

Grasping and Fixturing as Submodular Coverage Problems

John D. Schulman and Ken Goldberg and Pieter Abbeel

Abstract Grasping and fixturing are concerned with immobilizing objects. Most prior work in this area uses a small number of contacts that is sufficient for force closure. However, for delicate objects or surfaces such as glass or bone (in medical applications), a larger number of contacts can be used to reduce the forces needed at each contact to resist applied wrenches. We focus on the problem of choosing a set of contact points out of a larger set of candidates to optimize *grasp quality*, which is defined as the size of the ball in wrench space that can be resisted given a constraint on each contact force or the sum of the contact forces. In purely geometric terms, our algorithms select a set of vectors out of a larger set of candidates to maximize the residual radius of the Minkowski sum or convex hull. We provide a method that is guaranteed to find near-optimal solutions in linear time. At the core of our approach are (i) an novel formula for the quality functions, and a discretization technique that evaluates them with bounded error, (ii) the insight that the resulting problem is a submodular coverage problem. This allows us to exploit the submodular saturation algorithm, which has recently been derived for applications in sensor placement. Our alternative formulas also makes it possible relax the optimization problem into a linear program, and we give a branch-and-bound procedure for exactly optimizing the objective. Our approach is applicable in situations with or without friction, for a large class of friction models. We also show that grasp quality metrics are relevant to towing (carrying a heavy object with aerial vehicles), and describe how to use the same methods to find optimal towing configurations.

1 Introduction

The problem of choosing contact points on an object to securely hold it is relevant to robotics and manufacturing. Robotics is concerned with *grasping*, where a sin-

John D. Schulman, e-mail: joschu@berkeley.edu · Ken Goldberg, e-mail: goldberg@berkeley.edu · Pieter Abbeel, e-mail: pabbeel@cs.berkeley.edu
Department of Electrical Engineering and Computer Science, University of California, Berkeley

gle robot uses a multi-fingered hand to hold and manipulate an object, or multiple robots cooperate to pick up a large object. Manufacturing is concerned with *fixturing*, where a machinist uses locators and clamps to immobilize a part for operations such as inspection or machining. In both grasping and fixturing, there are often constraints on the forces that can be applied at the contact points.

Most prior work in this area strives to minimize the number of contacts needed for force closure, known to be four in the plane and seven in three dimensions. However, for delicate objects or surfaces such as glass or bone, extra contacts can be used to reduce the forces needed at each contact to resist applied wrenches.

Researchers have proposed a number of approaches to extend the notions of form and force closure with scalar “quality” measures for grasping or fixturing configurations. We use the wrench-space quality measures proposed by [7] and [5]. These elegant measures are based on convex geometry and maximize the disturbance that can be resisted given bounds on the contact forces. See [9] for a review on quality metrics. The metrics we consider are as follows:

- Q_1 : The norm of the smallest wrench that can’t be resisted, given a constraint on the sum of the normal forces.
- Q_∞ : The norm of the smallest wrench that can’t be resisted, given a constraint on the maximum normal force.

The notion of norm in wrench space is not well-defined, since the wrench vector has force and torque components, which have different units. Ferrari and Canny [5] address this problem by using the length scale of the object as a dimensional constant that relates forces to torques; we describe a different solution in Section 3 based on the minimum-volume ellipsoid containing a set of realistic wrenches.

Our main contributions are as follows:

- Novel formulas for Q_1 and Q_∞ and their generalizations that incorporate friction. Fast numerical evaluation of these formulas requires a discretization approximation for which we provide error bounds.
- A fast algorithm, based on the submodular saturation algorithm, for selection of the contact points that maximize Q_1 and Q_∞ , which is guaranteed to give a near-optimal solution. The run-time is roughly linear in the number of candidates and the number of contacts. Our implementation selects tens of contacts from one hundred candidates in seconds.
- A branch-and-bound algorithm, which provides the exact solution to the optimization problems. Our implementation solves large problems (choosing tens of contacts out of one hundred candidates) in minutes.
- A discussion of the problem of placing aerial vehicles for towing and its connection to grasping with friction.

2 Related Work

The Q_1 metric for grasping and its geometric interpretation was originally proposed by Kirkpatrick et al. [7]. They provided a fast algorithm to find the subset of a set

of two-dimensional vectors with maximal *residual radius* (defined in Section 3), but suggested that this problem is hard in higher-dimensional spaces. A later paper [12] provides a probabilistic algorithm to optimize the Q_1 metric in the frictionless case. Another article investigates fast methods for calculating the Q_∞ metric but not optimizing it [1]. Pollard [10] provides a method to find multi-contact grasps near a given grasp, with no guarantee on optimality.

The problem of contact point selection has also been studied from the perspective of fixture design. Brost and Goldberg [4] provide an algorithm to find viable peg and clamp placements in a planar modular fixturing apparatus, and Brost and Peters [3] extend these results to three dimensions and incorporate quality metrics. These methods require searching through a large number of configurations—exponential in the number of contacts. Wang [13] optimizes a different quality metric, which does not inherently guarantee force closure or force limits, by using the greedy algorithm on a max-det problem.

3 Background: Geometry of Quality Metrics

In this section, we review the geometric interpretation of Ferrari and Canny’s quality metrics [5] and introduce some notation. Given k frictionless contact points, suppose that applying unit normal forces at these contacts generates wrenches $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k$. Then if we apply normal forces f_1, f_2, \dots, f_k , the total wrench is

$$\mathbf{w} = \sum_i f_i \mathbf{w}_i. \quad (1)$$

Under the L_1 constraint $\sum_i f_i \leq 1$, the set of attainable wrenches \mathbf{w} is $\text{ConvexHull}(\mathbf{0}, \mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)$. Under the L_∞ constraint $\max_i f_i \leq 1$, the set of attainable wrenches is $\text{MinkowskiSum}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)$. In both cases, the constraint $f_i \geq 0$ is implied.

The *residual radius* of a compact set C is defined as the distance from the origin to the boundary of C , and 0 if C does not contain the origin:

$$r_{\text{res}}(C) = \begin{cases} \text{dist}(\mathbf{0}, \mathbb{R}^n \setminus C) & \text{if } \mathbf{0} \in C \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

The quality metrics, informally defined in the introduction, are formally defined as follows:

$$Q_1 = r_{\text{res}}(\text{ConvexHull}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)), \quad (3)$$

$$Q_\infty = r_{\text{res}}(\text{MinkowskiSum}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)). \quad (4)$$

These quality metrics can be generalized to the case where there is friction. Define W_i as the tolerable set of wrenches that can be produced at the i th contact. For example, we might specify that the magnitude of the force is no greater than 1. Then, in analogy with the frictionless case, we define the quality functions as

$$Q_1 = r_{\text{res}}(\text{ConvexHull}(W_1, W_2, \dots, W_k)), \quad (5)$$

$$Q_\infty = r_{\text{res}}(\text{MinkowskiSum}(W_1, W_2, \dots, W_k)). \quad (6)$$

Informally, the convex hull is the set where the total effort, summed over contacts, is constrained; and the Minkowski sum is the set where the effort at each individual contact is constrained.

Natural norms in wrench space

The quality metrics Q_1 and Q_∞ depend on a norm in wrench space. Since force and torque have different units, it is not obvious how to define the norm of vectors in wrench space, so that we can talk about the size of the ball of wrenches that can be resisted. In the following paragraphs we describe how to define the norm of wrench space in a natural way that depends on the object being immobilized and the disturbances applied to it.

Suppose that we know the set of “disturbances”—wrenches that will be applied to the object during a task. This set will typically include the wrench due to gravity, and it might contain wrenches arising from forces applied to the surface. If we don’t know the set of disturbances, we can simply use the contact wrenches. Given this set of task wrenches, calculate the minimum-volume enclosing ellipsoid for these points (see [2], 8.4 for a discussion of this problem.) We will change coordinates so that this ellipsoid becomes a unit ball, where the most extreme points in our set of expected wrenches will have norm 1.

Specifically, if the ellipsoid is parameterized as

$$\{\mathbf{w} \mid \|\mathbf{A}\mathbf{w} - \mathbf{x}_c\| \leq 1\}. \quad (7)$$

then we transform into normalized coordinates $\mathbf{x} = \mathbf{A}\mathbf{w}$, so in our new coordinates, the ellipsoid is

$$\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}_c\| \leq 1\}. \quad (8)$$

For the remainder of this paper, we will be working in the space of normalized wrenches, but for clarity, we will refer to these normalized wrenches with the letter \mathbf{w} . To transform the problem into normalized wrench space, multiply all of the disturbances and contact wrenches by the matrix \mathbf{A} .

We can also interpret the residual radius in normalized wrench space as a safety margin for the given set of disturbances. By definition, r_{res} is the scale factor so that if we blow up the unit sphere by a factor of r_{res} , it lies fully inside the set of tolerable wrenches that our contacts can resist. In normalized wrench space, the unit sphere encloses the set of disturbances. Thus if we scale all the disturbances by a factor of r_{res} , they lie inside the set of tolerable wrenches.

4 Alternate Formulation of Quality Metrics

In this section, we present an alternate expression for the residual radius of a convex set. Besides being efficient to evaluate, this alternate form reveals that the radius maximization problem can be solved by methods that exploit submodularity, and the problem can also be relaxed to a linear program.

Our results are as follows. For frictionless contact points,

$$Q_1 = r_{\text{res}}(\text{ConvexHull}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)) = \min_{\|\mathbf{y}\|=1} \max_i (\mathbf{y}^T \mathbf{w}_i)^+ \quad (9)$$

$$Q_\infty = r_{\text{res}}(\text{MinkowskiSum}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)) = \min_{\|\mathbf{y}\|=1} \sum_i (\mathbf{y}^T \mathbf{w}_i)^+ \quad (10)$$

where $(\cdot)^+$ is the “positive part”, i.e.,

$$(x)^+ = \begin{cases} x & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

For contact points with friction, where the wrenches that can be produced by the i th contact lie in set W_i , we obtain

$$Q_1 = r_{\text{res}}(\text{ConvexHull}(W_1, W_2, \dots, W_k)) = \min_{\|\mathbf{y}\|=1} \max_i h_{W_i}(\mathbf{y}) \quad (12)$$

$$Q_\infty = r_{\text{res}}(\text{MinkowskiSum}(W_1, W_2, \dots, W_k)) = \min_{\|\mathbf{y}\|=1} \sum_i h_{W_i}(\mathbf{y}), \quad (13)$$

where $h_{W_i}(\mathbf{y}) = \max_{\mathbf{x} \in W_i} \mathbf{y}^T \mathbf{x}$.

The key concept for deriving these formulas is the *support function*, illustrated in Figure 1. Given a compact, convex set C and a point \mathbf{y} , the support function $h_C(\mathbf{y})$ is defined as [6]

$$h_C(\mathbf{y}) = \max_{\mathbf{x} \in C} \mathbf{y}^T \mathbf{x}. \quad (14)$$

For $\|\mathbf{y}\| = 1$, the support function tells us the height of C in the direction \mathbf{y} . The support function has the following properties:

$$h_{\text{ConvexHull}(C_1, C_2, \dots, C_n)}(\mathbf{y}) = \max_i h_{C_i}(\mathbf{y}) \quad (15)$$

$$h_{\text{MinkowskiSum}(C_1, C_2, \dots, C_n)}(\mathbf{y}) = \sum_i h_{C_i}(\mathbf{y}) \quad (16)$$

Proposition 1. *For compact, convex set C containing the origin,*

$$r_{\text{res}}(C) = \min_{\|\mathbf{y}\|=1} h_C(\mathbf{y}) \quad (17)$$

$$= \min_{\|\mathbf{y}\|=1} \max_{\mathbf{x} \in C} \mathbf{y}^T \mathbf{x} \quad (18)$$

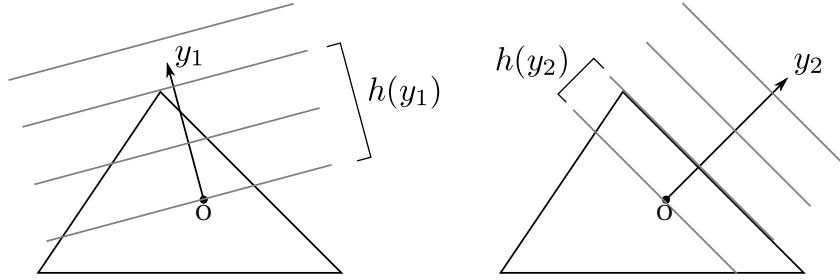


Fig. 1: An illustration of the support function h for a triangular set, evaluated at two unit vectors \mathbf{y}_1 and \mathbf{y}_2 . The proposition states that the residual radius of the triangle around O is the smallest value of the support function, which is $h(\mathbf{y}_2)$.

Proof. See appendix.

Combining this proposition with Equations (15) and (16), we obtain the formulas in Equations (12) and (13), which specialize to Equations (9) and (10) in the frictionless case.

We can evaluate these formulas by discretizing the sphere (or hypersphere) $\{\mathbf{y} \mid \|\mathbf{y}\| = 1\}$. The following proposition shows that the error in calculating the residual radius is bounded by about $1/n$ if we choose $6n^2$ samples on the 2-sphere or $12n^5$ samples on the 5-sphere, corresponding to the wrench space of 2D or 3D objects, respectively.

Proposition 2. *We can construct a set S_n of $(2p)n^{p-1}$ samples of the $(p-1)$ -sphere S (a subset of \mathbb{R}^p), such that the following bound holds:*

$$0 \leq \min_{\mathbf{y} \in S_n} h_C(\mathbf{y}) - \min_{\mathbf{y} \in S} h_C(\mathbf{y}) \leq \frac{\sqrt{p}}{2n} \|\mathbf{x}_n\| \quad (19)$$

where $\mathbf{x}_n = \arg \max_{\mathbf{x} \in C} \mathbf{y}_n^T \mathbf{x}$.

In the frictionless case, the computational cost of evaluating the quality function is just due to calculating the matrix of inner products $\mathbf{y}_i \cdot \mathbf{w}_j$. In the frictional case, we must calculate $h_{W_i}(\mathbf{y}_j)$ for every contact i and disturbance j . In general, this is a convex optimization problem, and under the Coulomb friction model it has a closed form solution. If contact force f has a normal component f^\perp and tangential components \mathbf{f}^\parallel , and the set of contact wrenches is given by

$$W_i = \{\mathbf{G}_i \mathbf{f} \mid \|\mathbf{f}^\parallel\| \leq \mu f^\perp, f^\perp \leq 1\}, \quad (20)$$

a straightforward calculation shows that

$$h_{W_i}(\mathbf{y}) = \begin{cases} \mathbf{y}^T \mathbf{G}_i^\perp + \mu \|\mathbf{y}^T \mathbf{G}_i^\parallel\| & \text{if } \mathbf{y}^T \mathbf{G}_i^\perp > 0 \\ 0 & \text{otherwise} \end{cases} \quad (21)$$

where we have written $\mathbf{G}_i = [\mathbf{G}_i^\perp \mid \mathbf{G}_i^\parallel]$ corresponding to the normal and frictional components.

5 Quality Function Optimization as a Submodular Coverage Problem

We are given a set S of candidate contact points, where the i th contact generates a set of object wrenches W_i when the maximum force is constrained. We would like to find a subset $S' \subseteq S$ to optimize the quality function 5 or 6. In other words, we would like to choose the subset of our contact wrench sets W_i whose convex hull or Minkowski sum has maximal residual radius. So we are solving the following optimization problems:

$$\max_{S' \subseteq S} Q_1 = \max_{S' \subseteq S} \min_{\|\mathbf{y}\|=1} \max_{i \in S'} h_{W_i}(\mathbf{y}), \quad (22)$$

$$\max_{S' \subseteq S} Q_\infty = \max_{S' \subseteq S} \min_{\|\mathbf{y}\|=1} \sum_{i \in S'} h_{W_i}(\mathbf{y}), \quad (23)$$

subject to the cardinality constraint $|S'| \leq k$.

Krause et al. [8] introduced the *submodular saturation* algorithm (henceforth called SATURATE), which solves problems where we are trying to optimize the minimum of a collection of objectives:

$$\max_{S' \subseteq S} \min_i F_i(S'), \quad \text{subject to } |S'| \leq k. \quad (24)$$

When the functions F_i are submodular, there are theoretical performance guarantees on this algorithm. Namely, if we relax the cardinality constraint to $|S'| \leq \alpha k$ (where α is a parameter depending on the problem) and run the algorithm, then the αk -element solution found by SATURATE is guaranteed to be better than the optimal k -element solution. SATURATE is very fast, and it involves performing the greedy algorithm several times with a transformed objective function. See the appendix for a review of SATURATE. In this paper, we use SATURATE to find suboptimal solutions to the k -element selection problem, so we do not use α .

The expressions $\max_{i \in S'} h_{W_i}(\mathbf{y})$ and $\sum_{i \in S'} h_{W_i}(\mathbf{y})$ are both submodular functions of the set S' , since max and sum are submodular. Each direction in wrench space \mathbf{y} indexes a different objective function that we are trying to optimize. Therefore, optimization of the quality functions over S' has exactly the form given in Equation (24), so SATURATE can be applied to this problem. To apply the algorithm, we need to calculate $h_{W_i}(\mathbf{y}_j)$ for every pair (i, j) consisting of a contact point i and a direction in

wrench space j . To evaluate the evaluate the collection of objective functions (one for each disturbance) when we perform SATURATE, we merely need to take the max and sum over the columns of a table—a computationally cheap operation. The computation time of the algorithm is thus $O(NKD)$, where N, K, D are the number of candidates, the number of contacts, and the number of samples of the sphere in our discretization.

We describe the process for selecting contact points in the frictionless situation with SATURATE in Algorithms 1 and 2 below. There are only a couple differences in the frictional case: (1) \mathbf{W} describes the wrenches from frictional as well as normal forces (which are also converted to normalized coordinates by matrix \mathbf{A}), and (2) the entries \mathbf{H} are not given by an inner product, but a support function, e.g. see Equation (21).

Algorithm 1: SelectContactPointsFrictionless

```

input : contact wrenches  $\mathbf{W}[6, N]$ ,
         disturbance wrenches  $\mathbf{Z}[6, D]$ 
         desired number of contacts  $K$ ,
         quality metric  $\in \{Q_1, Q_\infty\}$ 
         discretization resolution  $p$ 
output: selected indices  $I$ , quality  $Q$ 

 $\mathbf{A}[6, 6], \mathbf{x}_c[6] \leftarrow \text{MinVolEllipse}(\mathbf{Z})$ ;           // See Equation 7
 $\mathbf{X}[6, N] \leftarrow \mathbf{A}\mathbf{W}$ ;
 $\mathbf{Y}[6, M] \leftarrow \text{SphereSamples}(p)$ ;
 $\mathbf{H}[M, N] \leftarrow (\mathbf{Y}^T \mathbf{X})^+$ ;                       // Support function for each
                                                         // (disturbance, contact) pair
 $\mathbf{h}_c[M] \leftarrow \mathbf{Y}^T \mathbf{x}_c$ ;                          // Support function of center point
switch quality metric do
  case  $Q_1$  :  $\mathbf{f}(S') := (\max_{i \in S'} H_{i\bullet} - \mathbf{h}_c)^+$ ;      //  $\mathbf{H}_{i\bullet}$  is the  $i$ th row of  $\mathbf{H}$ 
  case  $Q_\infty$  :  $\mathbf{f}(S') := (\sum_{i \in S'} H_{i\bullet} - \mathbf{h}_c)^+$ ;
 $I[K], Q \leftarrow \text{Saturate}(\mathbf{f}, K, \{1, 2, \dots, N\})$ ;

```

6 A Branch-and-Bound Algorithm for Exact Optimization

In this section, we describe a branch-and-bound algorithm for finding the optimal solution to the quality function optimization (residual radius maximization) problem. Branch-and-bound algorithms globally maximize a function over a space of solutions, and they have the following features:

- A way of splitting the solution space into two or more disjoint subsets. We will call each subset of the solution space a *subproblem*.
- A heuristic or suboptimal method for generating a lower bound on the optimal value.
- A relaxation of the problem, which provides an upper bound on the optimal value.

Algorithm 2: Saturate

```

input : function  $\mathbf{f}: S' \rightarrow \mathbb{R}^M$ 
         number of elements to select  $K$ 
         set of candidates  $S$ 
output: selected subset  $S'$ , function value  $Q$ 

 $lb \leftarrow \min \mathbf{f}(\{s_1, s_2, \dots, s_K\});$  // choose  $K$  arbitrary elements from  $S$ 
                                                // to get a lower bound
 $ub \leftarrow \min \mathbf{f}(S);$  // choose all  $N$  elements to get an upper bound
                                                // (assuming  $\mathbf{f}$  is increasing)

 $n_{\text{iter}} \leftarrow \log(\text{desired precision});$  // we used  $n_{\text{iter}} = 10$ 
for  $iteration \leftarrow 1, 2, \dots, n_{\text{iter}}$  do
     $c \leftarrow (lb + ub)/2;$  // saturation level we are testing
     $\bar{\mathbf{f}}_c(S') := \min(\mathbf{f}(S'), c);$  // truncated objectives
     $S', Q \leftarrow \text{Greedy}(\bar{\mathbf{f}}_c, K, S);$  // using the greedy algorithm, select a
                                                //  $K$ -element subset of  $S$  to maximize  $\bar{\mathbf{f}}_c$ 
    if  $\mathbf{f}(S') \succeq c$  then // successfully saturated all objectives
         $lb \leftarrow c$ 
    else
         $ub \leftarrow c$ 

```

The algorithm constructs a tree with the original problem at the root. The children of a given subproblem are smaller subproblems, which partition its solution space. The algorithm searches through this tree, skipping nodes whose upper bound is less than the greatest lower bound found so far.

In our scheme, we are selecting a k -element subset $S' \subset S = \{1, 2, \dots, N\}$, and the branching procedure consists of selecting the next element. It will be convenient to represent a subset S' by a length- N vector \mathbf{z} , where $z_i = 1$ if $i \in S'$, and $z_i = 0$ if $i \notin S'$. A subproblem is defined by the pair (GOOD, BAD) where GOOD is the list of components of \mathbf{z} that are required to be 1, and BAD is the list of components that are required to be 0. For each subproblem, we calculate the lower and upper bounds, and, if its upper bound is not dominated by some other subproblem's lower bound, we split it into smaller subproblems.

We relax the optimization problem to a linear program by changing the constraint $z_i \in \{0, 1\}$ to $0 \leq z_i \leq 1$. For example, for the optimization of Q_∞ , the non-relaxed optimization problem is

$$\max_{S' \subset S} \min_j \sum_{i \in S'} h_{W_i}(\mathbf{y}_j) \quad \text{subject to } |S'| = k, \quad (25)$$

which is equivalent to

$$\max_{\mathbf{z}} \min_j \sum_{i \in S} h_{W_i}(\mathbf{y}_j) z_i, \quad \text{subject to } \sum_i z_i = k, \quad z_i \in \{0, 1\}. \quad (26)$$

The relaxed problem is

$$\max_{\mathbf{z}} \min_j \sum_{i \in S} h_{W_i}(\mathbf{y}_j) z_i, \quad \text{subject to } \sum_i z_i = k, \quad 0 \leq \mathbf{z} \leq 1. \quad (27)$$

If the optimal solution to the relaxed problem has $z_i = \{0, 1\}$ for some i , then so will the optimal solution to the non-relaxed problem. So in the algorithm below, we keep track of the list of components of \mathbf{z} that are required to be 0 or 1.

Algorithm 3: Branch-and-bound for subset selection

```

input : Number of candidates  $N$ 
         Number of selections  $K$ 
         functions calcLowerBound, calcUpperBound
output: function value  $Q^{\text{opt}}$ 
         indicator vector  $\mathbf{z}^{\text{opt}}$ 

 $Q^{\text{glb}} \leftarrow -\infty$ ; // Greatest lower bound so far
SubproblemList  $\leftarrow [(\emptyset, \emptyset)]$ ; // root problem: GOOD and BAD are empty
while notEmpty(SubproblemList) do
  (GOOD, BAD)  $\leftarrow$  pop first element of SubproblemList;
   $Q^{\text{upper}}, \mathbf{z}^{\text{upper}} \leftarrow$  calcUpperBound(GOOD, BAD,  $K$ );
  if  $Q^{\text{upper}} > Q^{\text{glb}}$  then
    GOOD'  $\leftarrow \{i \mid z_i^{\text{upper}} = 1\}$ ; // update GOOD and BAD
    BAD'  $\leftarrow \{i \mid z_i^{\text{upper}} = 0\}$ ; // based on integer solutions
     $Q^{\text{lower}}, \mathbf{z}^{\text{lower}} \leftarrow$  calcLowerBound(GOOD', BAD',  $K$ );
    if  $Q^{\text{lower}} < Q^{\text{glb}}$  then
       $Q^{\text{glb}} \leftarrow Q^{\text{lower}}$ ;
       $\mathbf{z}^{\text{glb}} \leftarrow \mathbf{z}^{\text{lower}}$ 
    MAYBE  $\leftarrow \{1, 2, \dots, N\} \setminus \{\text{GOOD} \cup \text{BAD}\}$ ; // List of candidate to add
    for  $i \leftarrow 1, 2, \dots, \text{length}(\text{MAYBE})$  do
      BAD''  $\leftarrow \text{BAD} \cup \{\text{MAYBE}_1, \dots, \text{MAYBE}_{i-1}\}$ ;
      GOOD''  $\leftarrow \text{GOOD} \cup \{\text{MAYBE}_i\}$ ;
      append (GOOD'', BAD'') to SubproblemList; // add new subproblem

 $Q^{\text{opt}} \leftarrow Q^{\text{glb}}$ 
 $\mathbf{z}^{\text{opt}} \leftarrow \mathbf{z}^{\text{glb}}$ 

```

7 Comparison of SATURATE and optimal solutions

In this section we report the performance and computation time for the algorithms. These experiments assume frictionless contacts. We focus on the Q_∞ metric, which is more relevant to most applications and improves as the number of contacts is increased. We obtained five 3D models from the Princeton shape database [11], chosen fairly arbitrarily to be representative: airplane, shark, brain, gun, and vase, corresponding to model numbers 1147, 288, 540, 649, and 80, respectively. We randomly selected a subset of 100 faces from each mesh, whose centers would serve as candidate contact points. Reducing the number of candidates to 100 made it possible to compare the numerical results from the different objects. We ran four algorithms on each model: branch-and-bound, SATURATE, and two other non-optimal methods: (1) random selection, (2) “uniform selection”, which aims to choose a set of

contacts that are uniformly distributed in wrench space¹. The performance of all four algorithms is shown in Figure 2.

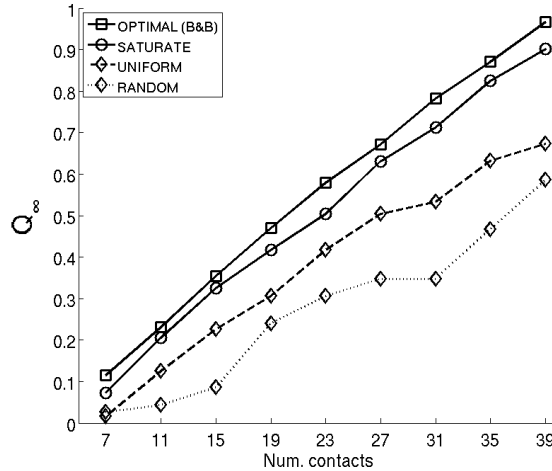


Fig. 2: Quality metric Q_∞ as the number of contact points is increased from 7, the minimum number of frictionless contacts necessary for force closure. The data were averaged over the five 3D models. Two simpler heuristics, UNIFORM and RANDOM are included for comparison.

As we show in Figure 3, SATURATE takes seconds to run, and the branch-and-bound takes minutes to run. Both algorithms were implemented in Matlab.² As expected, the run-time of SATURATE is roughly linear in the number of contacts selected, since it runs a greedy algorithm a fixed number of times. Surprisingly, branch-and-bound does not become more expensive as the number of contact points increases, even though the number of possible selections increases exponentially. Evidently, the algorithm is quite effective at restricting the search space, through eliminating subproblems with low upper bounds and by finding integer solutions to the relaxed subproblems.

In Figure 4 we show one of the models (a human brain) under two conditions: first with no gravity, second with a high amount of gravity. (Recall that gravity shifts the ball of wrenches that the grasp or fixture must resist.)

¹ In the uniform procedure, if $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k\}$ is the set of points (in wrench space) chosen already, then the index i of the next point is chosen as $\arg \max_i \min_j \text{dist}(\mathbf{w}_i, \mathbf{w}_j)$, i.e., the point that is furthest away from all of the previously chosen points.

² Source code related to this paper, including SATURATE and branch-and-bound, is available at <https://github.com/joschu/isrr2011grasping>

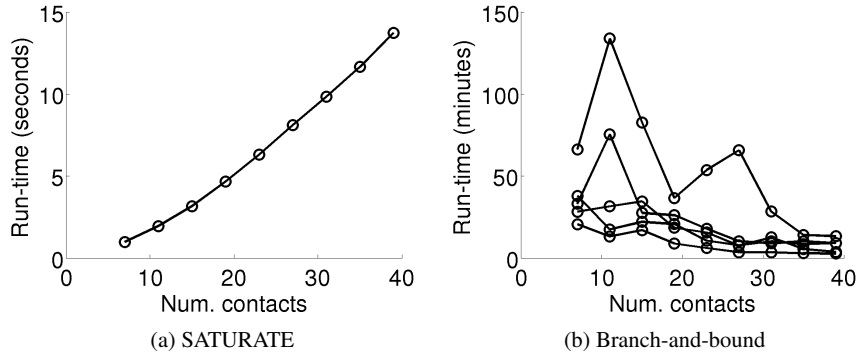


Fig. 3: Run-time of SATURATE and branch-and-bound on the five models. Note the different scales on the y-axes. The computation times for SATURATE are identical for each model (same number of candidate contact points), but the times vary greatly for branch-and-bound.

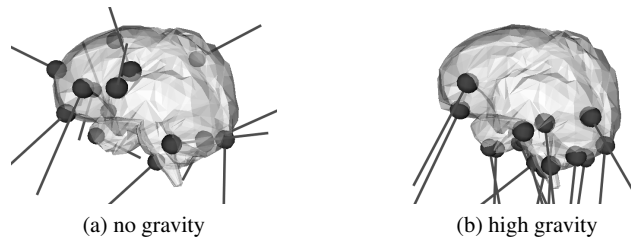


Fig. 4: Contact points chosen by SATURATE as gravity varies.

8 Application to towing

We will show that the methods of this paper can be applied to a different sort of force closure problem, in which we must arrange several aerial vehicles, such as helicopters, to tow a heavy object. Each helicopter may be attached to the object at multiple points with ropes. The helicopters must be able to resist disturbance wrenches that are applied to the object, but the total force that each helicopter can exert is limited. Assuming that the ropes that connect a given helicopter to a payload are all taut, the set of wrenches that helicopter can apply to the payload are a function of its position. Here, the problem is to optimize the position of the helicopters, to maximize the radius of the ball in wrench space that they can together resist.

Consider a single helicopter, which is attached to the payload with R ropes. For the i th rope, we define the following quantities: the force of tension is t_i , the force direction is $\hat{\mathbf{f}}_i$ (where $\|\hat{\mathbf{f}}_i\| = 1$), and the wrench per unit

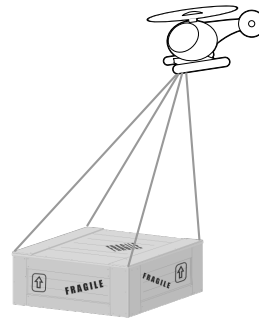


Fig. 5: Towing

force is $\mathbf{w}_i = \begin{pmatrix} \hat{\mathbf{f}}_i \\ \hat{\mathbf{f}}_i \times \mathbf{r}_i \end{pmatrix}$. Then the total force and wrench are

$$\mathbf{f}_{\text{total}} = \sum_i t_i \hat{\mathbf{f}}_i \quad (28)$$

$$\mathbf{w}_{\text{total}} = \sum_i t_i \mathbf{w}_i \quad (29)$$

The set of wrenches that the helicopter may apply to the object is

$$W = \{\mathbf{w}_{\text{total}} \mid \|\mathbf{f}_{\text{total}}\| \leq 1\} \quad (30)$$

where all the tensions $t_i \geq 0$. W is a convex set, and we can use the same formalism we developed earlier for calculating Q_∞ for a frictional grasp. We can efficiently evaluate the support function $h_W(\mathbf{y})$ for a disturbance \mathbf{y} .

As before, there is a set of disturbance wrenches that the towing arrangement must resist, which is not necessarily centered around the origin—it is centered around w_{load} , some combination of the wrench due to gravity and the anticipated inertial forces. By varying w_{load} , we see that different arrangements are optimal. In the case of a heavy load, where the helicopters can just barely lift the object, they must be positioned directly above the object. With a light load, a spread-out arrangement is optimal, since it is better for resisting horizontal or rotational disturbances.

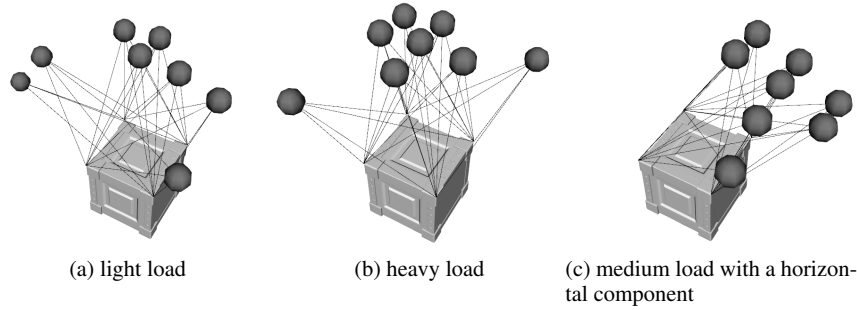


Fig. 6: Towing arrangements found by SATURATE for different loads. Gray spheres represent helicopters. Helicopter positions were selected from a set of points on a spherical surface.

9 Conclusions

We used some tools from convex geometry to find alternate formulas for the classic quality metrics. Since they are cheap to evaluate, these formulas may be useful in their own right for fast quality computation, especially in the frictional case. However, more interestingly, they reveal that the contact point selection problem can be viewed as a multi-objective submodular coverage problem, and SATURATE pro-

vides an efficient, near-optimal solution. We found that our algorithm can select a large number of contact points out of a larger number of candidates in a modest amount of time. We also provided a branch-and-bound algorithm for exact optimization, which is a couple orders of magnitude slower.

As far as we know, the SATURATE algorithm has exclusively been applied to problems of sensor placement. Our work shows that it can be used for actuator placement and may have other applications in engineering design. We must build our system with a small set of supports so it can survive a set of disturbances. Each disturbance corresponds to a cost function, which is submodular in the set of supports. In our work, the supports are contact points and the disturbances are the wrenches that might be applied to the object. Then, one uses SATURATE to find the best set of supports to handle the worst-case disturbance.

10 Acknowledgments

We thank James O'Brien for insightful discussions. This research was supported in part by the National Science Foundation (NSF) under award IIS-0905344.

Appendix

Proof of proposition 1

Let $\hat{r} = r_{\text{res}}(C)$ and $\tilde{r} = \min_{\|\mathbf{y}\|=1} h_C(\mathbf{y})$. First we show that $\hat{r} \leq \tilde{r}$ because the ball of radius \hat{r} lies inside of C . Then we show that $\tilde{r} \leq \hat{r}$ using the maximality of \hat{r} . Both of these arguments use the fact that for two convex sets A and B , $A \subseteq B$ if and only if $h_A \preceq h_B$ (i.e., $h_A(\mathbf{x}) \leq h_B(\mathbf{x})$ for all \mathbf{x}).

Let B_r be the ball of radius r around the origin. If $\hat{r} = r_{\text{res}}(C)$, then it follows that $B_{\hat{r}} \subseteq C$. It follows that $h_{B_{\hat{r}}}(\mathbf{y}) \leq h_C(\mathbf{y})$ for all \mathbf{y} . For $\|\mathbf{y}\| = 1$, $h_{B_{\hat{r}}}(\mathbf{y}) = \hat{r}$, so

$$\hat{r} \leq h_C(\mathbf{y}), \text{ for all } \mathbf{y} \text{ such that } \|\mathbf{y}\| \leq 1. \quad (31)$$

It follows that $\hat{r} \leq \tilde{r}$.

Next we'll prove the other direction of the inequality. It follows from the definition of \tilde{r} that

$$\tilde{r} \leq h_C(\mathbf{y}) \text{ for all } \|\mathbf{y}\| = 1 \quad (32)$$

Since $h_{B_{\tilde{r}}}(\mathbf{y}) = \tilde{r}$ for $\|\mathbf{y}\| = 1$, we have that

$$h_{B_{\tilde{r}}}(\mathbf{y}) \leq h_C(\mathbf{y}) \text{ for all } \|\mathbf{y}\| = 1. \quad (33)$$

Since the h is homogeneous, it follows that $h_{B_{\tilde{r}}} \preceq h_C$ so we have that $B_{\tilde{r}} \subseteq C$. By the maximality of \hat{r} , we must have $B_{\tilde{r}} \subseteq B_{\hat{r}}$. Thus $\tilde{r} \leq \hat{r}$.

Discretization error in formula for residual radius

We will consider a particular scheme for deterministically sampling the p -dimensional sphere (a subset of \mathbb{R}^{p+1}) and bound the error that results when one evaluates the support function $h_C(\mathbf{y})$ at only the sampled points \mathbf{y} to approximate its minimum. We take an $n \times n \times \dots \times n$ grid on every p -dimensional facet of the $p + 1$ -dimensional hypercube. This requires $(2p + 2)n^p$ points.

Let $\mathbf{y}_n = \arg \min h_C(\mathbf{y})$ be the optimal sampled point, and let $\mathbf{y}_* = \arg \min h_C(\mathbf{y})$ be the exact optimal point. Let $\mathbf{x}_n = \arg \max_{\mathbf{x} \in C} \mathbf{y}_n^T \mathbf{x}$, and let $\theta_{\mathbf{v}, \mathbf{w}}$ denote the angles between the vectors \mathbf{v} and \mathbf{w} .

$$\mathbf{y}_n^T \mathbf{x}_n = \|\mathbf{x}_n\| \cos \theta_{\mathbf{x}_n, \mathbf{y}_n} \quad (34)$$

$$\frac{d}{d\theta} \mathbf{y}_n^T \mathbf{x}_n = -\|\mathbf{x}_n\| \sin \theta_{\mathbf{x}_n, \mathbf{y}_n} \quad (35)$$

By the intermediate value theorem,

$$\mathbf{y}_n^T \mathbf{x}_n - \mathbf{y}_*^T \mathbf{x}_n = -\|\mathbf{x}_n\| \sin \tilde{\theta} \Delta \theta \quad (36)$$

where $\theta_{\mathbf{y}_n, \mathbf{x}_n} \leq \tilde{\theta} \leq \theta_{\mathbf{y}_*, \mathbf{x}_n}$, and $\Delta \theta = \theta_{\mathbf{y}_n, \mathbf{x}_n} - \theta_{\mathbf{y}_*, \mathbf{x}_n}$. Next, note that $h(\mathbf{y}_*) \geq \mathbf{y}_*^T \mathbf{x}_n$ so

$$h(\mathbf{y}_n) - h(\mathbf{y}_*) \leq \|\mathbf{x}_n\| |\sin \tilde{\theta} \Delta \theta| \quad (37)$$

The largest possible $\Delta \theta$ occurs at the nearby part of each face, between the vector $(1, 0, \dots, 0)$ and $(1, \frac{1}{2n}, \dots, \frac{1}{2n})$, where $\cos(\Delta \theta) = \sqrt{1 + p/4n^2}$. It follows that

$$|\Delta \theta| \leq \frac{\sqrt{p}}{2n} \quad (38)$$

Thus

$$h(\mathbf{y}_n) - h(\mathbf{y}_*) \leq \|\mathbf{x}_n\| |\sin \tilde{\theta} \Delta \theta| \quad (39)$$

$$\leq \|\mathbf{x}_n\| \frac{\sqrt{p}}{2n} \quad (40)$$

Note that $\tilde{\theta}$ is typically small, since $\arg \max_{\mathbf{x} \in C} \mathbf{y}_n^T \mathbf{x}$ is often nearly parallel to \mathbf{y}_n . So the error is typically smaller than this bound.

Submodular saturation algorithm

Here we briefly describe the theoretical guarantees on SATURATE. SATURATE finds solutions to problems where we are simultaneously trying to optimize a collection of submodular objectives F_i :

$$\max_{S' \subset S} \min_i F_i(S'), \quad \text{subject to } |S'| \leq k. \quad (41)$$

Krause et al. [8] proposed SATURATE, and they showed that if we run it on the objective (41) and request a αk -element solution S' , it is guaranteed to return a solution that is better than the optimal k -element solution, where

$$\alpha = 1 + \log\left(\max_{s \in S} \sum_i F_i(s)\right). \quad (42)$$

This bound applies when the functions F_i take integer values. Thus to apply this result to a general problem of the form in Equation (41), we must typically rescale and then round the objective functions. In the present problem, we can rescale and round the values $h_{w_i}(y_j)$, or, in the frictionless case $y_j^T w_i$. Note that because of rounding error, the αk -element solution found by SATURATE might be slightly worse (on the original objective) than the optimal k -element solution.

References

1. Ch. Borst, M. Fischer, and G. Hirzinger. A fast and robust grasp planner for arbitrary 3D objects. *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, 3(May):1890–1896, 1999.
2. S.P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge Univ Pr, 2004.
3. R. C. Brost and R. R. Peters. Automatic Design of 3-D Fixtures and Assembly Pallets. *The International Journal of Robotics Research*, 17(12):1243–1281, December 1998.
4. R.C. Brost and K.Y. Goldberg. A complete algorithm for synthesizing modular fixtures for polygonal parts. *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 535–542, 1994.
5. C. Ferrari and J. Canny. Planning optimal grasps. *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2290–2295, 1992.
6. B Grünbaum and V Klee. *Convex polytopes*. Springer Verlag, 2003.
7. David Kirkpatrick, Bhubaneswar Mishra, and Chee-Keng Yap. Quantitative Steinitz’s Theorems with Applications to Multifingered Grasping. *Discrete and Computational Geometry*, 7(1), 1992.
8. Andreas Krause and H Brendan Mcmahan. Robust Submodular Observation Selection. *Journal of Machine Learning Research*, 9:2761–2801, 2008.
9. B. Mishra. Grasp Metrics: Optimality and Complexity. In *Algorithmic Foundations of Robotics*, 1995.
10. Nancy S. Pollard. Closure and Quality Equivalence for Efficient Synthesis of Grasps from Examples. *The International Journal of Robotics Research*, 23(6):595–613, June 2004.
11. P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. *Proceedings Shape Modeling Applications, 2004.*, 08540:167–388, 2004.
12. M. Teichmann and B. Mishra. Probabilistic Algorithms for Efficient Grasping and Fixturing. *Algorithmica*, 26(3-4):345–363, March 2000.
13. M.Y. Wang. An optimum design for 3-D fixture synthesis in a point set domain. *IEEE Transactions on Robotics and Automation*, 16(6):839–846, 2000.