direction is within the friction cone, the gripper completes force closure. Second, if the second gripper jaw contacts a convex vertex, and this convex vertex is opposite a section of the first gripper jaw that contacts the part, force closure is successful. The third condition involves slip of the second gripper jaw. If the second gripper jaw can slip along the edge it contacts and come into contact with an adjacent edge, and this configuration produces valid force closure, the gripper is considered successful. While this condition is restrictive, it can be determined for ranges of gripper contact points, whereas more general slip conditions require each grasp to be tested individually. This allows our conservative-slip push test, which returns a range of possible aligned positions, to have force closure estimated efficiently for the entire range.

B. Lower Bound on Probability of Achieving Force Closure

Once the candidate grasp conditions have been evaluated for all part perturbations, the lower bound on the probability

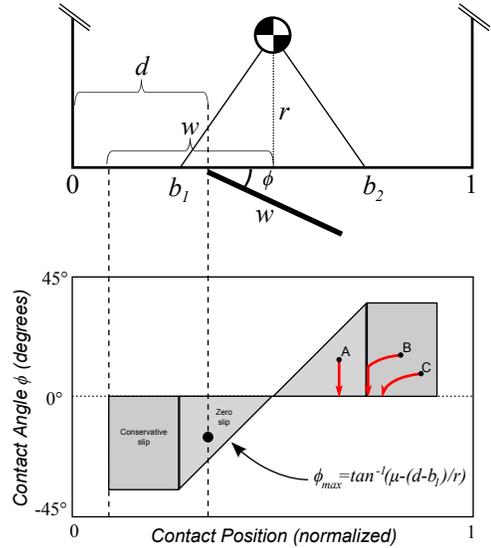


Fig. 3. Configuration space for fast analysis. The upper half of the figure shows a gripper of width w contacting the part at position d with (negative) contact angle $\phi = 30^\circ$, inverse friction cone bounds b_1 and b_2 and perpendicular distance r from the center of mass. Contact with this edge of the part results in the configuration space shown below it; the shaded area is the region where a conservative-slip push occurs. The red lines in the lower region show the configuration-space path for a zero-slip push (A) and possible paths for two conservative-slip pushes (B and C) from initial contact at the points shown. Conservative-slip paths are not predicted specifically, but cannot increase in contact angle or move away from the center of mass. If the path intersects the zero-slip region, it follows a zero-slip path, as displayed by path B.

of achieving force closure for that candidate grasp is estimated using a weighted percentage, where the estimated success or failure on a part perturbation is weighted by the probability of that situation occurring.

C. Adaptive Candidate Grasp Removal

Adaptive grasp candidate removal was added to the algorithm after the observation that the best grasps were already part of the top candidate grasps after only a few part perturbations had been tested, although their final Q -values were not predictable from their Q -values earlier in the analysis. Therefore, the adaptive procedure was developed to remove unpromising grasps, while still testing the promising grasps to refine their Q -values.

After all the part perturbations in a part perturbation set are tested, candidate grasps with low Q -values are removed from further testing. The criterion for removing a grasp is the parameter R . The number of grasps eliminated at step m is $R|G_m|$, giving the total grasp evaluations as $\eta = \sum_{m=1}^M |G_m| \mathcal{M}_m$ where $|G_m| = R|G_{m-1}|$ for $m = 2, \dots, M$. The algorithm checks the minimum Q -value of the top $(1 - R)|G_m|$ candidate grasps, Q_{\min} . If the set of candidate grasps $\{g | Q_g \geq Q_{\min}\}$ is bigger than $(1 - R)|G_m|$, ties between the lowest- Q grasps are broken randomly. The elimination criterion balances maximizing grasp elimination for faster execution with preventing the elimination of grasps that may eventually prove to have high Q -values. Other elimination criteria are possible; ties could be included rather than broken randomly, or all grippers above a certain fraction

of the current best Q -value could be retained. However, these criteria do not guarantee a fixed number of grasp evaluations. We denote the number of grasp evaluations in the adaptive algorithm normalized to the number of evaluations in the non-adaptive algorithm as $\hat{\eta} = \frac{\eta}{N|G_1|}$.

D. Grasp Planning

The grasp planning algorithm shown in Algorithm 2 uses two additional steps to generate candidate grasps and part perturbations, which are then analyzed using our grasp analysis algorithm.

1) *Generating Candidate Grasps*: The grasp planning algorithm generates an initial candidate grasp set $G_1 = \{g_{ij} = (\hat{p}_i, \phi_j) \mid \hat{p}_i \in \hat{P}, \phi_j \in \Phi\}$. While each (\hat{p}, ϕ) pair could be independently generated, we use a fixed set of ϕ values as a parameter, and apply them to a generated set of \hat{p} values, using the method in [21], which takes as parameters a configuration density ρ , a set of approach angles Φ , and a filtering parameter d_C .

2) *Sampling Part Perturbations*: Before testing the candidate grasps, part perturbations are created by sampling from the distributions of each vertex and the center of mass. The number N of part perturbations is determined by a parameter to the algorithm, \mathcal{M} . The part perturbations are collected into part perturbation sets $\mathcal{S}_1, \dots, \mathcal{S}_M$, where $|\mathcal{S}_i| = \mathcal{M}_i$. In previous work [21], we determined that using 100 part perturbations provides reliable results. We explore values of \mathcal{M} in Section V-B.2.

V. RESULTS

To test the algorithm in simulation, a set of images of parts were found on Google Image Search, and manually contoured by tracing a polygon over the image, as used in [21]. We tested the effectiveness of our algorithm for estimating part tolerance bounds by demonstrating a procedure to find these bounds. We also evaluated the new aspects of the algorithm relative to our previous work.

Except for where noted, tests used vertex variance of 20% of the maximum part radius, a center of mass variance of 70% of this radius (measured from the centroid to the vertices), a gripper width 25% of the maximum distance between vertices, and a coefficient of friction of 0.7. The tests were run on a Lenovo T420s laptop with a 2.70 GHz processor and 8 GB of RAM, using MATLAB R2011a.

A. Tolerancing

To test the effectiveness of our algorithm for estimating part tolerance bounds, we developed a procedure to find tolerance limits that allow a grasp to stay above a given Q threshold. Because the variance of different aspects of the part may affect a grasp to a greater or lesser degree, the variances for the parts were split into two groups: the variance of the vertices for the initial contact edge, and the variance of the remaining vertices. The vertices for the initial contact edge along with the center of mass determine the success of the stable push, while other vertices determine the success of closure.

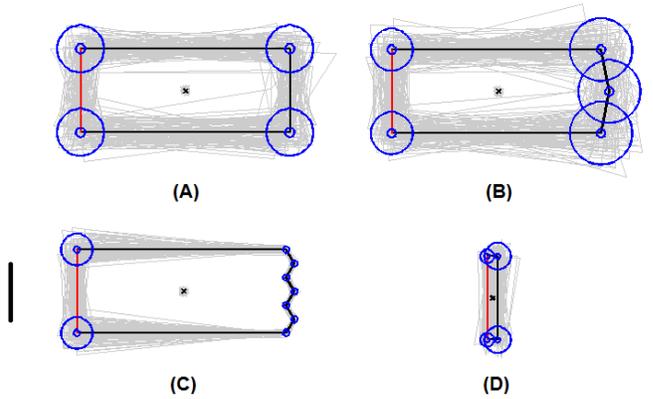


Fig. 4. Tolerancing results for selected parts. The best grasps on the highlighted edge were found with small tolerances shown as the smaller circles around the vertices with radius two standard deviations (95% confidence interval). The gripper width used for all parts is shown next to Part C. Tests were performed as described in Section V-A using $d_C = 0$, $\rho = 4$, and $|\Phi| = 5$, and for the indicated tests from that section, the tolerance for each vertex and center of mass is shown along with 100 perturbations of each part. Parts A and B are shown with tolerances that give comparable \hat{Q}^* (64.5 and 66.9, respectively), and suggest that friction closure is more sensitive to increased tolerances. Part D suggests that, relative to Part A, narrow parts have greater sensitivity to near-edge tolerances.

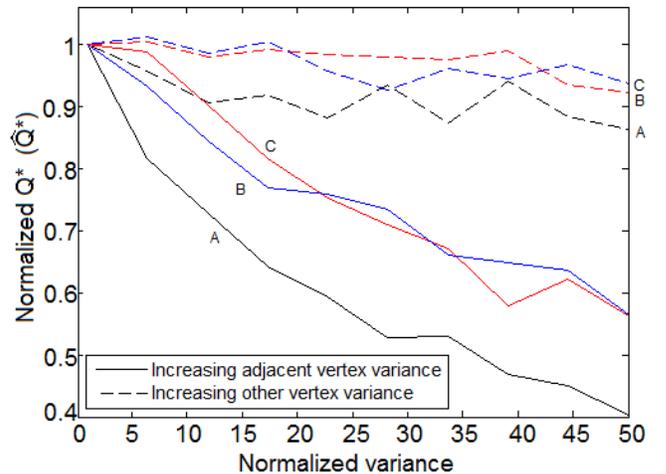


Fig. 5. Effect of increasing tolerances on quality. Tolerance is shown as vertex variance normalized to the initial variance. Each set of three lines show the results for Parts A, B, and C. The solid lines show the average \hat{Q}^* for increasing near-edge vertex variance, keeping other variance constant. The dashed lines show the average \hat{Q}^* for increasing values of the non-near-edge vertex variance, keeping the near-edge variance constant.

To test this tolerance bounding procedure and the effect of variance on closure, three simple parts were created with different features. These parts are shown in Figure 4. Part A, a simple rectangle, tested closure on a flat edge. Part B introduced a single convex vertex instead of a flat edge, to test closure against a vertex. Part C used a set of three vertices to test the effect of complex edges on closure. A fourth part, Part D, was created to test the effect of variance on the initial push. It is a thin rectangle, with the edge to be tested close to the center of mass, creating a smaller valid region more sensitive to higher tolerances. The best grasp on the highlighted edge shown in Figure 4 was found, and this grasp was tested under increasing variance for the near-edge

vertices and the other vertices. The center of mass was fixed to the centroid of each perturbation.

The results for Parts A, B, and C suggest that Q -values are significantly more sensitive to near-edge variance. As shown in Figure 5, as the variance of near-edge vertices increases while the remaining variances are kept constant, the value of Q^* reduces significantly. Keeping the near-edge variance constant while increasing the others had a smaller effect on Q^* , staying within 14% of its initial value.

While the response of these parts was similar when considering the relative change of Q^* , the absolute value showed differences between the parts. The minimum Q^* for Part A was 28.2, for Part B, 46.3, and for Part C, 43.9. Part A had lower Q^* -values because it used only friction closure. Large movements of the vertices can cause the angle between the near edge and the far edge to exceed frictional limits. Closure against a convex vertex is more robust to variance, since such closure does not depend on an angle with the gripper, and if it becomes concave, slip closure may allow force closure.

Part D retained $Q^* = 100$ for tests with high tolerances in the opposite vertices and center of mass, but low tolerances in the adjacent vertices.

We found that the initial contact edge vertices required lower variances, suggesting that success of the stable push was the component of the grasp most sensitive to higher tolerances. In designing a part, tolerance specifications could be defined using the results of this maximum allowable variance test.

B. Comparison with Previous Results

The two main enhancements over our previous work, namely allowing slip and adaptive sampling, contributed to a 16% gain in Q -value and a 71.5% reduction in number of grasp evaluations, respectively.

1) *Comparison With Excluding Slip:* We compared our results to results generated with a previous version of the algorithm that did not permit slip. On average, the value of Q^* was 16% higher when slip was included. Overall, the calculated quality for a given grasp tended to be more intuitive. In Figure 6, the effects of allowing slip are apparent. While the overall best gripper does not change, many of the candidate grasps have significantly higher quality, especially on the most intuitive locations, the long horizontal edges. In particular, the best grasp on the lower horizontal edge without slip has a Q -value of 43.6. With slip allowed, the Q -value of this grasp increases to 91.7.

In earlier work, the algorithm usually did not choose edges close to the center of mass. When limited to zero-slip pushes, edges close to the center of mass are less robust [21]. In our current version of the algorithm, this effect is less pronounced. Although the best grasp on the part shown in Figure 6 was found far away from the center of mass with a Q -value of 92.3, the lower horizontal edge has a grasp with a Q -value of 91.7. Without slip, the best grasp on that edge has a Q -value of 43.6.

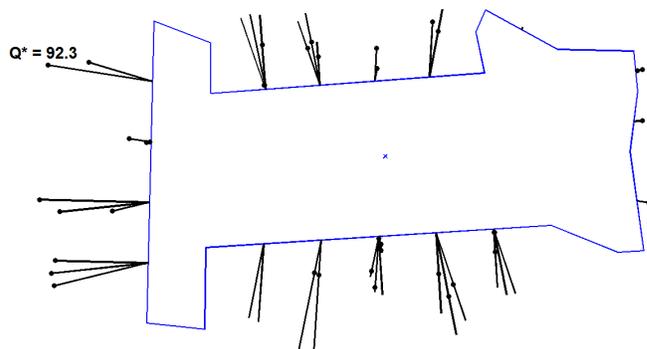


Fig. 6. Improvement of Q -values with slip over excluding slip. The line segments indicate approach lines for the tested nominal contact points, with the length indicating the Q -value relative to other segments. For grasps that have a non-zero Q -value when slip is prohibited, a dot indicates the length of the line segment for that zero-slip-only Q -value. The figure shows the effect of allowing slip was highest on grasps close to the center of mass. The test used $d_C = 0$, $\rho = 1.5$, and $|\Phi| = 5$.

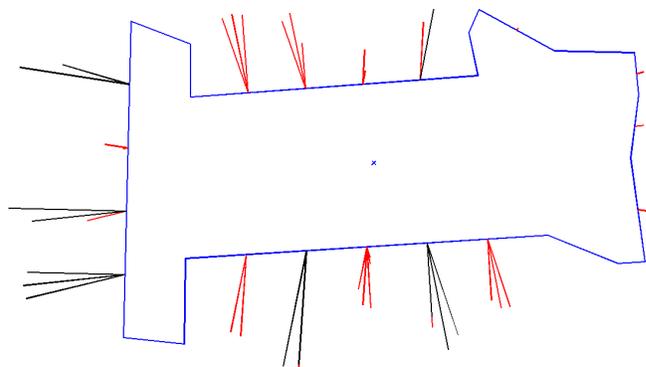


Fig. 7. Candidate grasps eliminated by the adaptive candidate grasp removal. Eliminated grasps are marked in red. The parameters for this test were $d_C = 0$, $\rho = 1.5$, and $|\Phi| = 5$, $R = 0.9$, and $M_1 = 19$.

Including slip results in a higher execution time; the increased complexity of the push and closure tests require a more line intersection checks. The tests we performed took an average of 7% longer when slip was included.

2) *Adaptive Sampling:* The adaptive removal procedure introduced two new parameters, so we tested these parameters to determine their effect on the algorithm's performance.

The adaptive grasp candidate removal step involves a tradeoff between low execution time and high-quality grasps. In particular, if a fixed number of grasp evaluations are allowed, then the larger the initial part perturbation set, the more aggressive the grasp candidate removal step must be. To explore this tradeoff, we tested the adaptive grasp candidate removal step by varying the parameters for both initial part perturbation set size and the grasp elimination criterion. We used a single grasp reduction step (that is, $|\mathcal{M}| = 2$) to do initial testing; our tests using more steps are described at the end of this section.

First, we ran the non-adaptive algorithm (i.e., $|\mathcal{M}| = 1$) on the dataset of parts from [21]; each of the twelve parts was tested using twenty separately generated perturbation sets (given a total of 120 part/perturbation set combinations), using $d_C = 0.06$, $\rho = 6$, and $|\Phi| = 5$, and $N = 70$. The value of Q^* for each test was thus the maximum

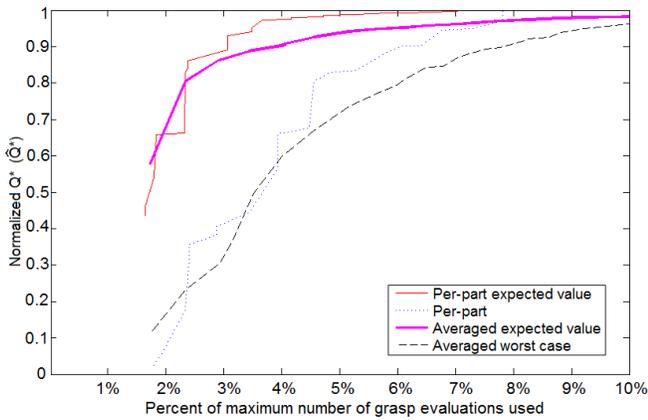


Fig. 8. Tradeoff between execution time and grasp quality, showing \hat{Q}^* vs. percent of grasp candidate evaluations performed ($100 \times \hat{\eta}$) for multiple test parts and adaptive parameters. The graph is truncated at 10% on the x axis because all per-part expected and worst-case values after this have a \hat{Q}^* of 1. A value of 1 on the y axis indicates the overall best gripper was still found by the adaptive algorithm. The Pareto curve of average expected \hat{Q}^* over all parts tested is shown as a solid magenta line, and the Pareto curve of worst case is shown as a dashed black line. The red and blue lines show the lower bound of the Pareto curves for per-part expected and worst-case values, respectively.

Q^* that could be found by the adaptive algorithm (i.e., it was Q_{max}^*). Then, for each initial perturbation set size $M_1 = 1, \dots, 70$ all possible distinct values of the adaptive elimination threshold were found. For each test, the unique Q -values at the M_1 -th iteration were found, and the values of the elimination threshold that would select those Q -values were found. Then, for each initial perturbation set size, all of the distinct values of the adaptive elimination threshold R from all of the tests were combined into a set, and for each threshold value (which was determined from a single part/perturbation set), the outcome of the adaptive algorithm on all of the 120 part/perturbation sets using that threshold value was analyzed.

To analyze the outcome of the adaptive algorithm on a part/perturbation set, we used data from the already-run non-adaptive test. At the given M_1 -th iteration, the grasp reduction step was simulated from the Q -values calculated previously. However, because the grasp elimination criterion randomly breaks ties, it couldn't be used directly. Instead, the worst-case and expected values were found. The worst value was found by retaining the tied candidate grasps with the lowest final Q -values. The expected value was calculated as the sum over all combinations of the maximum final Q -value in that combination weighted by the likelihood of occurrence of the combination.

The result of this analysis is shown in Figure 8. The adaptive sampling was able to aggressively reduce the candidate grasp set without reducing \hat{Q}^* . Considering the best parameter values for each part individually, the results suggest a very low number of perturbations must be tested to find high quality grasps. Above $\hat{\eta} = 0.031$ (that is, 3.1% of the possible grasp evaluations are performed), the expected value of Q^* was within 10% of the maximum, and the worst case values reached the maximum by $\hat{\eta} = 0.08$. Averaging

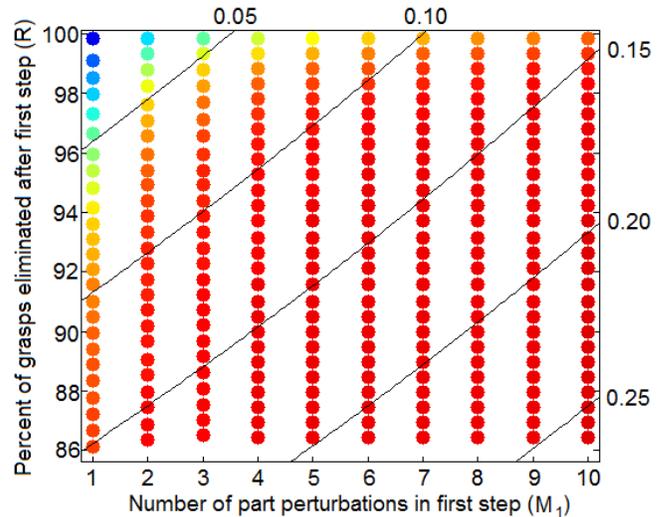


Fig. 9. Tradeoff in worst-case quality (color) and execution time (lines) over parameter combinations. The color of each dot indicates the average worst-case \hat{Q}^* for the parameter values. For example, the point in the upper left represents $M_1 = 1$ and 99.85% of grasps eliminated (i.e., $R = 0.0015$), meaning after one part perturbation is tested, one grasp is selected from the successful grasps on that perturbation, and tested on the remainder of the perturbations. This point has a \hat{Q}^* of 0.577. Contours of $\hat{\eta}$ between 0.05 and 0.25 are shown. The parameter values at any point along a contour require the same number of grasp evaluations.

\hat{Q}^* across all parts for each parameter combination, the performance reduces slightly: the expected value of \hat{Q}^* does not reach the maximum until $\hat{\eta} = 0.277$, and the worst case did not reach maximum until $\hat{\eta} = 0.285$. However, for $\hat{\eta} \geq 0.08$ (when the per-part worst case \hat{Q}^* reaches 1), the best expected value of \hat{Q}^* averaged over all parts was 0.978, and the worst case was 0.926.

This analysis would allow a designer to choose the best adaptive parameters satisfying design constraints, either reducing the number of evaluations given a minimum worst case or expected value, or maximizing worst case or expected value given a maximum number of evaluations.

Figure 8 does not indicate what parameter values produce the displayed Pareto curves. Figure 9 shows the average worst-case value of \hat{Q}^* over all tests for parameter ranges $M_1 \in [1, 10]$ and $R \in [0.85, 1]$. The contours of $\hat{\eta}$ are shown for several values between 0.05 and 0.25. Given a low limit on grasp evaluations, this analysis allows the best parameter combination satisfying the constraint to be found.

Good grasps are identified after testing a small number of part perturbations, as shown in our previous work. This allows the adaptive grasp elimination step to cull unpromising grasp candidates, and use the remaining part perturbations to refine the Q -value of the good grasp candidates. We experimented with using more than one iteration of grasp candidate removal, but the extra reduction in number of grasp candidates was of minimal benefit.

VI. DISCUSSION AND FUTURE WORK

We have presented an approach for quickly analyzing conservative-slip push grasps on planar parts by finding the value of a quality metric that estimates a lower bound on

the probability of force closure. We improved our previous algorithm by allowing slip and using an adaptive candidate grasp elimination step. Allowing slip improves Q^* by an average of 16%. The adaptive elimination step reduces grasp evaluations by 91.5% while maintaining 92.6% of grasp quality. In future work, we will expand the adaptive sampling to eliminate bad grasps faster, resample near good grasps to allow coarser initial grasp candidate sets, and separate force closure tests into multiple subtests to enable more flexible evaluation of grasps in a search-like manner.

In addition to the algorithmic improvements listed above, we developed a procedure to find part tolerance bounds. In future work, we will use the analysis of the effect of tolerances on grasp quality to iterate on the design of features to improve grasp quality under the given tolerances.

VII. ACKNOWLEDGMENTS

We thank James Kuffner and Frank van der Stappen for valuable discussions. This work was supported in part by NSF Award 0905344.

REFERENCES

- [1] IEEE Society of Robotics and Automation Technical Committee on Networked Robotics. <http://tab.ieee-ras.org/committeeinfo.php?tcid=15>.
- [2] S. Akella and M.T. Mason. Posing polygonal objects in the plane by pushing. In *IEEE International Conference on Robotics and Automation*, pages 2255–2262. IEEE Comput. Soc. Press, 1992.
- [3] Rajesh Arumugam, V.R. Enti, Liu Bingbing, Wu Xiaojun, Krishnamoorthy Baskaran, F.F. Kong, A.S. Kumar, K.D. Meng, and G.W. Kit. In *IEEE International Conference on Robotics and Automation*, pages 3084–3089.
- [4] Dmitry Berenson, Siddhartha S. Srinivasa, and James J. Kuffner. Addressing pose uncertainty in manipulation planning using Task Space Regions. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1419–1425, October 2009.
- [5] R. C. Brost. Automatic Grasp Planning in the Presence of Uncertainty. *The International Journal of Robotics Research*, 7(1):3–17, February 1988.
- [6] Jingliang Chen, Ken Goldberg, Mark H. Overmars, Dan Halperin, Karl F. Böhringer, and Yan Zhuang. Computing tolerance parameters for fixturing and feeding. *Assembly Automation*, 22(2):163–172, 2002.
- [7] Jae-Sook Cheong, Herman J. Haverkort, and A Frank van der Stappen. Computing All Immobilizing Grasps of a Simple Polygon with Few Contacts. *Algorithmica*, 44(2):117–136, December 2005.
- [8] Jae-Sook Cheong, Heinrich Kruger, and A Frank van der Stappen. Output-Sensitive Computation of Force-Closure Grasps of a Semi-Algebraic Object. *IEEE Transactions on Automation Science and Engineering*, 8(3):495–505, July 2011.
- [9] Vassilios N Christopoulos and Paul R Schrater. Handling Shape and Contact Location Uncertainty in Grasping Two-Dimensional Planar Objects. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1557–1563. IEEE, 2007.
- [10] Matei Ciocarlie, Kaijen Hsiao, E.G. Jones, Sachin Chitta, R.B. Rusu, and I.A. Sucas. Towards reliable grasping and manipulation in household environments. In *Intl. Symposium on Experimental Robotics*, pages 1–12, 2010.
- [11] Matei Ciocarlie, Caroline Pantofaru, Kaijen Hsiao, Gary Bradski, Peter Brook, and Ethan Dreyfuss. A Side of Data With My Robot. *IEEE Robotics & Automation Magazine*, 18(2):44–57, June 2011.
- [12] J. Cornelia and R. Suarez. Efficient Determination of Four-Point Form-Closure Optimal Constraints of Polygonal Objects. *IEEE Transactions on Automation Science and Engineering*, 6(1):121–130, January 2009.
- [13] Mehmet R Dogar and Siddhartha S Srinivasa. Push-grasping with dexterous hands: Mechanics and a method. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2123–2130. IEEE, October 2010.
- [14] Javier Felip and Antonio Morales. Robust sensor-based grasp primitive for a three-finger robot hand. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1811–1816, October 2009.
- [15] Jared Glover, Daniela Rus, and Nicholas Roy. Probabilistic Models of Object Geometry for Grasp Planning. In *Robotics: Science and Systems*, 2008.
- [16] Ken Goldberg. Algorithmic Automation. <http://goldberg.berkeley.edu/algorithmic-automation/>.
- [17] Ken Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10(2-4):201–225, October 1993.
- [18] Corey Goldfeder and Peter K. Allen. Data-Driven Grasping. *Autonomous Robots*, 31(1):1–20, April 2011.
- [19] Kaijen Hsiao, Leslie Pack Kaelbling, and Tomas Lozano-Perez. Grasping POMDPs. In *IEEE International Conference on Robotics and Automation*, pages 4685–4692. IEEE, April 2007.
- [20] Leo Joskowicz, Yaron Ostrovsky-Berman, and Yonatan Myers. Efficient representation and computation of geometric uncertainty: The linear parametric model. *Precision Engineering*, 34(1):2–6, January 2010.
- [21] B. Kehoe, D. Berenson, and K. Goldberg. Toward Cloud-Based Grasping with Uncertainty in Shape: Estimating Lower Bounds on Achieving Force Closure with Zero-Slip Push Grasps. In *IEEE International Conference on Robotics and Automation*. IEEE, 2012. To appear, available at <http://goldberg.berkeley.edu/pubs/kehoe-cloud-icra-2012-final.pdf>.
- [22] Ellen Klingbeil, Deepak Rao, Blake Carpenter, Varun Ganapathi, A.Y. Ng, and Oussama Khatib. Grasping with Application to an Autonomous Checkout Robot. In *IEEE International Conference on Robotics and Automation*, 2011.
- [23] James J. Kuffner. Cloud-Enabled Robots. In *IEEE-RAS International Conference on Humanoid Robotics*, Nashville, TN, 2010.
- [24] K.M. Lynch. The mechanics of fine manipulation by pushing. In *IEEE International Conference on Robotics and Automation*, pages 2269–2276. IEEE Comput. Soc. Press, 1992.
- [25] M.T. Mason. Manipulator grasping and pushing operations. Technical report, Massachusetts Inst. of Tech., Cambridge (USA). Artificial Intelligence Lab., 1982.
- [26] G. McKee. What is Networked Robotics? *Informatics in Control Automation and Robotics*, 15:35–45, 2008.
- [27] Van-Duc Nguyen. Constructing stable force-closure grasps. In *ACM Fall Joint Computer Conference*, pages 129–137. IEEE Computer Society Press, 1986.
- [28] M. Peshkin and A. Sanderson. Planning robotic manipulation strategies for sliding objects. In *IEEE International Conference on Robotics and Automation*, volume 4, pages 696–701. Institute of Electrical and Electronics Engineers, 1987.
- [29] Robert Platt, Leslie Kaelbling, Tomas Lozano-Perez, and Russ Tedrake. Simultaneous Localization and Grasping as a Belief Space Control Problem. In *International Symposium on Robotics Research*, pages 1–16, 2011.
- [30] A A G Requicha. Toward a Theory of Geometric Tolerancing. *The International Journal of Robotics Research*, 2(4):45–60, December 1983.
- [31] Alberto Rodriguez, M.T. Mason, and Steve Ferry. From Caging to Grasping. In *Robotics: Science and Systems*, Los Angeles, CA, USA, 2011.
- [32] C. Rosales, L. Ros, J. M. Porta, and R. Suarez. Synthesizing Grasp Configurations with Specified Contact Regions. *The International Journal of Robotics Research*, July 2010.
- [33] John D Schulman, Ken Goldberg, and Pieter Abbeel. Grasping and Fixturing as Submodular Coverage Problems. In *International Symposium on Robotics Research*, pages 1–12, 2011.
- [34] G. Smith, E. Lee, K. Goldberg, K. Bohringer, and J. Craig. Computing parallel-jaw grips. *IEEE International Conference on Robotics and Automation*, 3:1897–1903, 1999.
- [35] Chao-Ping Tung and A.C. Kak. Fast construction of force-closure grasps. *IEEE Transactions on Robotics and Automation*, 12(4):615–626, June 1996.
- [36] Markus Waibel. RoboEarth: A World Wide Web for Robots. Automaton Blog, IEEE Spectrum. <http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/roboearth-a-world-wide-web-for-robots>, Feb. 5, 2011.
- [37] T. Zhang, L. Cheung, and K. Goldberg. Shape Tolerance for Robot Gripper Jaws. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 1782–1787. IEEE, 2001.