Planning of Power Grasps Using Infinite Program Under Complementary Constraints

Zherong Pan[1], Duo Zhang[2], Changhe Tu[2], Xifeng Gao[1]

***Abstract*—We propose an optimization-based approach to plan power grasps. Central to our method is a reformulation of grasp planning as an infinite program under complementary constraints (IPCC), which allows contacts to happen between arbitrary pairs of points on the object and the robot gripper. We show that IPCC can be reduced to a conventional finite-dimensional nonlinear program (NLP) using a kernel-integral relaxation. Moreover, the values and Jacobian matrices of the kernel-integral can be evaluated efficiently using a modified Fast Multipole Method (FMM). We further guarantee that the planned grasps are collision-free using primal barrier penalties. We demonstrate the effectiveness, robustness, and efficiency of our grasp planner on a row of challenging 3D objects and high-DOF grippers, such as Barrett Hand and Shadow Hand, where our method achieves superior grasp qualities over competitors.**

***Index Terms*—Grasping, optimization, infinite program and complementary constraints**

I. Introduction

Grasp planning remains a fundamental and perennial problem, although intense research efforts have been invested over the past decades. A vast majority of prior works view grasp planning as a non-smooth, noise-corrupted search problem, and rely on model-free stochastic optimizations, such as simulated annealing [1], Bayesian optimization [2], and multi-armed bandits [3], to optimize the grasp quality. Although these methods make minimal assumptions on geometries of objects and kinematics of grippers, they are typically sample-intensive. Instead, several relatively recent works [4, 5, 6] demonstrate the advantage of model-based approaches in terms of fast convergence [7], amenability to machine learning [6], and global optimality [8, 9, 10]. Model-based approaches utilize certain properties of grasp metrics, object shapes, or gripper types, such as derivatives [6], submodularity [9], and monotonicity [10], to guide the search of optimal grasps and achieve improved efficacy.

Despite their various advantages, model-based approaches are relatively less used due to a limited robustness and generality in several ways. Most model-based algorithms [6, 8, 9, 10] are limited to precision grasps by pre-sampling a small set of contact points either on the gripper or the object. In comparison, model-free, sampling-based approaches are agnostic to contact points and can easily handle power grasps. Moreover, some model-based approaches [9, 11] only plan grasp points without considering gripper fesasibility. Other methods [10, 8, 7] can account for gripper kinematics, but they either resort to model-free sampling-based method [10], require a long computational time [8], or cannot handle complex object shapes [7, 8].

If we switch gears from grasp planning to general contact-rich path planning, there has been numerous efforts to sidestep the above limitations. In particular, contact-implicit trajectory optimization [12, 13, 14] generates trajectories with unprecedented complexity by allowing a numerical optimizer to make or break contact points. In this paper, we propose to borrow these techniques and design a model-based grasp planner without using pre-sampled contact points. Unlike contact-rich path planning where contacts only happen on robot end-effectors, we propose to consider every pair of points on the object and the gripper for potential contacts, and allow the optimizer to determine their status. However, there are infinitely many such point pairs, for which a naïve discretization is computationally intractable.

Main Result: We study the grasp planning problem through the lens of IPCC formulation. We introduce a pair of complementary constraints between each pair of points on the object and the gripper. Complementary constraints allow the optimizer to jointly choose contact positions, forces, and gripper's kinematic poses, during which the contact state is implicitly determined. IPCC is one of the most challenging optimization problems that are typically solved by constraint approximation or instantiation [15]. However, we show that, in the special case of grasp planning with Q_∞ metric objective function, IPCC reduces to a standard NLP via the technique of kernel-integral relaxation, which reduces an infinite set of constraints to a single constraint involving an surface integral of a kernel function. Moreover, we adapt the Fast Gauss Transform (FGT) [16], a variant of Fast Multiple Method (FMM) [17], to efficiently evaluate the surface integrals and its Jacobian matrix. This technique leads to significantly speed-up over brute-force evaluation, as shown in Figure 4. Our new approach provides much larger solution space than prior works and inherently allow both precision and power grasps. Finally, we use log-barrier functions and robust line-search scheme to guarantee the satisfaction of penetration- and self-collision free constraints. We summarize the new features of our method in Table I.

Our grasp planning method is fast and robust, which has been verified by batch processing 20 objects with various geometrical and topological complexities using a 3-fingered, 15-DOF Barrett Hand and a 5-fingered, 24-DOF Shadow Hand. Compared with prior state-of-the-arts, our algorithm achieves considerably less computational time than [8], higher robustness to penetrations than [7], or higher quality of grasps than [6].

Manuscript received: July 29, 2021; Revised: October 17, 2021; Accepted: November 13, 2021.

This paper was recommended for publication by Editor Hong Liu upon evaluation of the Associate Editor and Reviewers' comments.

[1] Lightspeed & Quantum Studio, Tencent America. {zherong.pan.usa, gxf.xisha}@gmail.com. [2]Department of Computer Science, Shandong University. galenzhang@mail.sdu.edu.cn. Duo Zhang and Changhe Tu are supported in part by NSFC No. 61772318.

Digital Object Identifier (DOI): see top of this page.

Method	Non-Convex	Power Grasps	Collision-Free	Gripper
[6]	✓			✓
[8, 18]	✓		✓	✓
[9, 19]	✓			
[7]		✓		✓
Ours	✓	✓	✓	✓

TABLE I: We compare representative model-based grasp planners in terms of handling complex non-convex objects, planning power grasps, ensuring collision-free, planning both grasp qualities and gripper poses. Note that some methods [7, 6] consider collision-free constraints but the underlying numerical model cannot ensure the constraints are satisified.

II. Related Work

We briefly review related works in model-free and model-based grasp planning. We then provide background on contact-implicit path planning and fast multipole method.

Model-free grasp planners treat a grasp simulator as a black-box. All the existing model-free planners are sampling-based and inherit celebrated completeness and optimality properties [20, 21]. Various techniques have been proposed to improve their efficacy. Early works [22, 1] reduce the dimension of search space by limiting the DOF of a gripper. More recent approaches utilize correlation between samples and formulate the grasp planning in Bayesian optimization [2] or multi-arm bandits [3] settings. Model-free method features a high versatility in generalizing to all kinds of 3D objects, gripper modalities, and types of grasps (see e.g. [23]). These methods have recently witnessed significant progress thanks to the use of data-driven techniques, e.g. [24, 25], but this topic is out of the scope of this work.

Model-based grasp planners exploit additional assumptions on a grasp simulator or use additional outputs from the simulator to further improve the planning performance. For example, Miller *et al.* [26] assumed the 3D objects resemble some simple geometric primitives and Dai *et al.* [7] assumed the 3D objects are convex. Other works make assumptions on the grasp quality metrics, Hang *et al.*, Liu *et al.* [10, 8] relies on the grasp metric being monotonic and Schulman *et al.* [9] proved that $Q_{1,\infty}$ metrics are submodular and used this property to approximate optimal grasps with bounded sub-optimality. Finally, a large body of model-based planners [7, 5, 6, 8] formulate the problem as gradient-based numerical optimization and require a grasp simulator to be differentiable.

Contact-implicit optimization is a key component in rigid body simulators such as [27] via the position-force complementary conditions. [28, 12, 13, 29] has proven capable of generating complex robot motion trajectories from trivial initialization. Central to these formulations is the use of position-force complementary conditions as hard constraints in a trajectory optimizer. However, solving these optimization problems with complementary constraints is notoriously difficult. Some methods [30, 31] need to build search trees to explore the contact state for each pair of potentially contacting objects. Our method can be interpreted as a generalization of these techniques to grasp planning. The main application of contact-implicit optimization lies in legged robotics, where contacts are assumed to only happen on a few robot end-effectors. However, to enable both precision and power grasps, we need to consider all pairs of potential contact points, leading to an infinite number of decision variables. We emphasize that two prior works [28, 29] lifted the contact-on-end-effector assumption, and allows contacts to happen anywhere on the robot. However, these methods rely on smooth contact models and do not pertain (self-)collision-free guarantee.

Fast multipole method finds most applications in large scale numerical simulation of N-body problems using Boundary Element Methods (BEM), where each pair of two bodies have influences on each other. As a result, summing up the total influences on all bodies incur a computational cost of $\mathcal{O}(N^2)$. FMM reduces this cost to $\mathcal{O}(N\log(N))$ or even $\mathcal{O}(N)$ by aggregating bodies into clusters and approximating the cluster-wise influences using truncated Taylor or Laurent series, while the approximation error can be arbitrarily bounded (see [17] for more details). A major advantage of BEM over Finite Element Methods (FEM) [32] is that BEM only uses a surface mesh while FEM requires a volume mesh. This property has been exploited in [11] to account for object deformations under grasp. In this work, we show that infinite complementary constraints can be replaced with a single constraint involving a kernel integration, whose value and Jacobian matrix can be evaluated efficiently using the FGT [16].

III. Grasp Planning as IPCC

We first review the basics of grasp planning using notations summarized in Table II. We assume that there is an object with surface S_o and a robot surface S_r determined by the robot's configuration θ, denoted as $S_r(\theta)$, both of which are 2D manifolds. A robot can apply a wrench $w(x)$ on $x \in S_o$ if and only if x is in contact or $x \in S_r$. The wrench is associated with a contact force $f(x) \in \mathcal{C}(x)$ by the relationship: $w(x) = \big(f(x), x \times f(x)\big)^T$, where $\mathcal{C}(x)$ is the friction cone at x defining feasible forces, $x\times$ is the cross-product matrix, and we assume the object's center-of-mass is placed at the origin. When the object is undergoing external wrench w_o, the robot must immobilize the object via an counteracting wrench w_{sum} to maintain a grasp, defined as $w_{sum} \triangleq \int_{S_o} w(x)dx$. The quality of a grasp measured using Q_∞ metric is defined as:

$$Q_\infty \triangleq \begin{cases} \min\limits_{\|w_o\|=1} \max\limits_{f(x)\in\mathcal{C}(x)} & \langle w_o, w_{sum}\rangle \\ \text{s.t.} & \langle n(x), f(x)\rangle \le 1 \quad \forall x \in S_o \end{cases}, \tag{1}$$

where $n(x)$ is the unit inward normal at $x \in S_o$ and $\langle \bullet, \bullet \rangle$ is the inner-product. For brevity, we omit "$\forall x \in S_o$" hereafter. Intuitively, Q_∞ equals to the largest magnitude of external wrench that the robot can counteract along all possible directions, using bounded grip force. Note that the above integral must be well-defined because the constraint $\langle n(x), f(x)\rangle \le 1$ makes the integrand bounded and the domain of integral is also bounded. In this paper, we consider the following discretized Q_∞ by limiting w_o to a finite set $w_o^1, \cdots, w_o^D$:

$$Q_\infty \triangleq \begin{cases} \min\limits_{d=1,\cdots,D} \max\limits_{f^d(x)\in\mathcal{C}(x)} & \langle w_o^d, w_{sum}\rangle \\ \text{s.t.} & \langle n(x), f^d(x)\rangle \le 1 \quad \forall d = 1, \cdots, D \end{cases}, \tag{2}$$

where f^d is the contact force to resist external wrench along w_o^d. Again, we omit "$\forall d = 1, \cdots, D$" hereafter for brevity. Combining the definition of Q_∞ and the force-position com-

TABLE II: Symbol Table.

Variable	Definition	Variable	Definition
S_o	object surface	$G^d(\theta)$	resisting wrench for direction d
S_r	robot surface	K	kernel function
θ	robot configuration	$L, L_{o,r}$	collision avoidance term
x	a point on object	$\{P, R, n, n_0\}^{pq}$	separating plane
y	a point on robot in global coordinates	$S_r^l, V(l)$	lth link, number of vertices
R, t	local-to-global rotation, translation	L	number of links
y^l	a point on robot in local coordinates	ϕ, ρ	merit function, constraint weight
$\mathcal{C}$	feasible force cone	γ	constraint weight in merit function
$n(x)$	inward normal on x	r	radius of Poisson's disk
f, w_{sum}	force,sum of wrench on object	N, M	number of source, target points
f^d, w_o^d	dth external force,wrench	c, b	center point of source, target box
Q_∞	grasp quality metric	H_n, h_n, h_n^2	Hermite functions
D	number of sampled directions	$A_n, B_n^l, C_n, \{E, F, H, I\}_m$	FGT coefficients
d_r	distance to the robot	n_0	number of truncated terms in FGT
α	complementary relaxation parameter	$\mathcal{B}$	clustering box of FGT
$g^d(x)$	resisting wrench on x	$\mathcal{S}(y)$	source strength

plementary condition, a grasp planning problem is defined by the following IPCC:

$$\begin{aligned} &\underset{\theta, f^d(x)\in\mathcal{C}(x)}{\text{argmax}} \quad Q_\infty \\ &\text{s.t. } 0 \le \left\langle n(x), f^d(x)\right\rangle \perp d_r(x,\theta) \ge 0, \end{aligned} \tag{3}$$

which inherently handles power grasps using infinitely many variables $f^d(x)$, each involved in a complementary constraint dictating that only points in contact can impose non-zero forces on the object. Here $d_r(x,\theta)$ is the distance between x and the robot surface at configuration θ.

IV. Kernel-Integral Reduction

In this section, we propose a practical reformulation of Equation 3 as a standard NLP by using the relaxed complementary constraint [33]. Each complementary constraint is equivalent to three inequalities:

$$\begin{cases} \left\langle n(x), f^d(x)\right\rangle \ge 0 \\ d_r(x,\theta) \ge 0 \\ \left\langle n(x), f^d(x)\right\rangle d_r(x,\theta) \le 0, \end{cases} \tag{4}$$

and Hoheisel *et al.* [33] proposed to replace the third inequality with $\left\langle n(x), f^d(x)\right\rangle d_r(x,\theta) \le \alpha$ for some small, positive relaxation constant α, and then use Sequential Quadratic Programming (SQP) to satisfy a sequence of relaxed, differentiable constraints with a monotonically decreasing series of α that tends to zero. However, SQP cannot handle our relaxed form due to non-differentiable term d_r, the distance between a point and a general surface of the robot. To sidestep this incompatibility, we rewrite $d_r(x,\theta) = \min_{y\in S_r(\theta)} \|x-y\|$ and replace each relaxed complementary constraint with an infinite set:

$$\left\langle n(x), f^d(x)\right\rangle \|x-y\| \le \alpha \quad \forall y \in S_r(\theta). \tag{5}$$

With a slight rearrangement and by introducing a so-called kernel function $K(\bullet,\alpha) \triangleq \alpha/\bullet$, each complementary constraint takes the form:

$$\begin{aligned} &\left\langle n(x), f^d(x)\right\rangle \le K(\|x-y\|,\alpha) \quad \forall y \in S_r(\theta) \\ &0 \le \left\langle n(x), f^d(x)\right\rangle \le 1, \end{aligned} \tag{6}$$

where we have merged the requirement of Q_∞ that normal force magnitude is less than 1. We show that, as $\alpha \to 0$, the infinite set of constraint Equation 6 is equivalent to the following single constraint for a specific choice of kernel function $K(\bullet,\alpha)$:

$$\left\langle n(x), f^d(x)\right\rangle \le \int_{S_r} K(\|x-y\|,\alpha)dy. \tag{7}$$

Lemma IV.1. *Suppose we choose:*

$$K(\bullet,\alpha) = \frac{-1}{(2\pi\log\alpha)(\bullet^2+\alpha^2)} \quad \textit{s.t.} \quad \alpha < 1, \tag{8}$$

and the constraint Equation 7 is satisfied for a monotonic sequence $0 \le \alpha^k \to 0$:

$$0 \le \left\langle n(x^k), f^d(x^k)\right\rangle \le \int_{S_r} K(\|x^k-y\|,\alpha^k)dy, \tag{9}$$

and there is a convergent subsequence that tends to x^*, then we have $\left\langle n(x^*), f^d(x^*)\right\rangle d_r(x^*,\theta) \le 0$ and $\left\langle n(x^*), f^d(x^*)\right\rangle \le 1$. (Here the superscript $\bullet^k$ means the solution to Equation 3 when $\alpha = \alpha^k$.)

Proof. Without a loss of generality, we can assume the entire sequence x^k is convergent to x^*. **Case I:** If $d_r(x^*,\theta) > 0$, then by the choice of kernel function we have $K(\|x^k-y\|,\alpha^k) \to 0$ and Equation 7 implies $\left\langle n(x^*), f^d(x^*)\right\rangle = 0$. **Case II:** If $d_r(x^*,\theta) = 0$, then there is a unique point $y^* \in S_r$ such that $\|x^*-y^*\| = 0$ and the integral is singular at y^*. The integral is thereby nonzero only within an infinitesimal disk around y^* with radius δr (δr is a positive constant independent of k). By changing the integral under polar coordinates, we have:

$$\begin{aligned} &\lim_{\alpha\to 0}\int_0^{\delta r} \frac{-r}{\log\alpha(r^2+\alpha^2)}dr \\ &= \lim_{\alpha\to 0}\frac{\log\alpha - \frac{1}{2}\log(\delta r^2+\alpha^2)}{\log\alpha} = 1. \end{aligned} \tag{10}$$

We conclude that Equation 7 is an appropriate equivalence of Equation 6 in the limit of α. □

Note that K does not need to take the exact form as in Lemma IV.1 in practice. This is because Lemma IV.1 only considers the limiting behavior of K when $\alpha \to 0$, but we would terminate optimization with a finite, positive α due to limited machine precision. Our experiments show that it suffice to choose any $K(\|x-y\|,\alpha)$ that decay quickly as $\|x-y\| \to \infty$. Indeed, we find that choosing K to be an exponential function would lead to an efficient algorithm for evaluating the integral in Equation 7 and refer readers to Section V for more details.

Next, we show that f^d has closed-form solution. We notice the inner max function in Equation 2 can be moved into the integral, giving:

$$\min_{d=1,\cdots,D}\int_{S_o}\max_{f^d(x)\in\mathcal{C}(x)}\left\langle w_o^d, w^d(x)\right\rangle dx, \tag{11}$$

where the integrand is the only term related to $w^d(x)$ and $w^d(x)$ is positively proportional to Q_∞. If we fix all other variables, $f^d(x)$ is the solution of the following subproblem:

$$\begin{aligned} &\underset{f^d(x)\in\mathcal{C}(x)}{\text{argmax}} \quad \left\langle w_o^d, w^d(x)\right\rangle \\ &\text{s.t. } 0 \le \left\langle n(x), f^d(x)\right\rangle \le \int_{S_r} K(\|x-y\|,\alpha)dy. \end{aligned} \tag{12}$$

Using a similar reasoning as [6, 9], the solution to Equation 12 has a closed form:

$$f^d(x) = g^d(x)\int_{S_r} K(\|x-y\|,\alpha)dy \tag{13}$$

$$g^d(x) \triangleq \begin{cases} \underset{f^d(x)\in\mathcal{C}(x)}{\text{argmax}} & \left\langle w_o^d, w^d(x)\right\rangle \\ \text{s.t.} & 0 \le \left\langle n(x), f^d(x)\right\rangle \le 1 \end{cases}, \tag{14}$$

and we refer readers to [9] for the derivation of the expression of $g^d(x)$, which is the support function of convex set $\{f^d(x) \in \mathcal{C}(x) | 0 \le \langle n(x), f^d(x) \rangle \le 1\}$ and endows a closed-form expression. When plugged into Equation 3, the closed-form solution already incorporates the relaxed complementary constraints and eliminates all the complementary variables, thus reducing the IPCC to the following standard NLP:

$$\begin{aligned} \underset{\theta}{\text{argmax}}\ Q_\infty &\triangleq \min_{d=1,\cdots,D} G^d(\theta) \\ G^d(\theta) &\triangleq \int_{S_o} g^d(x) \int_{S_r} K(\|x-y\|, \alpha) dy dx \\ \text{s.t. } & d_r(x,\theta) \ge 0, \end{aligned} \tag{15}$$

which provides a variational explanation of Q_∞ that allows any point on the robot surface to make contact with any other point on the object, thereby unifying precision and power grasps. The choice of grasp points is implicitly encoded in the double integral over the object and robot surfaces. We will show that such integrals can be approximately efficiently using FMM.

A. Guaranteed (Self-)Collision-Free

Equation 15 is still semi-infinite due to the infinitely many collision constraints: $d_r(x,\theta) \ge 0$. In prior work [6], the collision-free constraint $d_r \ge 0$ is imposed using soft penalty terms, which is not guaranteed to be satisfied. We propose to ensure collision-free via the log-barrier function:

$$L_o(\theta) = -\int_{S_o} \log\left[d_r(x,\theta)\right] dx. \tag{16}$$

Using a line-search algorithm, we can guarantee that L_o takes a finite value throughout the optimization, which in turn implies collision-free between the robot and the gripper. In practice, we assume the object is provided as a point cloud and replace the integral of S_o with a summation over each point. L_o is differentiable as shown in [6] and the evaluation of summation can be accelerated using a bounding volume hierarchy and log-barrier function with local support. Note that, although we introduce our method for point-cloud approximations of the objects, a similar technique can be used to avoid collisions for objects represented by triangle meshes as done in [34].

We further consider self-collision assuming each robot link takes a convex shape. Assuming that the robot surface is decomposed into L links $S_r = \bigcup_{l=1}^{L} S_r^l$ where each S_r^l is the convex hull of $V(l)$ vertices $\{y_1^l(\theta), \cdots, y_{V(l)}^l(\theta)\}$. Then a separating plane $P^{pq}(y) = \langle n^{pq}, y \rangle + n_0^{pq}$ could be introduced to avoid collision between a pair of links $S_r^{p,q}$, where n^{pq}, n_0^{pq} are plane normal and offset. The log-barrier function for self-collision takes the following form:

$$\begin{aligned} & L_r(\theta, n^{pq}, n_0^{pq}) \\ =& -\sum_{1\le p<q\le L} \sum_{i=1}^{V(p)} \log\left[\left\langle n^{pq}, y_i^p(\theta)\right\rangle + n_0^{pq}\right] \\ & -\sum_{1\le p<q\le L} \sum_{j=1}^{V(q)} \log\left[-\left\langle n^{pq}, y_j^q(\theta)\right\rangle - n_0^{pq}\right]. \end{aligned} \tag{17}$$

We propose to use block coordinate descend algorithm and interleave the optimization for θ and P^{pq}, so that the optimization for each plane P^{pq} is independent. To ensure the plane normal has unit length, we use reparameterize $n^{pq} = R^{pq} e$ with $R^{pq} \in \mathcal{SO}(3)$ represented using Rodriguez formula and e being an arbitrary unit vector (see [35] for analytic derivatives of the Rodriguez rotation formula).

B. Simplified SQP for Minimizing Q_∞

Putting everything together, we recast NLP (Equation 15) as an unconstrained optimization:

$$\underset{\theta, P^{pq}}{\text{argmin}}\ L(\theta) - Q_\infty(\theta) \quad L \triangleq L_o + L_r, \tag{18}$$

which can be solved using a simplified SQP algorithm. The non-differentiable $\min$ operator in Q_∞ can be replaced with hard constraints:

$$\underset{\theta, P^{pq}, Q}{\text{argmin}}\ L(\theta) - Q \quad \text{s.t. } Q \le G^d(\theta), \tag{19}$$

where Q is a slack variable. We show in our extended report [36] that SQP takes a simplified form when solving Equation 19 by observing that the QP subproblem is always feasible.

V. Numerical Integral Evaluation

Although we have derived the standard NLP Equation 19, the integrals involved in G^d and $L(\theta)$ do not have analytic expressions and need to be evaluated numerically. The double integral involved in G^d is known as Fredholm integral of the first kind, where the integrand is a multiplication of a source term $g^d(x)$ and a kernel function $K(\|x - y(\theta)\|, \alpha)$ that is singular when x is close to $y(\theta)$ and decay quickly as the distance increases. An intuitive method to discretize G^d would sample the two surfaces S_o, S_r with dense set of N points $\{x\} \in S_o$ and M points $\{y(\theta)\} \in S_r(\theta)$ using Poisson disk sampling with radius r and approximate G^d with double integral:

$$G^d \approx (\pi r^2)^2 \sum_x g^d(x) \sum_y K(\|x-y\|, \alpha), \tag{20}$$

which incurs a cost of $\mathcal{O}(NM)$. We introduce a modified FGT [16], a variant of FMM that can be applied if K is chosen to be $K(\bullet, \alpha) = \exp(-\bullet^2/\alpha)$. The standard FGT would only compute G^d and we derive extra equations to evaluate $\partial G^d / \partial \theta$ as required by SQP. (We use the same set of samples to discretize the integral in Equation 16, the computational cost of which is $\mathcal{O}(N)$. Compared with G^d, the cost to evaluate Equation 16 is marginal.)

We use x to denote a point on the object and y denotes a point on the gripper. As illustrated in Figure 1, FGT first cluster all the sampled points into boxes of side length $2\sqrt{\alpha}$, where each of $\mathcal{B}_x$ and $\mathcal{B}_y$ denotes a box that contains some point x, y, respectively. For each $\mathcal{B}_y$, FGT first uses Multipole-to-Multipole (M2M) step to approximate their contribution (to the integral) via Hermite expansion. Then for each $\mathcal{B}_{x,y}$ pair, FGT uses Multipole-to-Local (M2L) step to transfer the contribution from $\mathcal{B}_y$ to $\mathcal{B}_x$. Finally, FGT uses Local-to-Local

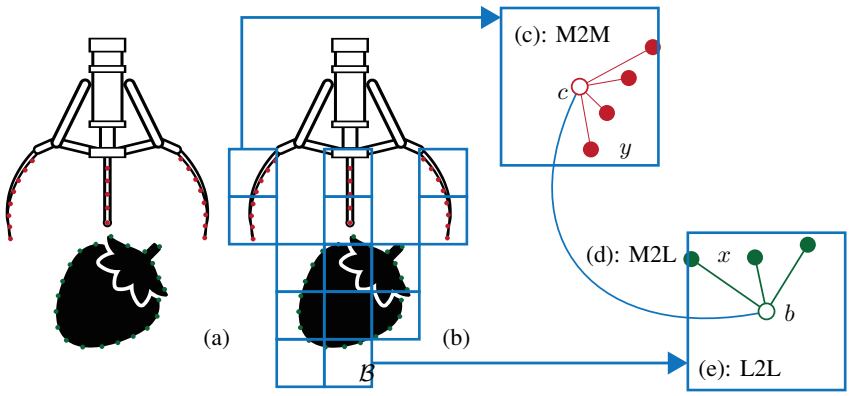

Fig. 1: We illustrate FGT applied to grasp planning. (a): We sample possible contact points both on the gripper (red) and the object surface (green). (b): The number of sample points is large and we cluster them into axis-aligned boxes $\mathcal{B}$ (blue). FGT works in three steps. (c): M2M step substitutes the contributions (to G^d) of source point y with the center point c using Hermite expansion (red line). (d): M2L step substitutes the contributions of center point c with the center point b using Taylor expansion (blue line). (e): L2L step evaluates G^d for target point x around b (green line).

(L2L) step to distribute the contribution from the $\mathcal{B}_x$ to each x.

A. M2M Step

Assuming x, y are two 1D points, the FGT is based on the Hermite expansion of exponential function:

$$\exp(-(\frac{y-x}{\sqrt{\alpha}})^2) = \sum_{|n|=0}^{\infty} \frac{1}{n!}(\frac{c-y}{\sqrt{\alpha}})^n h_n(\frac{x-c}{\sqrt{\alpha}}), \tag{21}$$

where h_n are Hermite polynomials and c is the center point of $\mathcal{B}_y$. If x, y, c are 3D points, then we use subscript to denote the coordinate index and the expansion takes the same form as above but n is a vector $\left(n_1, n_2, n_3\right)$. We have $n! \triangleq n_1!n_2!n_3!$, $|n| \triangleq n_1 + n_2 + n_3$, $r^n \triangleq \Pi_{i=1}^3 r_i^{n_i}$, and $h_n(r) \triangleq \Pi_{i=1}^3 h_{n_i}(r_i)$. The gradient with respect to y_i has the following Hermite expansion:

$$\begin{aligned} &\frac{\partial}{\partial y_j}\left[\exp(-(\frac{y-x}{\sqrt{\alpha}})^2)\right] = \frac{2(x_j - y_j)}{\alpha}\exp(-(\frac{y-x}{\sqrt{\alpha}})^2) \\ &= \sum_{|n|=0}^{\infty} \frac{-2}{\sqrt{\alpha} n!}\left[(\frac{c-y}{\sqrt{\alpha}})^{n+e_j} h_n(\frac{x-c}{\sqrt{\alpha}}) + (\frac{c-y}{\sqrt{\alpha}})^n h_n^j(\frac{x-c}{\sqrt{\alpha}})\right], \end{aligned} \tag{22}$$

where $h_n^j(r) \triangleq r_j h_n(r)$. The two above expansions form the Multipole-to-Multipole (M2M) step of FGT. If there is a set of points $y \in \mathcal{B}_y$ around a center point c, then we have:

$$\begin{aligned} &\sum_{y \in \mathcal{B}_y} \mathcal{S}(y)\exp(-(\frac{y-x}{\sqrt{\alpha}})^2) = \sum_{|n|=0}^{\infty} A_n h_n(\frac{x-c}{\sqrt{\alpha}}) \\ &A_n \triangleq \sum_{y \in \mathcal{B}_y} \mathcal{S}(y)\frac{1}{n!}(\frac{c-y}{\sqrt{\alpha}})^n. \end{aligned} \tag{23}$$

Similarly for the gradient, we have:

$$\begin{aligned} &\sum_{y \in \mathcal{B}_y} \mathcal{S}(y)\frac{\partial}{\partial y_j}\left[\exp(-(\frac{y-x}{\sqrt{\alpha}})^2)\right] \\ &= \sum_{|n|=0}^{\infty} B_n^j h_n(\frac{x-c}{\sqrt{\alpha}}) + \sum_{|n|=0}^{\infty} C_n h_n^j(\frac{x-c}{\sqrt{\alpha}}) \\ &B_n^j \triangleq \sum_{y \in \mathcal{B}_y} \mathcal{S}(y)\frac{-2}{\sqrt{\alpha} n!}(\frac{c-y}{\sqrt{\alpha}})^{n+e^j} \quad C_n = \frac{-2}{\sqrt{\alpha}} A_n, \end{aligned} \tag{24}$$

where $\mathcal{S}(y)$ is some y-dependent coefficients. The M2M step involves dividing the space into a set of axis-aligned boxes $\mathcal{B}_y$ with side length $2\sqrt{\alpha}$. For all the source points y belonging to a $\mathcal{B}_y$, M2M identifies their contributions with a single center point c using Hermite expansion (Equation 23 and Equation 24). FGT only retains terms with $n \le n_0$, where n_0 is chosen to ensure error is small than a user chosen threshold (see [16] for more details).

B. M2L Step

The center points c can still be faraway from target points x. M2L step identifies the contributions of center points c with some other points b that is close to target points using Taylor expansion. A Hermite expansion has the following equivalent Taylor expansion:

$$\begin{aligned} &\sum_{|n|=0}^{\infty} A_n h_n(\frac{x-c}{\sqrt{\alpha}}) = \sum_{|m|=0}^{\infty} E_m(\frac{x-b}{\sqrt{\alpha}})^m \\ &E_m \triangleq \frac{(-1)^{|m|}}{m!}\sum_{|n|=0}^{\infty} A_n h_{n+m}(\frac{c-b}{\sqrt{\alpha}}). \end{aligned} \tag{25}$$

For the gradient, we have:

$$\begin{aligned} &\sum_{y \in \mathcal{B}_y} \mathcal{S}(y)\frac{\partial}{\partial y_j}\left[\exp(-(\frac{y-x}{\sqrt{\alpha}})^2)\right] \\ &= \sum_{|m|=0}^{\infty}\left[H_m(\frac{x-b}{\sqrt{\alpha}})^{m+e_j} + (F_m + I_m)(\frac{x-b}{\sqrt{\alpha}})^m\right] \\ &F_m \triangleq \frac{(-1)^{|m|}}{m!}\sum_{|n|=0}^{\infty} B_n^j h_{n+m}(\frac{c-b}{\sqrt{\alpha}}) \\ &H_m \triangleq \frac{(-1)^{|m|}}{m!}\sum_{|n|=0}^{\infty} C_n h_{n+m}(\frac{c-b}{\sqrt{\alpha}}) \\ &I_m \triangleq \frac{(-1)^{|m|+1}}{m!}\sum_{|n|=0}^{\infty} C_n h_{n+m}^j(\frac{c-b}{\sqrt{\alpha}}). \end{aligned} \tag{26}$$

Again we only retain all the terms with $m \le n_0$. The M2L step involves dividing the space into another set of axis-aligned boxes $\mathcal{B}_x$ with side length $2\sqrt{\alpha}$. For each pair of boxes with center points c, b, M2L transfers the contribution from c to b (Equation 25 and Equation 26). This only needs to be done for pairs of boxes that are certain distances away.

C. L2L Step

After substituting the center of expansion from b to c, L2L step evaluates G^d around some target point x contained in a box, $\mathcal{B}_x$, with center point b using (Equation 25 and Equation 26). In summary, the cost of evaluating each G^d is $O(N+M)$ by setting $\mathcal{S}(y) = (\pi r^2)^2$. To evaluate $\partial G^d / \partial \theta$, we assume that the rigid object is an articulated body so that $y(\theta) = R(\theta)y^l + t(\theta)$ where $R(\theta), t(\theta)$ are the rotation

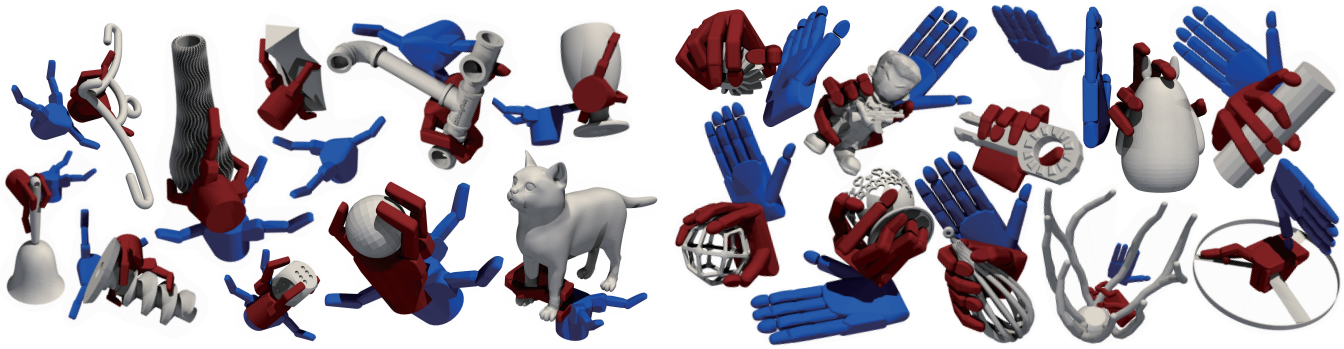

Fig. 2: We apply our method to grasp 10 complex objects using Barrett Hand (left) and Shadow Hand (right).

Barrett Hand	1	2	3	4	5	6	7	8	9	10
Time(s)	445.089	423.593	372.292	664.984	2301.98	1378.55	332.63	1295.92	778.848	165.251
Ours	1.9×10^{-6}	2.29×10^{-6}	1.26×10^{-5}	2.43×10^{-6}	7.05×10^{-6}	2.31×10^{-5}	2.17×10^{-5}	5.29×10^{-6}	2.14×10^{-6}	1.69×10^{-5}
Q_1-[6]	2.15×10^{-7}	1.34×10^{-14}	2.45×10^{-6}	1.78×10^{-6}	1.51×10^{-6}	1.41×10^{-5}	7.96×10^{-6}	1.88×10^{-6}	1.97×10^{-6}	3.22×10^{-6}
Closeness	1.9×10^{-6}	6.76×10^{-7}	1.05×10^{-5}	2.22×10^{-6}	7.61×10^{-6}	1.64×10^{-5}	2.11×10^{-5}	1.67×10^{-6}	2.96×10^{-6}	3.43×10^{-6}
Q_1-[23]	1.61×10^{-6}	2.18×10^{-6}	9.4×10^{-8}	3.13×10^{-6}	3.37×10^{-6}	1.48×10^{-5}	1.94×10^{-7}	1.11×10^{-8}	2.38×10^{-6}	5.12×10^{-9}
Shadow Hand	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Time(s)	149.603	326.604	477.113	1406.31	666.1	675.451	627.562	270052	519.32	748.311
Ours	5.32×10^{-6}	6.47×10^{-6}	1.17×10^{-5}	7.71×10^{-6}	1.48×10^{-5}	1.78×10^{-5}	1.16×10^{-5}	1.14×10^{-5}	1.01×10^{-5}	8.96×10^{-6}
Q_1-[6]	3.76×10^{-6}	1.13×10^{-6}	4.94×10^{-19}	2.2×10^{-6}	2.61×10^{-6}	4.34×10^{-6}	8.11×10^{-6}	9.66×10^{-6}	7.05×10^{-6}	2.8×10^{-6}
Closeness	4×10^{-6}	7.41×10^{-6}	2.94×10^{-6}	4.43×10^{-6}	6.34×10^{-6}	8.3×10^{-6}	5.18×10^{-6}	5.77×10^{-6}	3.54×10^{-6}	2.71×10^{-6}
Q_1-[23]	7.94×10^{-7}	3.99×10^{-6}	8.03×10^{-6}	4.91×10^{-6}	4.31×10^{-7}	3×10^{-6}	4.15×10^{-6}	7.46×10^{-6}	7.62×10^{-6}	2.68×10^{-7}

TABLE III: A comparison of grasp quality (Q_∞) using different algorithms on Barrett Hand (top row) and Shadow Hand (bottom row). From top to bottom: our method, differentiable grasp planner [6] guided by sub-gradients, our method with objective replaced by closeness measure, and EigenGrasp [23] using Q_1 objective function. On the first row, we profile the total computational time of our method. For fairness in comparison, we tune the number of iterations of other methods to use approximately the same computational time.

Algorithm 1 FGT

Input: Initial error threshold ϵ and $n_0(\epsilon)$

Cluster all x, y into boxes of side length $2\sqrt{\alpha}$
for $\mathcal{B}_y$ **do**
  $c \leftarrow$ center of $\mathcal{B}_y$
  Precompute A_n, B_n^j, C_n (Equation 23,24)
for $\mathcal{B}_{x,y}$ pair **do**
  $c \leftarrow$ center of $\mathcal{B}_y$
  $b \leftarrow$ center of $\mathcal{B}_x$
  Precompute E_m, F_m, H_m, I_m (Equation 25,26)
for $\mathcal{B}_x$ **do**
  Compute $G^d, \frac{\partial G^d}{\partial \theta}$ (Equation 27)

and translation of a rigid link, and y^l is the point y in local coordinates of the robot link. By the chain rule, we have:

$$\begin{aligned}\frac{\partial G^d}{\partial \theta} &= \frac{\partial G^d}{\partial \big(R,t\big)}\frac{\partial \big(R,t\big)}{\partial \theta}\\ \frac{\partial G^d}{\partial t_j} &= \frac{\partial G^d}{\partial y_j} \quad \frac{\partial G^d}{\partial R_{ij}} = \frac{\partial G^d}{\partial y_i} y_j^l.\end{aligned} \tag{27}$$

We first evaluate $\frac{\partial G^d}{\partial \big(R,t\big)}$ and then multiple by $\frac{\partial \big(R,t\big)}{\partial \theta}$. Each evaluation of $\frac{\partial G^d}{\partial t_j}$ can be performed using FGT by setting $\mathcal{S}(y) = (\pi r^2)^2$, and each evaluation of $\frac{\partial G^d}{\partial R_{ij}}$ can be performed by setting $\mathcal{S}(y) = (\pi r^2)^2 y_j^l$. Using the articulated body algorithm [37], the multiplication by $\frac{\partial \big(R,t\big)}{\partial \theta}$ incurs $\mathcal{O}(|\theta|)$. Altogether, the cost of evaluating $G^d, \frac{\partial G^d}{\partial \theta}$ is $\mathcal{O}(13(N+M)+|\theta|)$ and the cost of evaluating all the constraint gradients is $\mathcal{O}((13(N+M)+|\theta|)D)$. We further notice that M2M and M2L steps are irrelevant to the D wrench directions and need to be done only once, so the ultimate cost is: $\mathcal{O}(13(N+MD)+|\theta|D)$. We summarize FGT in Algorithm 1.

VI. Results

To validate the effectiveness of our approach, we employ a small dataset (Figure 2) containing 20 models from the Thingi10k object dataset [38], which is divided into two groups. The first group of 10 objects are to be grasped using the (6+4)-DOF three-fingered Barrett Hand [39] and the second group is to be grasped using the (6+22)-DOF Shadow Hand [40]. All experiments are carried out on a machine with 2.3 GHz 8-Core Intel Core i9 CPU and we use OpenMP to parallelize the double summations (Equation 20) and FMM computations (Algorithm 1) on CPU. For all the experiments, we choose $D = 128, \alpha = 10^{-3}, \gamma = 0.1, \beta = 0.5, c = 0.1, \tau = 10^{-10}$. We choose w_o^d to contain 64 evenly distributed unit vectors with non-zero forces and zero torques, plus another 64 evenly distributed unit vectors with zero forces and non-zero torques. We choose n_0 to ensure FMM approximation error is less than 10^{-6} according to [16].

Robustness: Our algorithm successfully processed the entire dataset, where the objects exhibit high geometrical and

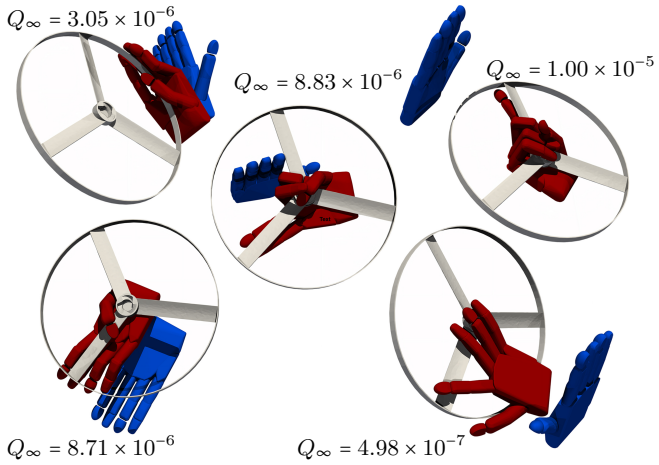

Fig. 3: For one of the objects in Figure 2, we plot the final grasp poses generated from 5 different, trivial initial guesses, where 4 out of 5 initial guesses lead to meaningful grasp poses.

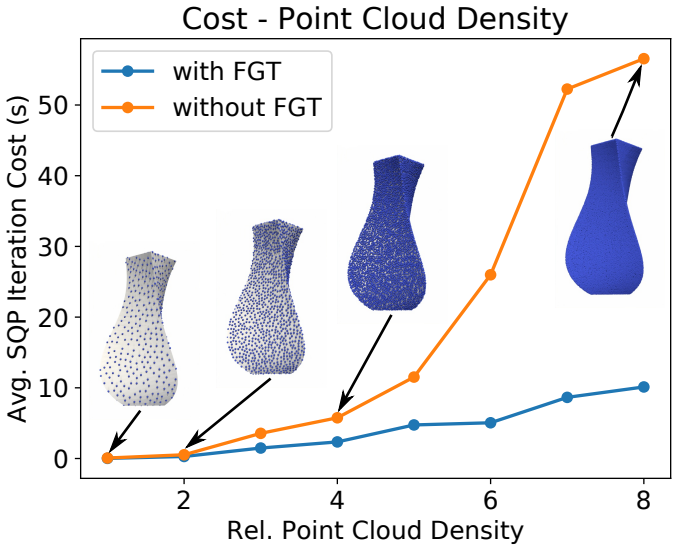

Fig. 4: The averaged computational cost of each SQP iteration plotted against the relative sample density. FMM acceleration achieves up to 5.6× speedup over brute-force summation at the highest sample density.

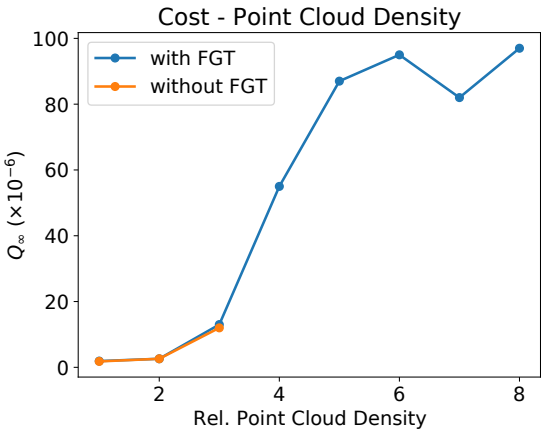

Fig. 5: The averaged grasp quality plotted against the relative sample density. FMM acceleration generates almost identical Q_∞ metric values as compared with brute-force summation.

topological complexities including both thin and tiny features that are oftentimes challenging in terms of collision-avoidance and contact point selection. However, our method can find human-like solutions (red poses in Figure 2) from trivial initializations (blue poses in Figure 2). We choose our initial gripper poses by randomly picking their approaching directions and setting all the joint angles to zero. In Figure 3, we show that our method is insensitive with respect to random initial guesses. We observe good grasp qualities from 4 out of 5 initial guesses. The grasp quality optimized using different algorithms are summarized in Table III (Larger numbers in Table III indicate better quality and all the numbers have small absolute values due to scaling of objects). As compared with GraspIt [23], our method achieves Min/Average/Max Q_∞ improvement rate of $0.78/402.30/3292.72$ on the BarrettHand and $1.33/9.07/34.37$ on the ShadowHand. This is the first time for model-based, optimization-based grasp planners to generate results at this level of complexity.

Comparisons: We have also compared our method with two prior gradient-based grasp planner. The first method is our prior work [6], where we use sub-gradients of the Q_1 metric to optimize grasp poses. The second method uses the closeness energy as objective function, which minimizes the distance between point on grippers and object surfaces. The closeness energy has also been used by [23]. Note that we compare all these methods in terms of the Q_∞ metric for fairness. According to Table III, our method significantly outperforms both these methods. We found that the method in [6] requires a near-optimal initial guess and they rely on groundtruth data to derive initial guesses. By starting from trivial initial guesses as in Figure 2, sub-gradients cannot find meaningful grasps. On the other hand, the closeness energy does not consider force equilibrium condition.

FMM Acceleration: In Figure 4 we plot the averaged iteration cost of SQP, with and without FMM acceleration. The accelerated SQP solver achieves up to 5.6× speedup as compared with brute-force summation at the highest density of sampled contact points. The use of FMM never deteriorate the quality of planned grasps, achieving almost identical results as compared with brute-force summation as illustrated in Figure 5. For reference, we plot the Q_∞ metric computed via brute-force summation for small densities. The computational time for larger densities is longer than 1 hour. We also observe improved optimized qualities when using a higher density, which will ultimately converge.

VII. Conclusion & Future Work

We present a full-featured, model-based, differentiable grasp planner that can plan both precision and power grasps. We first establish the connection between grasp planning and contact-implicit path planning, which takes the form of an IPCC. We further show that IPCC can be rewritten as an NLP via the kernel-integral relaxation. Finally, we propose a SQP-based practical algorithm to solve the NLP, where the kernel-integral is approximately and efficiently evaluated using FMM. Our method achieves a higher level of generality in terms of 3D object types and gripper types, and we provide guaranteed (self-)collision-free results. In the future, we plan to apply

our method to the training of robust, real-time grasp policies as in [6]. Our method can only find locally optimal grasps, and we plan to integrate our method with a stochastic global optimizer, such as Bayesian optimization [2], which can also handle uncertainties in object shapes.

REFERENCES

[1] M. Ciocarlie, C. Goldfeder, and P. Allen, "Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem," in *Robotics: Science and Systems Manipulation Workshop - Sensing and Adapting to the Real World*, 2007.

[2] J. Nogueira, R. Martinez-Cantin, A. Bernardino, and L. Jamone, "Unscented bayesian optimization for safe robot grasping," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2016, pp. 1967–1972.

[3] M. Laskey, J. Mahler, Z. McCarthy, F. T. Pokorny, S. Patil, J. Van Den Berg, D. Kragic, P. Abbeel, and K. Goldberg, "Multi-armed bandit models for 2d grasp planning with uncertainty," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, IEEE, 2015, pp. 572–579.

[4] J. Fontanals, B.-A. Dang-Vu, O. Porges, J. Rosell, and M. A. Roa, "Integrated grasp and motion planning using independent contact regions," in *2014 IEEE-RAS International Conference on Humanoid Robots*, IEEE, 2014, pp. 887–893.

[5] L. Wang, Y. Xiang, and D. Fox, "Manipulation Trajectory Optimization with Online Grasp Synthesis and Selection," in *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, 2020.

[6] M. Liu, Z. Pan, K. Xu, K. Ganguly, and D. Manocha, "Deep Differentiable Grasp Planner for High-DOF Grippers," in *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, 2020.

[7] H. Dai, A. Majumdar, and R. Tedrake, "Synthesis and optimization of force closure grasps via sequential semidefinite programming," in *Robotics Research: Volume 1*, A. Bicchi and W. Burgard, Eds. Cham: Springer International Publishing, 2018, pp. 285–305.

[8] M. Liu, Z. Pan, K. Xu, and D. Manocha, "New formulation of mixed-integer conic programming for globally optimal grasp planning," *IEEE Robotics and Automation Letters (2020)*, vol. abs/1909.05430, 2020.

[9] J. D. Schulman, K. Goldberg, and P. Abbeel, "Grasping and fixturing as submodular coverage problems," in *Robotics Research*, Springer, 2017, pp. 571–583.

[10] K. Hang, J. A. Stork, F. T. Pokorny, and D. Kragic, "Combinatorial optimization for hierarchical contact-level grasping," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 381–388.

[11] Z. Pan, X. Gao, and D. Manocha, "Grasping fragile objects using a stress-minimization metric," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 517–523.

[12] I. Mordatch, E. Todorov, and Z. Popović, "Discovery of complex behaviors through contact-invariant optimization," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, pp. 1–8, 2012.

[13] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 69–81, 2014.

[14] Z. Manchester, N. Doshi, R. J. Wood, and S. Kuindersma, "Contact-implicit trajectory optimization using variational integrators," *The International Journal of Robotics Research*, vol. 38, no. 12-13, pp. 1463–1476, 2019.

[15] O. Stein, "How to solve a semi-infinite optimization problem," *European Journal of Operational Research*, vol. 223, no. 2, pp. 312–320, 2012.

[16] M. Spivak, S. K. Veerapaneni, and L. Greengard, "The fast generalized gauss transform," *SIAM Journal on Scientific Computing*, vol. 32, no. 5, pp. 3092–3107, 2010.

[17] R Beatson and L. Greengard, "A short course on fast multipole methods," in *Wavelets, multilevel methods, and elliptic PDEs*, Oxford University Press, 1997, pp. 1–37.

[18] K. Hang, J. A. Stork, N. S. Pollard, and D. Kragic, "A framework for optimal grasp contact planning," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 704–711, 2017.

[19] Y. Zheng, "Computing the best grasp in a discrete point set," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2208–2214.

[20] N. Vahrenkamp, M. Do, T. Asfour, and R. Dillmann, "Integrated grasp and motion planning," in *2010 IEEE International Conference on Robotics and Automation*, IEEE, 2010, pp. 2883–2888.

[21] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Simultaneous grasp and motion planning: Humanoid robot armar-iii," *IEEE Robotics Automation Magazine*, vol. 19, no. 2, pp. 43–57, 2012.

[22] M. Ciocarlie, C. Goldfeder, and P. Allen, "Dimensionality reduction for hand-independent dexterous robotic grasping," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 3270–3275.

[23] *Graspit*, https://graspit-simulator.github.io/, Accessed: 2021-02-28, 2021.

[24] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *2016 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2016, pp. 1957–1964.

[25] H. Y. Li, M. Danielczuk, A. Balakrishna, V. Satish, and K. Goldberg, "Accelerating grasp exploration by leveraging learned priors," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, 2020, pp. 110–117.

[26] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 2, 2003, 1824–1829 vol.2.

[27] J. Xie and N. Chakraborty, "Rigid body dynamic simulation with line and surface contact," in *2016 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR)*, IEEE, 2016, pp. 9–15.

[28] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2012, pp. 4906–4913.

[29] Z. Pan, B. Ren, and D. Manocha, "Gpu-based contact-aware trajectory optimization using a smooth force model," in *Proceedings of the 18th annual ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2019, pp. 1–12.

[30] M. Toussaint, K. Allen, K. Smith, and J. Tenenbaum, "Differentiable physics and stable modes for tool-use and manipulation planning," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, 2018.

[31] M. Toussaint, J.-S. Ha, and D. Driess, "Describing physics for physical reasoning: Force-based sequential manipulation planning," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6209–6216, 2020.

[32] D. L. Logan, *A First Course in the Finite Element Method Using Algor*, 2nd. USA: Brooks/Cole Publishing Co., 2000.

[33] T. Hoheisel, C. Kanzow, and A. Schwartz, "Theoretical and numerical comparison of relaxation methods for mathematical programs with complementarity constraints," *Mathematical Programming*, vol. 137, no. 1-2, pp. 257–288, 2013.

[34] R. Ni, T. Schneider, D. Panozzo, Z. Pan, and X. Gao, "Robust & asymptotically locally optimal uav-trajectory generation based on spline subdivision," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[35] Z. Pan and D. Manocha, "Active animations of reduced deformable models with environment interactions," *ACM Trans. Graph.*, vol. 37, no. 3, Aug. 2018.

[36] Z. Pan, D. Zhang, C. Tu, and X. Gao, "Planning of power grasps using infinite program under complementary constraints," *arXiv:2108.00285*, 2021.

[37] R. Featherstone, *Rigid body dynamics algorithms*. Springer, 2014.

[38] Q. Zhou and A. Jacobson, "2018 cover image: Thingi10k," in *Computer Graphics Forum*, Wiley Online Library, vol. 37, 2018, pp. 451–452.

[39] *Barretthand*, https://advanced.barrett.com/barretthand, Accessed: 2021-02-28, 2021.

[40] *Shadow hand*, https://www.shadowrobot.com/dexterous-hand-series/, Accessed: 2021-02-28, 2021.