GraspQP: Differentiable Optimization of Force Closure for Diverse and Robust Dexterous Grasping

René Zurbrügg
ETH Zürich and ETH AI Center
zrene@ethz.ch

Andrei Cramariuc
ETH Zürich
crandrei@ethz.ch

Marco Hutter
ETH Zürich
mahutter@ethz.ch

| **Shadow Hand** | **Shadow Hand** | **Allegro Hand** | **Ability Hand** |
| Precision Grasp | Power Grasp | Pinch Grasp | Precision Grasp |

Figure 1: Visualization of predicted grasp poses for dexterous hands.

Abstract: Dexterous robotic hands enable versatile interactions due to the flexibility and adaptability of multi-fingered designs, allowing for a wide range of task-specific grasp configurations in diverse environments. However, to fully exploit the capabilities of dexterous hands, access to diverse and high-quality grasp data is essential- whether for developing grasp prediction models from point clouds, training manipulation policies, or supporting high-level task planning with broader action options. Existing approaches for dataset generation typically rely on sampling-based algorithms or simplified force-closure analysis, which tend to converge to power grasps and often exhibit limited diversity. In this work, we propose a method to synthesize large-scale, diverse, and physically feasible grasps that extend beyond simple power grasps to include refined manipulations, such as pinches and tri-finger precision grasps. We introduce a rigorous, differentiable energy formulation of force closure, implicitly defined through a Quadratic Program (QP). Additionally, we present an adjusted optimization method (MALA*) that improves performance by dynamically rejecting gradient steps based on the distribution of energy values across all samples. We extensively evaluate our approach and demonstrate significant improvements in both grasp diversity and the stability of final grasp predictions. Finally, we provide a new, large-scale grasp dataset for 5,700 objects from DexGraspNet, comprising five different grippers and three distinct grasp types.
Dataset and Code: https://graspqp.github.io/

1 Introduction

Dexterous grasping is a fundamental capability in robotics, enabling a wide range of applications, including industrial automation, service robotics, and human-robot collaboration. It is critical for allowing robots to interact with and manipulate objects in complex, unstructured environments, and remains a key focus of current research. Central to progress in dexterous grasping is the ability to predict stable grasps for various morphologies and under partial observation. A task that heavily relies on learning from large-scale datasets [1, 2, 3, 4, 5]. However, existing datasets are costly to generate and often limited in their diversity of gripper morphologies, grasp types, and physical realism.

Traditional grasp synthesis methods for collecting these datasets rely either on sampling-based algorithms [6, 7] or analytical techniques that evaluate force closure by solving (conic) optimization problems [8, 9, 10]. While effective at a basic level, these approaches suffer from poor sample efficiency, especially when accurate physical modeling is required, and tend to converge to simple power grasps rather than diverse, task-appropriate configurations. More recent gradient-based methods [11, 12, 1] improve efficiency by optimizing differentiable approximations of grasp quality metrics, but often oversimplify critical physical properties, thereby limiting grasp realism and diversity.

To overcome these limitations, we propose a new framework for synthesizing large-scale, diverse, and physically grounded dexterous grasps. First, we develop a differentiable force closure energy term that rigorously models grasp stability by considering frictional forces within an implicit Quadratic Program (QP) formulation. In contrast to prior approaches [1, 13, 3, 11] that rely on simplified or relaxed metrics, our formulation preserves the core physical properties of grasp stability while remaining fully differentiable. This enables gradient-based optimization without compromising physical accuracy and allows the synthesis of grasps that include refined manipulation strategies such as pinch and tri-finger precision grasps. Second, we propose MALA*, an enhanced optimization and sampling strategy inspired by the Metropolis-Adjusted Langevin Algorithm (MALA). MALA* dynamically rejects gradient steps that would lead to poor sample diversity, using information about the global distribution of existing samples. This adjustment mitigates mode collapse during optimization and encourages exploration of a broader range of feasible and physically distinct grasps. We extensively evaluate our approach and demonstrate significant improvements over existing techniques in terms of grasp diversity and stability. Additionally, we release modular grasping environments for evaluation and reinforcement learning, built on IsaacLab [14].

Together, these advancements enable the efficient synthesis of large-scale, high-diversity grasp datasets. To validate our approach, we perform detailed evaluations and generate a new dataset comprising grasps for 5,700 objects from DexGraspNet [1]. The dataset includes annotations for five different grippers and spans three distinct grasp types, substantially expanding the diversity of available data for dexterous manipulation.

In summary, our contributions are:

  (i) We improve upon existing analytical grasp generation methods using a more analytically rigorous force closure formulation.

 (ii) We introduce an adjusted optimization method, MALA*, that improves convergence and diversity.

(iii) We generate large-scale grasp predictions across various objects, grippers, and grasp taxonomies.

2 Related Work

2.1 Analytical Grasp Synthesis

Grasp synthesis aims to generate a diverse set of stable grasp predictions for known hand and object pairs. Traditional approaches to grasp synthesis primarily rely on sampling-based methods or sequences of linear programs [6, 2, 15, 16], which explore the space of possible hand poses to optimize

a given grasp quality metric. However, these methods can be computationally expensive, particularly for high-degree-of-freedom (DoF) dexterous hands. To improve efficiency, recent methods have shifted toward gradient-based optimization [11, 13, 12, 3, 17]. These approaches leverage differentiable grasp quality metrics, enabling faster convergence compared to sampling-based methods. However, this reliance on differentiability introduces limitations. Classical grasp quality metrics such as Q_1/ε [18, 19, 20], which involve solving min-max optimization problems, are not inherently differentiable. To address this, recent works have proposed relaxed, differentiable formulations [21, 11], bilevel optimization [22, 16, 23] and sampling-based wrench space analysis [24], which allow for gradient-based optimization. Despite their computational advantages, these relaxed formulations often introduce implicit assumptions that can oversimplify grasp stability. Notably, some methods fully neglect frictional forces [11, 1], which are critical for certain grasp configurations. BilevelOpt [16] incorporates the force closure condition into the constraint formulation of an optimization problem and relies on suboptimal initial solutions from IK solvers and contact points. FRoGGeR [22] employs a differentiable force closure metric similar to ours but constrains the interaction wrench coefficients to sum to one to prevent vanishing gradients. Similarly, concurrent work such as BoDEX [23] enforces a bound on contact wrenches by requiring the sum of normal forces to exceed a predefined hyperparameter γ. Alternative approaches mitigate this by incorporating differentiable physics simulations, using velocity-based metrics to evaluate grasp quality [12, 3], but they depend on simplified contact assumptions or relaxations, such as penalty-based contact modeling.

2.2 Dexterous Grasping Datasets

A wide range of datasets [2, 3, 1, 13, 25, 4, 26, 20, 27, 5] provide synthesized grasp poses for rigid objects across different grippers. These datasets vary in scale, from sets with a few hundred thousand grasps [21, 13] to large-scale datasets containing a million or more grasp annotations [4, 13, 1, 2]. In terms of gripper diversity, many focus on parallel jaw grippers [4, 28], while few incorporate dexterous hands such as the Shadow and Allegro Hand [2, 3]. Our dataset extends the diversity of grippers by providing annotations for the Psyonic Ability Hand, a commonly used research hand. We additionally focus on increasing the diversity of generated grasps per object, both in joint entropy and in taxonomy.

3 Preliminaries

Positive Spanning Sets: As formally defined in literature [29], positive spanning sets are characterized as follows:

Definition 3.1 (Positive span). *The positive span of a finite set of vectors $\mathcal{S} = \{v_1, ..., v_k\}$, denoted $pos(\mathcal{S})$, is given by:* $pos(\mathcal{S}) := \{\lambda_1 v_1 + ... + \lambda_k v_k \ : \ \lambda_i \geq 0 \ \forall i = 1, ..., k\}$.

Additionally, [29] introduce the following theorem about the range of positive spanning sets:

Theorem 3.1 (Range of positive spanning sets). *Suppose $S = \{v_1, ..., v_k\} \subseteq \mathbb{R}^n$, with $v_i \neq 0$, linearly spans the subspace V of $\mathbb{R}^n$. The following statements are equivalent:*
  (i) The set S positively spans V
  (ii) $\exists \ \alpha_1, ..., \alpha_k > 0$ such that $\sum_{i=1}^{k} \alpha_i v_i = 0$
  (iii) $\exists \ \gamma_1, ..., \gamma_k \geq 0$ such that $\sum_{i=1}^{k} \gamma_i v_i = - \sum_{i=1}^{k} v_i$

Note that Theorem 3.1 - (ii) is well known and often referred to as "*the origin must lie in the convex hull of the spanning set*" in the literature.

Form and Force Closure: Given a set of contact points $\{x_i\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^3$, a grasp is defined to be in *form closure* if the static contact points, together with their associated frictionless interaction forces $\{f_i\}_{i=1}^{N}$, $f_i \in \mathbb{R}^3$, completely constrain all possible motions of the object. More formally, this condition is satisfied if the set of contact wrenches $\mathcal{W} = \{w_i\}_{i=1}^{N}$ with $w_i = [f_i; (f_i \times x_i)] \in \mathbb{R}^{6 \times 1}$, constructed from the contact points $\{x_i\}_{i=1}^{N}$ and the corresponding interaction forces $\{f_i\}_{i=1}^{N}$, positively spans the entire wrench space $\mathbb{R}^6$: $pos(\mathcal{W}) = \mathbb{R}^6$.
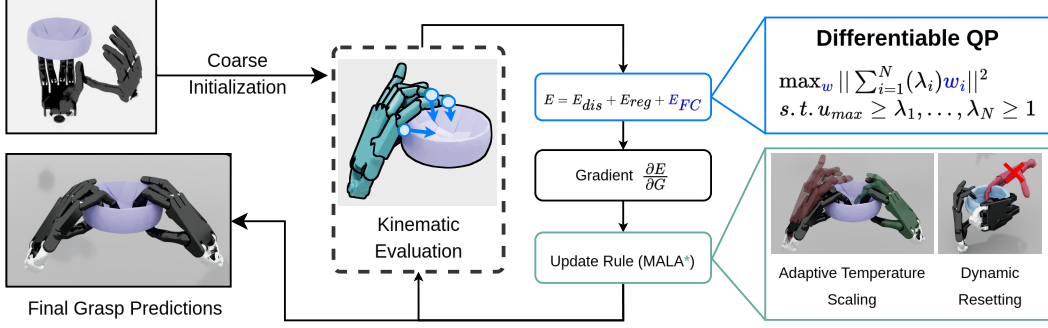
Figure 2: **Overview of our Grasp Generation Method:** Starting from a coarse initialization, grasp candidates are optimized by minimizing a composite energy function composed of distance, regularization, and force closure terms. The force closure energy is computed via a differentiable quadratic program (QP) that ensures the contact wrenches positively span $\mathbb{R}^6$. Gradients of the energy are used to iteratively refine grasp poses through our modified Metropolis-adjusted Langevin (MALA*) algorithm, which dynamically adapts updates and resets based on the energy landscape. Grasp proposals that significantly underperform relative to the current energy distribution are reset (*Dynamic Resetting*), and the acceptance probability of each gradient step is modulated according to the proposal's relative energy performance (*Adaptive Temperature Scaling*). After convergence, the method produces diverse and physically stable grasp predictions.

Furthermore, *force closure* requires that the grasp can resist any arbitrary external wrench applied to the object by additionally leveraging frictional properties at contact points, i.e. the set of wrenches generated by the contact forces $\mathcal{W}_{\mathcal{FC}} = \{\hat{w}|\hat{w} \in \bigcup_{i=1}^N WC(w_i)\}$ positively span the entire wrench space $\mathbb{R}^6$: $\mathrm{pos}(\mathcal{W}_{\mathcal{FC}}) = \mathbb{R}^6$ (1).

Finally, we define the *Wrench Matrix* $W_{FC} = [\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N] \in \mathbb{R}^{6 \times N}$ as the matrix whose j-th column is the contact wrench $\hat{w}_j \in \mathcal{W}_{\mathcal{FC}}$, where N is the total number of interaction wrenches. Force closure is satisfied iff the positive span of the columns of W_{FC} spans $\mathbb{R}^6$.

Grasp Representation: For a given robotic gripper with n_q joints, we parameterize each grasp as $G = (\chi, q, \Delta q)$, where $\chi \in SE(3)$ denotes the wrist pose, $q \in \mathbb{R}^{n_q}$ the joint positions of the n_q joints of the robot and $\tau_q \in \mathbb{R}^{n_q}$ the desired joint torques. Furthermore, each grasp is associated with a set of N_C contact points $\mathcal{C} = \{c_i\}_{i=1}^{N_c}$, given in the object frame, and their corresponding object normals $\mathcal{N} = \{n_i\}_{i=1}^{N_c}$. Given these contact points, we calculate the set of interaction wrenches as $\mathcal{W} = \{w_i\}_{i=1}^{n_c}$, where $w_i = [n_i, c_i \times n_i] \in \mathbb{R}^6$. Finally, when considering friction, we rely on the four-sided pyramid under-approximation of the friction cone $FC(f_i, c_i, n_i)$ [30].

4 Method

Given an accurate 3D model of an object and a dexterous hand, we aim to synthesize a diverse set of feasible grasp configurations $\mathcal{G} = \{G_i\}_{i=1}^N$ that satisfy the force closure property in Eq. (1). To accomplish this, we first generate a diverse initial set of feasible grasps and then refine them through an analytical optimization process that maximizes the span of the associated friction cones. An overview of our method is depicted in Fig. 2, and final grasp predictions are shown in Fig. 1.

4.1 Analytical Grasp Generation

We build upon a grasp generation pipeline commonly employed in gradient-based grasp synthesis [3, 12, 11, 1], following an approach similar to DexGraspNet [1], which we briefly review below. Each gripper is associated with a predefined set of potential contact points $\mathcal{C}$, of which only a subset $\mathcal{C}'$ is active at a time (i.e., considered to be in contact with the object). Given a grasp-object configuration, the energy of the grasp is computed as

$$E = E_{FC} + w_{dis}E_{dis} + w_{reg}E_{reg}, \tag{2}$$

with E_{FC} being a force closure metric, E_{dis} denotes the distance of the active contact points ($\mathcal{C}'$) to the object surface, and E_{reg} consists of regularization energies ($E_{pen}, E_{joints}, E_{spen}$) which are used to prevent the grasp from penetrating the object, itself, or exceeding the joint limits, respectively.

Force Closure Metric: DexGraspNet [1] assumes the grasp to be frictionless and that the force applied at each contact point has equal magnitude, which does not accurately model real-world hardware, where fingers can be actuated separately. Formally, under these assumptions, their force closure metric becomes a form closure assumption and a simplification of Theorem 3.1 - (ii) where $\alpha_i = 1$. This results in $E_{FC} = ||\sum_{i \leq |\mathcal{C}'|} w_i||_2 \to 0$, i.e., the sum of the column vectors of the wrench space must sum up to zero.

A more general formulation of a force closure metric could be achieved by directly using Theorem 3.1 - (ii) to find the optimal coefficients α_i for the current contact points $\mathcal{C}'$, i.e.,

$$E_{FC} = || \sum_{i \leq |\mathcal{C}'|} \alpha_i w_i ||_2 \ \text{ s.t. } \alpha_i > 0 \ \forall i = 1, ..., n_c. \tag{3}$$

While this approach would be more general, it is challenging to include as an optimization term, since the error term E_{FC} can become arbitrarily small for arbitrarily small choices of α_i, leading to vanishing gradients. Hence, we propose relying on the more robust formulation in Theorem 3.1 - (iii) to ensure that the grasp is in force closure:

$$E_{FC} = || \sum_{i \leq |\mathcal{C}'|} \gamma_i w_i + \sum_{i \leq |\mathcal{C}'|} w_i ||_2 = || \sum_{i \leq |\mathcal{C}'|} \hat{\gamma}_i w_i ||_2 \ \text{ s.t. } u \geq \hat{\gamma}_i \geq 1 \ \forall i = 1, ..., n_c, \tag{4}$$

where u is an upper bound on the interaction force for each contact point. While this is not strictly necessary for force closure, it ensures that the required interaction forces remain bounded, which is necessary when using a real robot with torque limits. The resulting coefficients $\hat{\gamma}_i$ directly reflect the force applied at each contact point to keep the object stationary, as opposed to the more degenerate formulation in Eq. (3), which incentivizes the fingers to apply a minimal amount of force. Note that this formulation is equivalent to the force closure metric from Liu et al. [11] if $\hat{\gamma}_i = 1$ is chosen for all i.

Furthermore, we note that the force closure metric is implicitly defined through an optimization problem and has imposed hard constraints (introduced by the bounds on $\hat{\gamma}$). This makes propagating the gradient of the entire problem more challenging. We propose and evaluate two different approaches to make the optimization problem differentiable:

1. *Unconstrained Optimization*: By introducing barrier functions, the energy term can be converted into an unconstrained optimization problem. Hence, the energy term E_{FC} is replaced with $E_{FC} + \texttt{barrier}(\hat{\gamma})$ and the resulting unconstrained optimization problem is solved using a trust region-based optimizer (Powell's Hybrid Method) [31].

2. *Differential QP*: We formulate the optimization problem as a Quadratic Program (QP):

$$E_{FC} = \min_z \frac{1}{2} z^T H z + g^T z \ \text{ s.t. } Az \geq b,$$
$$H = W_{FC}^T W_{FC}, \ g = 0, \ b = [1_{n_c}; u_{n_c}],$$
$$z = [\hat{\gamma}_1, ..., \hat{\gamma}_{N_c}], \ A = \texttt{diag}(1_{n_c \times n_c}, -1_{n_c \times n_c}).$$

   The gradients with respect to the input parameters z can be calculated by differentiating the Karush–Kuhn–Tucker (KKT) conditions of the QP, as shown by Amos *et al.* [32].

Finally, Theorem 3.1 guarantees force closure only if the wrench space linearly spans $\mathbb{R}^6$. The original metric by Liu *et al.* [11] ensures this by additionally optimizing over the smallest singular value of the wrench matrix W_{FC}. However, subsequent works [1, 13] do not include this in their energy formulation and hence lose force-closure guarantees. To ensure the wrench matrix forms a linear basis, we propose to scale the energy term by $e^{-\prod_i \sigma_i(W_{FC})}$, where σ_i are the singular values of the wrench matrix. All $\sigma_i > 0$ ensures full rank, satisfying Theorem 3.1, while maximizing their product maximizes the wrench space volume.

Therefore, we define the final energy term as:

$$E_{FC} = || \sum_{i \leq |\mathcal{C}'|} \hat{\gamma}_i w_i ||_2 \cdot e^{-\prod_i \sigma_i(W_{FC})} \ \text{ s.t. } \hat{\gamma}_i \geq 1 \ \forall i = 1, ..., n_c. \tag{5}$$
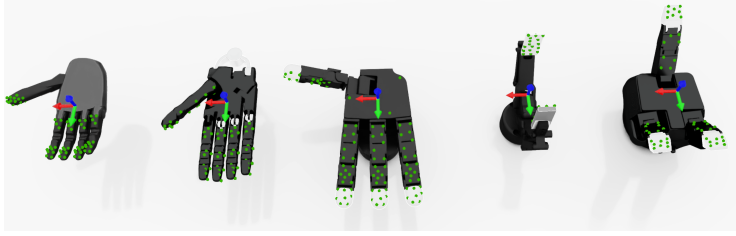
5

Figure 3: **Overview of the Five Robotic Grippers used for Evaluation:** From left to right, Psyonic Ability Hand, Shadow Hand, Allegro Hand, Robotiq2f140 and Robotiq3F. We show the standardized wrist frame convention with axes indicating thumb (x-axis), fingers (y-axis), and palm normal (z-axis) directions. Randomly sampled contact points from the manually defined contact meshes are shown in green.

Distribution-based MALA Optimizer: We often observe that certain grasp proposals get stuck in local minima (i.e., bad grasp configurations) during optimization. Escaping these minima proves challenging since the optimization is rolled out for each grasp independently without any knowledge of the current loss landscape. However, such knowledge (and therefore the detection of local minima) can be made available by incorporating additional information from the full grasp distribution used for optimization at the current time step t. We therefore introduce the following changes to the optimization subroutine[1], which is applied to the grasp distribution at each iteration k: $\mathcal{G}^{(k)} = \{G_0^{(k)}, ..., G_N^{(k)}\}$ with energies $E^{(t)} = \{E_0^{(k)}, ..., E_N^{(k)}\}$:

Dynamic Resetting: To prevent grasp starvation (i.e., grasp poses converging to local minima), we reset the optimizer for grasps that are performing significantly worse compared to the overall grasp distribution. After every n_{reset} steps, we fit a normal distribution, parameterized by $\mathcal{N}_E(\mu, \sigma)$, over the current set of energies $E^{(k)}$. We then re-initialize all grasp poses with $\Phi_E(E_i; \mu, \sigma) \leq p_{th}$, i.e., the grasps that are within the lowest quantile of the fitted normal distribution, which is calculated based on the cumulative distribution function Φ_E and the upper probability threshold p_{th} are re-initialized.

Adaptive Temperature Scaling: We update the Metropolis–Hastings acceptance criterion for accepting gradient steps. In the original formulation, a gradient step that introduces an energy change ΔE is accepted with probability $p \sim e^{-\Delta E/T}$. We propose conditioning the temperature T on the energy distribution, i.e., if a grasp performs significantly worse than the overall distribution, we increase the temperature for that grasp. Hence, we change the acceptance criterion to $p \sim e^{-\Delta E/T_i}$, where T_i is the temperature for grasp i, and calculated using the following formula: $T_i = T \cdot (1 + \Phi_E(E_i))$. We refer the reader to Appendix 1 for the complete formulation and pseudocode of the optimizer.

5 Results

Main Evaluation Metrics: To assess the performance of our method, we introduce the following evaluation metrics:

Unique Grasp Rate (UGR) captures the number of distinct, stable grasps that the method can generate. This metric is calculated by discretizing the successful grasp poses using three different resolutions $\delta_r = 2\,\text{cm}$, $\delta_\phi = 4°$, and $\delta_q = 1.15°$ for the position, orientation (Euler angles), and joint states, respectively. We then calculate the number of unique grasps and report the UGR as the ratio of unique successful grasps over the total number of grasps for each object.

Entropy (H), a common measure for the diversity of grasp configurations is the mean entropy of the finger joints. We extend this metric to also include the position and orientation of the grasp configuration since, depending on the task at hand, different grasp positions and orientations might be more suitable. Note that calculating the entropy of an SE(3) rotation is non-trivial. We refer the reader to Appendix 8.1 for the full derivation.

Evaluation: We assess the stability of each grasp proposal using Isaac Lab [14] and an evaluation method as introduced in [1]. Specifically, while grasping, forces are applied to the objects in the six canonical world directions. We assess stability by checking that the center of mass (CoM) of the objects stays within a sphere of radius $3\,\text{cm}$ for at least one of the principal axis pairs ($\pm x$, $\pm y$, $\pm z$).

---

[1]The full algorithm is described in the Appendix - Alg.1

Table 1: **SYNTHESIZED GRASP POSES**: We compare the influence of different analytical optimization schemes on the final grasp quality and diversity metrics, for 4 and 12 contact points respectively, using our test set of 50 objects. We report the unique grasp rate (*UGR*) as the percentage of successful unique grasps over all generated unique grasps, the joint entropy (H) of the successful grasp distribution, and the average max penetration depth (D). All evaluations are performed in a no-gravity environment with 5 N force.

Grasp Type	Method	Optimizer	Allegro UGR↑	H↑	Shadow Hand UGR↑	H↑	Robotiq3f UGR↑	H↑	Robotiq2f UGR↑	H↑	Ability Hand UGR↑	H↑	Overall UGR↑	H↑	D↓[mm]
4 Ctc.	DexGraspNet [1]	MALA	67%	3.5	38%	2.9	26%	2.3	28%	3	15%	1.6	35%	2.7	1.0
	DexGraspNet [1]	MALA*	71%	3.5	36%	2.9	34%	2.5	32%	3.1	14%	1.5	38%	2.7	**0.9**
	GraspQP(ours)	MALA*	**74%**	**3.6**	**49%**	**3.2**	**49%**	**2.9**	**34%**	**3.3**	**19%**	**1.8**	**45%**	**3.0**	**0.9**
12 Contacts	DexGraspNet [1]	MALA	37%	2.7	14%	1.6	5%	0.6	–	–	14%	1.5	18%	1.6	**1.9**
	GenDexGrasp [13]	MALA*	48%	3.1	23%	2.2	9%	1.1	–	–	18%	1.8	25%	2.0	2.6
	TDG [24]	MALA*	49%	3.3	23%	2.3	8%	0.9	–	–	16%	1.6	24%	2.0	2.7
	GraspQP(ours)	MALA*	**55%**	**3.3**	**29%**	**2.5**	**9%**	**1.1**	–	–	**26%**	**2.3**	**30%**	**2.3**	2.3
	DexGraspNet [1]	MALA	31%	2.5	10%	1.3	4%	0.5	–	–	11%	1.2	14%	1.4	**1.8**
	GenDexGrasp [13]	MALA*	44%	2.9	14%	1.6	4%	0.4	–	–	17%	1.8	20%	1.7	2.6
	TDG [24]	MALA*	42%	3.0	13%	1.4	5%	0.6	–	–	16%	1.7	19%	1.7	2.7
	GraspQP(ours)	MALA*	**49%**	**3.1**	**17%**	**1.8**	**4%**	**0.5**	–	–	**23%**	**2.1**	**23%**	**1.9**	2.3
	MultiGripper [2]	GraspIt! [6]	62%	3.3	17%	1.6	35%	2.6	-	-	-	-	-	-	–
	DexGraspNet [1]	MALA	66%	3.5	35%	2.9	33%	2.6	30%	3.2	18%	1.8	37%	2.8	1.6
	DexGraspNet [1]	MALA*	68%	3.5	41%	3.0	40%	2.8	31%	3.2	18%	1.7	40%	2.9	1.6
	GenDexGrasp [13]	MALA*	74%	3.6	47%	3.2	51%	2.9	32%	3.4	23%	2.2	45%	3.0	1.7
	TDG [24]	MALA*	69%	3.5	47%	3.2	54%	2.9	28%	3.3	25%	2.2	45%	3.0	1.7
	GraspQP(ours)	MALA*	**76%**	**3.6**	**63%**	**3.5**	**57%**	**3.0**	29%	3.2	**35%**	**2.6**	**52%**	**3.2**	1.4

Dataset: We convert the dataset introduced by Wang et al. [1], comprising 5,700 objects, into the Universal Scene Description (USD) format to enable direct use within Isaac Sim. To improve collision handling, we recompute the collision meshes using Quad Remesher [33], and set the refined meshes as the new collision geometry for GPU-accelerated signed distance field (SDF) collision checking. For evaluation purposes, we randomly select a subset of 50 objects from the dataset to form our test set. All reported results are based on this test set.

Main Results: We report our main results in Table 1, comparing grasp quality across different contact points, gripper types, and baseline methods when exposed to a 5 N disturbance force. We report the Unique Successful Grasp Rate (UGR), entropy (H), and average maximum penetration depth evaluated per gripper. We compare our method against four baselines, namely DexGraspNet [1], GenDexGrasp [13], TDG [24], and MultiGripperDataset [2]. Implementation details for all methods are provided in Appendix 8.3. Across all settings, our method (GraspQP) consistently outperforms prior work both in grasp entropy (H) and unique grasp rate (UGR). We evaluate our approach under two settings, restricting the number of active contact points $|\mathcal{C}'|$ to either 4 or 12, and observe that our method scales better with a larger number of contact points, albeit at the cost of slightly larger penetration depths. This trend is expected, as increasing the number of contact points not only promotes greater contact area but also facilitates easier satisfaction of the force closure condition. We further observe that the addition of our modified optimizer, MALA*, improves the overall UGR as well as the entropy. This effect is more pronounced in the 12 contact setting and discussed in Tab. 2.

The Robotiq2f gripper denotes the only case where our method does not significantly outperform existing methods. We largely attribute this to the fact that the gripper only consists of two finger pads that are always parallel, and the influence of the grasp metric becomes less important. Additionally, the UGR for this gripper tends to collapse, as many grasps converge to similar gripper widths when interacting with object parts that have uniform geometries. However, even though the unique grasp rate is limited, the raw grasp success rate (reported in Appendix 8.4) remains high (above 80%), demonstrating that the generated grasps are still physically plausible and stable despite reduced diversity. Furthermore, when evaluating different grasp taxonomies, we observe a substantial decline in the number of unique successful grasps, particularly for the Robotiq 3F gripper. In this case, fewer than 10% of the optimized grasps remain both successful and unique, corresponding to approximately three valid grasps per object. In general, we primarily attribute this degradation in performance to the reduced stability of pinch and precision grasps, which inherently involve a limited number of contact points. However, as shown in Appendix 8.4, when the external disturbance force is limited to 1 N, the unique grasp rate improves significantly, reaching 41% for pinch and 49% for precision grasps. For completeness, we report all metrics across all disturbance forces in Appendix 8.4 and additionally list the raw grasp success rates for all cases.

Table 2: **ABLATION STUDY:** Impact of optimizer variants and design choices on grasp quality. We report changes in entropy (H↑), unique grasp rate (UGR↑), and hand-object penetration (D↓). Improvements from replacing MALA with our MALA* optimizer are shown across different methods, while subsequent rows analyze the effect of changing parts of the optimizer, optimization problem and relaxing the form closure condition in Theorem (3.1) from (iii) to (ii) or using the unconstrained softmax formulation $\alpha_i = \frac{\exp \beta_i}{\sum_j \exp(\beta_j)}$. All values are calculated and averaged over all power grasps for all dexterous hands.

Method	Optimizer	Energy Formulation	Δ UGR ↑ [%]	ΔH ↑	Δ D ↓ [mm]
GenDexGrasp	MALA → MALA*	Liu et al. [11]	+7.0	+0.3	-0.3
TDG		Chen et al. [24]	+5.0	+0.2	-0.3
		ours	+5.4	+0.2	-0.4
GraspQP	MALA* → MALA* w/o resets	ours	-4.8	-0.1	+0.1
	MALA* → MALA* w/o temp scaling		-4.5	-0.1	+0.0
	MALA*	ours → ours w/o exp	-0.3	+0.0	-0.1
		Constrained (iii) → Unconstrained	-10.0	-0.2	+0.1
		Constrained (iii) → Unconstrained-Softmax	-10.1	-0.3	-0.3
		Constrained (iii) → Constrained (ii)	-4.8	-0.1	-0.2

Our approach is slower than the baseline (3.4 s vs. 1.15 s per grasp on the 24-DoF Shadow Hand) but is intended for offline dataset generation, where speed is less critical. The extra computation results in significantly more diverse grasps. As shown in Figure 4, it may produce fewer grasps under tight time limits but ultimately reaches far more unique grasps (about 80 with 128 seeds) compared to the baseline's saturation at 60 even with 512 seeded grasps.

Ablation Studies: We conduct ablation studies to assess the impact of our proposed MALA* optimizer and alternative optimization strategies on the UGR and entropy. An overview is shown in Table 2. For each design choice, we report both the absolute and relative changes in entropy (ΔH), unique grasp rate (ΔUGR), and average max penetration depth (ΔD), which are evaluated for the power grasp configuration and averaged across all hand types. The results demonstrate that using MALA* consistently improves entropy (H) and unique grasp rate (UGR), and reduces penetration depth (D). In order to motivate
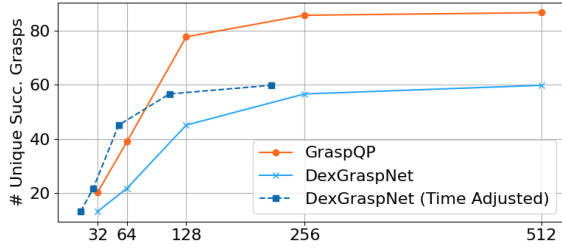

Figure 4: **Unique Successful Grasps (UGR) vs. Seed Size**. Average UGR per object w.r.t. the number of initialized grasps (seeds). DexGraspNet yields more grasps under tight time constraints (Time Adjusted), but GraspQP scales more effectively, reaching 80 unique grasps with only 128 seeds. DexGraspNet saturates at 60 even with 512 seeds, indicating a diversity bottleneck that more samples cannot overcome.

our design choices, we further investigate the influence of our optimization formulation and solver. We show that reformulating the optimization problem as a nonlinear constraint problem (Sec. 1 using Theseus [34]) leads to a significant degradation in performance, highlighting the necessity for hard constraints within the optimization formulation. A similar (but less strong) decline is observed when reformulating the form closure condition from (iii) to (ii) (Theorem 3.1 and additionally including the constraint $\sum_i^N \alpha_i = N$, to prevent the convergence to $\alpha_i \to 0$).

6   Conclusion

In this work, we proposed a method to synthesize diverse and stable grasps for a wide range of objects and robotic grippers. Our approach combines a differentiable force closure energy, defined through a Quadratic Program (QP), with an adjusted optimization strategy (MALA*) that promotes diversity during grasp synthesis. We showed that our method is capable of generating a broad set of stable and unique grasps across different object geometries and gripper types, outperforming existing approaches in terms of both grasp stability and entropy. Finally, we provide a detailed ablation study and analysis, validating the effectiveness of each component of our method. As future work, we plan to extend our framework to dynamic manipulation scenarios and aim to use the generated dataset to train deep learning models for real-world grasp synthesis and robotic deployment.

7 Limitations

While our method can generate a diverse set of stable grasps for a wide range of objects and grippers, there are still some limitations to our approach. For the analytical optimization, our results demonstrate that incorporating a rigorous force closure metric directly enhances grasp quality and diversity. However, this comes at the cost of increased computational complexity, with our optimization-based formulation requiring approximately 1.5–3 times more computation time than baseline methods (worst case 3.4 s / grasp (ours) vs 1.15 s / grasp for the 24-DoF Shadow Hand). While this is acceptable for offline grasp synthesis, it may pose limitations for real-time applications or when used as a reward function in reinforcement learning. A potential solution could involve implementing ADMM solvers [35] on GPU.

Another challenge is the occurrence of mode collapse, where multiple grasps converge to very similar configurations, reducing the number of unique grasps. Introducing coupling between grasp proposals to prevent them from occupying the same spatial region could help alleviate this issue. Although preliminary experiments using density-based repulsion forces have not yielded promising results, we intend to explore this direction further in future work. Finally, we observed some limitations within the Isaac Sim physics engine, where fingertips sometimes penetrate the object surface, leading to potential false positives in the grasp evaluation.

Acknowledgments

References

[1] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11359–11366. IEEE, 2023.

[2] L. F. Casas, N. Khargonkar, B. Prabhakaran, and Y. Xiang. Multigrippergrasp: A dataset for robotic grasping from parallel jaw grippers to dexterous hands, 2024. URL https://arxiv.org/abs/2403.09841.

[3] D. Turpin, T. Zhong, S. Zhang, G. Zhu, E. Heiden, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg. Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation. In *ICRA*, 2023.

[4] C. Eppner, A. Mousavian, and D. Fox. Acronym: A large-scale grasp dataset based on simulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6222–6227, 2021. doi:10.1109/ICRA48506.2021.9560844.

[5] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen. The columbia grasp database. In *2009 IEEE International Conference on Robotics and Automation*, pages 1710–1716, 2009. doi:10.1109/ROBOT.2009.5152709.

[6] A. Miller and P. Allen. Graspit! a versatile simulator for robotic grasping. *Robotics Automation Magazine, IEEE*, 11(4):110 – 122, dec. 2004. ISSN 1070-9932. doi:10.1109/MRA.2004.1371616.

[7] J. Huh, B. Lee, and D. D. Lee. Constrained sampling-based planning for grasping and manipulation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 223–230, 2018. doi:10.1109/ICRA.2018.8461265.

[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*, 5(3):4663–4670, 2020.

[9] L. Han, J. Trinkle, and Z. Li. Grasp analysis as linear matrix inequality problems. *IEEE Transactions on Robotics and Automation*, 16(6):663–674, 2000. doi:10.1109/70.897778.

[10] Y.-H. Liu. Qualitative test and force optimization of 3-d frictional form-closure grasps using linear programming. *IEEE Transactions on Robotics and Automation*, 15(1):163–173, 1999. doi:10.1109/70.744611.

[11] T. Liu, Z. Liu, Z. Jiao, Y. Zhu, and S.-C. Zhu. Synthesizing diverse and physically stable grasps with arbitrary hand structures using differentiable force closure estimator. *IEEE Robotics and Automation Letters*, 7(1):470–477, 2021.

[12] D. Turpin, L. Wang, E. Heiden, Y.-C. Chen, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg. Grasp'd: Differentiable contact-rich grasp synthesis for multi-fingered hands. In *European Conference on Computer Vision*, pages 201–221. Springer, 2022.

[13] P. Li, T. Liu, Y. Li, Y. Zhu, Y. Yang, and S. Huang. Gendexgrasp: Generalizable dexterous grasping. *arXiv preprint arXiv:2210.00722*, 2022.

[14] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, R. Singh, Y. Guo, H. Mazhar, A. Mandlekar, B. Babich, G. State, M. Hutter, and A. Garg. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6):3740–3747, 2023. doi:10.1109/LRA.2023.3270034.

[15] H. Dai, A. Majumdar, and R. Tedrake. *Synthesis and Optimization of Force Closure Grasps via Sequential Semidefinite Programming*, pages 285–305. Springer International Publishing, Cham, 2018. ISBN 978-3-319-51532-8. doi:10.1007/978-3-319-51532-8_18. URL https://doi.org/10.1007/978-3-319-51532-8_18.

[16] A. Wu, M. Guo, and C. K. Liu. Learning diverse and physically feasible dexterous grasps with generative model and bilevel optimization. *arXiv preprint arXiv:2207.00195*, 2022.

[17] S. Chen, J. Bohg, and C. K. Liu. Springgrasp: Synthesizing compliant, dexterous grasps under shape uncertainty. *arXiv preprint arXiv:2404.13532*, 2024.

[18] C. Ferrari and J. F. Canny. Planning optimal grasps. *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2290–2295 vol.3, 1992. URL https://api.semanticscholar.org/CorpusID:32592111.

[19] D. Kirkpatrick, B. Mishra, and C.-K. Yap. Quantitative steinitz's theorems with applications to multifingered grasping. *Discrete & Computational Geometry*, 7(3):295–318, Mar 1992. ISSN 1432-0444. doi:10.1007/BF02187843. URL https://doi.org/10.1007/BF02187843.

[20] D. Kappler, J. Bohg, and S. Schaal. Leveraging big data for grasp planning. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311, 2015. doi:10.1109/ICRA.2015.7139793.

[21] M. Liu, Z. Pan, K. Xu, K. Ganguly, and D. Manocha. Deep differentiable grasp planner for high-dof grippers. *arXiv preprint arXiv:2002.01530*, 2020.

[22] A. H. Li, P. Culbertson, J. W. Burdick, and A. D. Ames. Frogger: Fast robust grasp generation via the min-weight metric. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6809–6816. IEEE, 2023.

[23] J. Chen, Y. Ke, and H. Wang. Bodex: Scalable and efficient robotic dexterous grasp synthesis using bilevel optimization. *arXiv preprint arXiv:2412.16490*, 2024.

[24] J. Chen, Y. Chen, J. Zhang, and H. Wang. Task-oriented dexterous hand pose synthesis using differentiable grasp wrench boundary estimator. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5281–5288. IEEE, 2024.

[25] H. Zhang, S. Christen, Z. Fan, O. Hilliges, and J. Song. Graspxl: Generating grasping motions for diverse objects at scale. In *European Conference on Computer Vision*, pages 386–403. Springer, 2025.

[26] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv preprint arXiv:1703.09312*, 2017.

[27] J. Lundell, E. Corona, T. Nguyen Le, F. Verdoja, P. Weinzaepfel, G. Rogez, F. Moreno-Noguer, and V. Kyrki. Multi-fingan: Generative coarse-to-fine sampling of multi-finger grasps. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4495–4501, 2021. doi:10.1109/ICRA48506.2021.9561228.

[28] L. Shao, F. Ferreira, M. Jorda, V. Nambiar, J. Luo, E. Solowjow, J. A. Ojea, O. Khatib, and J. Bohg. Unigrasp: Learning a unified model to grasp with multifingered robotic hands. *IEEE Robotics and Automation Letters*, 5(2):2286–2293, 2020.

[29] R. G. Regis. On the properties of positive spanning sets and positive bases. *Optimization and Engineering*, 17(1):229–262, 2016.

[30] K. M. Lynch and F. C. Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, USA, 1st edition, 2017. ISBN 1107156300.

[31] M. J. D. Powell. A hybrid method for nonlinear equations. In P. Rabinowitz, editor, *Numerical Methods for Nonlinear Algebraic Equations*, pages 87–114. Gordon and Breach, London, 1970.

[32] B. Amos and J. Z. Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *International conference on machine learning*, pages 136–145. PMLR, 2017.

[33] Exoside. Quad remesher - auto retopology - exoside. URL https://exoside.com/.

[34] L. Pineda, T. Fan, M. Monge, S. Venkataraman, P. Sodhi, R. Chen, J. Ortiz, D. DeTone, A. Wang, S. Anderson, J. Dong, B. Amos, and M. Mukadam. Theseus: A Library for Differentiable Nonlinear Optimization. *arXiv preprint arXiv:2207.09442*, 2022.

[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

8 Appendix

8.1 Metrics

Entropy Formulation: Directly calculating the entropy of the quaternion representation or the euler angles introduces bias due to the non-uniformity and/or singularities of the representation. We therefore calculate the entropy of grasp orientation by first converting it into axis-angle representation, which ensures that each orientation lies within a sphere of radius π. By finally converting the axis-angles into spherical coordinates, we can calculate the entropy of the orientation as the joint entropy of the spherical coordinates. For each object, we calculate the joint entropy of the pose χ (position χ_p, orientation χ_θ) and joint states q and report the mean over all objects. The final entropy is calculated as

$$H = \frac{1}{2}H(q) + \frac{1}{2}H(\chi) = \frac{1}{2}H(q) + \frac{1}{2}\left(H(\chi_p) + H(\chi_\theta)\right),$$

where $H(\bullet)$ denotes the entropy of the respective discretized distribution $\bullet$. We calculate the entropy by using uniform discretizations with 32 evenly spaced bins per dimension and assume each component to be independent.

Penetration Depth: The penetration metric captures the amount of penetration between the object O and the gripper $\mathcal{G}$. It is calculated as the sum of the maximal penetration depths of a discrete set of points ($p_s(O) \in R^{3000 \times 3}$) uniformly sampled from the object surface. For each of these points, we calculate the max penetration depth with respect to all gripper links $d(\cdot, \cdot)$. The final penetration depth is the mean over all worst case penetrations, i.e.

$$P = \frac{1}{N_{\text{obj}} \times N_{\text{grasps}}} \sum_{(O,\mathcal{G})} \max_{p \in p_s(O)} \left(d(p, \mathcal{G})\right),$$

where N_{obj} denotes the number of objects and N_{grasps} the number of predicted grasps for each object (i.e. 32).

Success Rate with regard to applied Force:
We employ two different success metrics
i) $Succ^1$: As used in [13], this denotes the success rate for withstanding a force applied along at least one of the main canonical axes (i.e., $\pm x$, $\pm y$, or $\pm z$). To ensure consistency with prior work, this metric is used by default to determine successful grasps (e.g., in calculating Unique Grasp Rate (UGR) or Entropy (H)), unless specified otherwise:

$$\text{Succ}^1 = \frac{1}{N_{\text{obj}} \times N_{\text{grasps}}} \sum_{(O,\mathcal{G})} \mathbb{1}\left[\bigvee_{a \in \{\pm x, \pm y, \pm z\}} \text{Success}(O, \mathcal{G}, a)\right],$$

where Success is true if the center of gravity of the object O grasped by the grasp $\mathcal{G}$ remains stationary (within a radius of 3cm) during the application of the force along the axis a.
ii) $Succ^3$: A more rigorous formulation of the success rate, reporting the proportion of cases where the grasp can withstand forces applied along all three main canonical axes:

$$\text{Succ}^3 = \frac{1}{N_{\text{obj}} \times N_{\text{grasps}}} \sum_{(O,\mathcal{G})} \mathbb{1}\left[\bigwedge_{a \in \{\pm x, \pm y, \pm z\}} \text{Success}(O, \mathcal{G}, a)\right].$$

8.2 Metropolis-adjusted Langevin algorithm (MALA)

We base our implementation on the MALA optimizer used in [1] and highlight our key differences (shown in blue) in the pseudo code 1. We refer the reader to [1] for the implementation of the full algorithm.

Algorithm 1: Analytical Grasp Generation

Input: Initial grasp proposals $\mathcal{G}_0 = \{G_0^{(i)}\}_{i=0}^{N_{\text{grasps}}}$, Number of iterations N, temperature T.
Output: Predicted Grasps $\mathcal{G}_N$
for $t = 1$ *to* N **do**
    $\hat{\mathcal{C}}' = ...$ // Re-sample active contact points
    $(\hat{\chi}_t, \hat{q}_t) = ...$ // Calculate next grasp using SGD, based on energy
    $\hat{\mathcal{G}}_t = (\hat{\chi}_t, \hat{q}_t, \hat{\mathcal{C}}')$
    $\Delta E = E(\hat{\mathcal{G}}_t) - E(\mathcal{G}_{t-1})$
    // Update According to Metropolis-Hastings
    $\mu_t, \sigma_t = \text{mean}(E(\hat{\mathcal{G}}_t)), \text{std}(E(\hat{\mathcal{G}}_t))$
    for $i = 0$ *to* N_{grasps} **do**
        $T_t^{(i)} = T_t \cdot \left(1 + \Phi(E(\hat{\mathcal{G}}_t^{(i)}); \mu_t, \sigma_t)\right)$
        $p \sim \mathcal{U}(0, 1)$
        if $p < e^{-\Delta E^{(i)}/T_t^{(i)}}$ **then**
            $\mathcal{G}_t^{(i)} = \hat{\mathcal{G}}_t^{(i)}$ // Accept Sample
        end
        else
            $\mathcal{G}_t^{(i)} = \mathcal{G}_{t-1}^{(i)}$ // Reject Sample
        end
    end
end

8.3 Implementation Details

Simulation Environment: We use Isaac Lab [14] (along with Isaac Sim) as our simulation environment, which offers high-fidelity physics simulations and a broad selection of robotic grippers and objects. The underlying physics time step is set to 200 Hz, and actuator commands are also applied at 200 Hz. We disable gravity for both the object and the gripper and configure the actuators torque and velocity limits based on realistic, gripper-specific values taken from their respective datasheets. A proportional-derivative (PD) controller with a high proportional gain (approximately 100) is employed, effectively mimicking a position controller for each gripper. The maximum simulation time is set to 4.8 seconds, with the external force changing every 0.5 seconds. For the Shadow Hand, we replace the tendon driven actuators with direct-drive actuators, as we observed instability when using tendons during simulation.

Grasp Generation: To allow for a fair comparison, we use the same hyperparameters and optimization setup as done in [1]. We modify the handcrafted set of contact points with contact meshes. We then randomly sample a discrete set of contact points from these contact meshes using farthest point sampling to ensure good coverage. These points are then used as potential contact points $\mathcal{C}$ for the optimization process. For Pinch and Precision grasps, we select a subset of the contact meshes (i.e. fingertips of thumb, index for pinch, index and middle for precision grasps).

We reset all environments after n_{reset} steps if the energy E_t is below one standard deviation of the overall distribution (i.e., $p_{th} = 0.8413$). Finally, we optimize over a total of 7000 steps and evaluate the final grasp proposals using our simulation environment.

In case of the unconstrained optimization problem, we use the implementation of Theseus [34] and unroll the last four optimization steps to compute the gradient. We implement the barrier functions as

$$\text{barrier}(\hat{\gamma}) = -h_{[h<0]} - (1/\tau \cdot \log(h+\varepsilon))_{[h\geq 0]},$$

with $\tau = 100$, $\epsilon = 10^{-6}$ and $h = [\hat{\gamma} - 1, u - \hat{\gamma}]$.

Dataset Processing: We postprocess the data set introduced in [1]. Besides re-meshing for better collision detection in Isaac Sim, we also re-scale the assets (which by default are normalized to the unit cube). To this end, we first scale each mesh by a factor of 0.08, which should make the assets roughly fit within an 8cm bounding box. However, we observe that not all meshes follow the unit-cube convention, meaning that a few become disproportionately small or large after this initial scaling. To address these issues, we compute the bounding box dimensions for each rescaled mesh and apply a corrective rescaling step: if the smallest axis exceeds 8cm, we downscale the mesh to ensure that the smallest axis is at most 8cm wide. If the largest axis falls below 7cm, we upscale it to ensure the the largest axis is at least 7cm large. This ensures that all assets remain within a consistent and physically meaningful size range.

Baseline Methods:

- DexGraspNet [1]: We use the official DexGraspNet implementation as the basis for our experiments. To improve performance, we adjust the relative weighting between force and torque by a factor of 5, as we found this leads to higher success rates. Specifically, we compute the energy as:

$$E_{FC} = \|Gc\|_W \text{ with } W = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix}.$$

- GenDexGrasp [13]: We implement GenDexGrasp by adding their normal based distance weighting. Hence, the final energy formulation becomes:

$$E_{\text{dis}}^{\texttt{GenDex}} = \sum_{c\in\mathcal{C}} e^{(1-\langle n_c, n_o\rangle)}\|d(c,O)\|_2$$

where n_c is the normal vector at the fingertip, n_o is the surface normal of the closest point on the object, and $d(c,O)$ denotes the closest distance from the contact point c to the object surface O.

- MultiGripperDataset [2]: We follow the same procedures as the authors and use the code provided by the authors for the supported grippers. More specifically, we use GraspIt! to sample an initial set of grasps using a maximal amount of 100000 steps. Similarly to [2],we sample the final (in our case 32) grasps using farthest point sampling. These grasps are then evaluated using our evaluation environment.

- TDG [24]: We use the implementation provided by the authors. We modify the relative force/torque weighting to $1/5$ and scale the energy term by a factor of 100 to make sure that the energy term has a comparable magnitude with the other baseline methods. Furthermore, we additionally add the normal weighting term from GenDexGrasp, as this has lead to improved performance.

- GraspQP (ours): We use a friction coefficient of 0.2 and upper limit on $\hat{\lambda}$ of 50. We again, use a relative force/torque weighting of $1/5$ and use the normal based distance weighting from [13].

Numerical Results and Success Rates: For completeness, we report the unique grasp rate and entropy (Tab. 3) and grasp success rate (Tab. 4) for all force thresholds, baselines and grasp types.

Table 3: extscUnique Grasp Rate (UGR) and Entropy for Various Interaction Forces: We list the UGR and entropy (H) for various interaction forces, ranging from 1 to 10 N. Succ[3] denotes the success rate to withstand all interaction forces along all three main directions ($\pm x$, $\pm y$, $\pm z$), and Succ[1] denotes the success rate along either one of the main directions.

Grasp Type	Method	Optimizer	Allegro UGR↑	H↑	Shadow Hand UGR↑	H↑	Robotiq3f UGR↑	H↑	Robotiq2f UGR↑	H↑	Ability Hand UGR↑	H↑	Overall UGR↑	H↑
1N														
power	DexGraspNet	MALA	68%	3.54	43%	3.12	50%	3.08	32%	3.28	20%	1.99	43%	3.00
		MALA*	69%	3.51	49%	3.23	61%	3.27	**33%**	3.28	20%	1.86	46%	3.03
	GenDexGrasp	MALA*	74%	3.56	53%	3.34	73%	3.36	32%	**3.40**	24%	2.22	51%	3.18
	TDG	MALA*	69%	3.51	53%	3.34	77%	3.31	29%	3.30	27%	2.36	51%	3.16
	GraspQP (ours)	MALA*	**77%**	**3.61**	**67%**	**3.56**	**78%**	**3.39**	29%	3.23	**37%**	**2.73**	**58%**	**3.30**
pinch	DexGraspNet	MALA	32%	2.52	29%	2.50	17%	1.93		–	15%	1.73	23%	2.17
		MALA*	36%	2.69	26%	2.27	22%	2.10		=	13%	1.51	24%	2.14
	GenDexGrasp	MALA*	44%	2.96	37%	2.77	40%	**2.90**		–	23%	2.18	36%	2.70
	TDG	MALA*	43%	2.99	33%	2.61	38%	2.79		–	21%	1.99	34%	2.59
	GraspQP (ours)	MALA*	**50%**	**3.11**	**40%**	**2.87**	**41%**	2.89		–	**29%**	**2.35**	**40%**	**2.81**
precision	DexGraspNet	MALA	39%	2.82	35%	2.60	23%	2.26		–	17%	1.82	29%	2.38
		MALA*	43%	2.94	32%	2.48	31%	2.54		=	16%	1.63	30%	2.40
	GenDexGrasp	MALA*	48%	3.11	42%	2.94	45%	3.08		–	21%	1.98	39%	2.78
	TDG	MALA*	49%	3.27	43%	3.01	47%	3.07		–	19%	1.85	39%	2.80
	GraspQP (ours)	MALA*	**57%**	**3.36**	**48%**	**3.07**	**49%**	**3.11**		–	**32%**	**2.53**	**46%**	**3.02**
2N														
power	DexGraspNet	MALA	67%	3.52	42%	3.09	45%	2.97	32%	3.27	20%	1.90	41%	2.95
		MALA*	68%	3.51	47%	3.22	55%	3.13	**33%**	3.27	19%	1.84	44%	2.99
	GenDexGrasp	MALA*	75%	3.56	52%	3.31	69%	**3.30**	32%	**3.40**	24%	2.29	50%	3.17
	TDG	MALA*	68%	3.51	52%	3.33	72%	3.25	29%	3.29	27%	2.33	49%	3.14
	GraspQP (ours)	MALA*	**76%**	**3.60**	**67%**	**3.55**	71%	3.26	29%	3.23	**37%**	**2.70**	**56%**	**3.27**
pinch	DexGraspNet	MALA	32%	2.52	23%	2.23	9%	1.08		–	13%	1.56	19%	1.85
		MALA*	36%	2.67	19%	1.95	11%	1.27		=	12%	1.34	20%	1.81
	GenDexGrasp	MALA*	44%	2.94	30%	2.54	17%	1.77		–	21%	2.06	28%	2.33
	TDG	MALA*	43%	2.99	28%	2.41	**20%**	1.96		–	19%	1.87	27%	2.31
	GraspQP (ours)	MALA*	**50%**	**3.11**	**34%**	**2.65**	19%	**1.96**		–	**27%**	**2.24**	**32%**	**2.49**
precision	DexGraspNet	MALA	39%	2.80	31%	2.49	14%	1.59		–	17%	1.75	25%	2.15
		MALA*	42%	2.91	28%	2.35	18%	1.84		=	14%	1.52	26%	2.15
	GenDexGrasp	MALA*	49%	3.12	38%	2.81	26%	2.38		–	21%	1.98	34%	2.57
	TDG	MALA*	50%	3.27	39%	2.90	28%	**2.41**		–	18%	1.78	34%	2.59
	GraspQP (ours)	MALA*	**56%**	**3.35**	**44%**	**2.99**	28%	2.39		–	**30%**	**2.39**	**40%**	**2.78**
5N														
power	DexGraspNet	MALA	66%	3.51	35%	2.86	33%	2.56	30%	3.21	18%	1.77	37%	2.78
		MALA*	68%	3.51	41%	3.04	40%	2.76	31%	3.23	18%	1.72	40%	2.85
	GenDexGrasp	MALA*	74%	3.56	47%	3.16	51%	2.93	32%	**3.38**	23%	2.19	45%	3.04
	TDG	MALA*	69%	3.51	47%	3.19	54%	2.93	28%	3.28	25%	2.16	45%	3.01
	GraspQP (ours)	MALA*	**76%**	**3.60**	**63%**	**3.46**	**57%**	**3.02**	29%	3.22	**35%**	**2.63**	**52%**	**3.19**
pinch	DexGraspNet	MALA	31%	2.47	10%	1.28	4%	0.49		–	11%	1.25	14%	1.37
		MALA*	35%	2.64	7%	0.92	3%	0.35		=	10%	1.09	14%	1.25
	GenDexGrasp	MALA*	44%	2.92	14%	1.57	4%	0.39		–	17%	1.78	20%	1.66
	TDG	MALA*	42%	2.96	13%	1.43	**5%**	**0.58**		–	16%	1.72	19%	1.67
	GraspQP (ours)	MALA*	**49%**	**3.11**	**17%**	**1.84**	4%	0.45		–	**23%**	**2.09**	**23%**	**1.87**
precision	DexGraspNet	MALA	37%	2.73	14%	1.64	5%	0.55		–	14%	1.53	18%	1.61
		MALA*	43%	2.94	14%	1.52	6%	0.65		=	11%	1.32	18%	1.61
	GenDexGrasp	MALA*	48%	3.06	23%	2.18	**9%**	1.07		–	18%	1.80	25%	2.02
	TDG	MALA*	49%	3.26	23%	2.30	8%	0.87		–	16%	1.59	24%	2.00
	GraspQP (ours)	MALA*	**55%**	**3.32**	**29%**	**2.45**	9%	**1.14**		–	**26%**	**2.28**	**30%**	**2.30**
10N														
power	DexGraspNet	MALA	65%	3.49	23%	2.31	22%	2.01	29%	3.16	17%	1.65	31%	2.53
		MALA*	67%	3.50	28%	2.55	28%	2.31	**31%**	3.20	17%	1.66	34%	2.64
	GenDexGrasp	MALA*	72%	3.53	35%	2.83	37%	2.60	31%	**3.34**	21%	2.01	39%	2.86
	TDG	MALA*	67%	3.49	36%	2.79	**42%**	2.58	28%	3.24	23%	2.07	39%	2.83
	GraspQP (ours)	MALA*	**76%**	**3.59**	**48%**	**3.12**	45%	**2.75**	28%	3.21	**33%**	**2.56**	**46%**	**3.05**
pinch	DexGraspNet	MALA	30%	2.45	2%	0.16	**2%**	**0.22**		–	8%	0.92	11%	0.94
		MALA*	34%	2.58	1%	0.02	1%	0.05		=	7%	0.80	11%	0.86
	GenDexGrasp	MALA*	42%	2.89	2%	0.25	1%	0.14		–	14%	1.61	15%	1.22
	TDG	MALA*	42%	2.94	3%	0.35	2%	0.18		–	12%	1.25	15%	1.18
	GraspQP (ours)	MALA*	**48%**	**3.08**	2%	0.23	2%	0.14		–	**19%**	**1.80**	**18%**	**1.31**
precision	DexGraspNet	MALA	37%	2.72	3%	0.24	2%	0.20		–	11%	1.27	13%	1.11
		MALA*	41%	2.91	2%	0.15	2%	0.19		=	10%	1.12	14%	1.09
	GenDexGrasp	MALA*	46%	3.05	5%	0.72	**3%**	**0.33**		–	15%	1.49	17%	1.40
	TDG	MALA*	48%	3.21	6%	0.73	2%	0.19		–	12%	1.37	17%	1.38
	GraspQP (ours)	MALA*	**54%**	**3.30**	**8%**	**0.91**	3%	0.27		–	**22%**	**2.02**	**22%**	**1.63**

Table 4: GRASP SUCCESS RATES FOR VARIOUS INTERACTION FORCES: We list the raw success rates for various interaction forces, ranging from 1 to 10 N. $Succ^3$ denotes the success rate to withstand all interaction forces along all three main directions ($\pm x, \pm y, \pm z$), and $Succ^1$ denotes the success rate along either one of the main directions.

| Grasp Type	Method	Optimizer	Allegro Succ¹↑	Succ³↑	Shadow Hand Succ¹↑	Succ³↑	Robotiq3f Succ¹↑	Succ³↑	Robotiq2f Succ¹↑	Succ³↑	Ability Hand Succ¹↑	Succ³↑	Overall Succ¹↑	Succ³↑
1N														
power	DexGraspNet	MALA	71%	50%	45%	37%	53%	34%	64%	58%	21%	12%	51%	38%
	DexGraspNet	MALA*	71%	48%	50%	41%	64%	42%	70%	66%	20%	10%	55%	41%
	GenDexGrasp	MALA*	_77%_	_55%_	_55%_	47%	76%	54%	83%	76%	25%	14%	63%	_49%_
	TDG	MALA*	71%	48%	55%	_49%_	_79%_	_55%_	_86%_	_79%_	_28%_	_15%_	_64%_	49%
	GraspQP (ours)	MALA*	**80%**	**57%**	**70%**	**63%**	**80%**	**56%**	**88%**	**82%**	**39%**	**24%**	**71%**	**56%**
pinch	DexGraspNet	MALA	33%	27%	31%	19%	19%	5%			16%	10%	25%	15%
	DexGraspNet	MALA*	38%	31%	28%	16%	23%	6%			14%	9%	26%	16%
	GenDexGrasp	MALA*	_46%_	_41%_	_39%_	_26%_	_43%_	10%			_24%_	16%	_38%_	_23%_
	TDG	MALA*	45%	40%	35%	23%	41%	**12%**			22%	15%	36%	22%
	GraspQP (ours)	MALA*	**52%**	**47%**	**42%**	**29%**	**44%**	10%			**30%**	**21%**	**42%**	**27%**
precision	DexGraspNet	MALA	41%	32%	37%	26%	24%	9%			18%	11%	30%	20%
	DexGraspNet	MALA*	44%	36%	34%	24%	33%	10%			16%	11%	32%	20%
	GenDexGrasp	MALA*	50%	41%	_45%_	_34%_	46%	15%			_23%_	_15%_	41%	26%
	TDG	MALA*	_51%_	_42%_	45%	34%	_49%_	_20%_			20%	13%	_41%_	_27%_
	GraspQP (ours)	MALA*	**59%**	**49%**	**50%**	**39%**	**51%**	_19%_			**33%**	**22%**	**48%**	**32%**
2N														
power	DexGraspNet	MALA	69%	50%	44%	33%	47%	22%	63%	57%	21%	11%	49%	35%
	DexGraspNet	MALA*	71%	50%	49%	37%	57%	28%	71%	64%	20%	10%	54%	38%
	GenDexGrasp	MALA*	_77%_	_54%_	_54%_	43%	71%	37%	83%	76%	25%	_14%_	_62%_	_45%_
	TDG	MALA*	70%	49%	54%	_44%_	**74%**	_39%_	_85%_	_78%_	_28%_	14%	_62%_	45%
	GraspQP (ours)	MALA*	**78%**	**56%**	**69%**	**59%**	_73%_	**41%**	**88%**	**82%**	**38%**	**23%**	**69%**	**52%**
pinch	DexGraspNet	MALA	33%	27%	25%	10%	9%	**2%**			14%	8%	20%	12%
	DexGraspNet	MALA*	37%	30%	21%	7%	11%	1%			13%	7%	20%	11%
	GenDexGrasp	MALA*	_46%_	40%	_32%_	_13%_	18%	1%			_22%_	_13%_	_30%_	_17%_
	TDG	MALA*	44%	39%	30%	12%	_21%_	2%			20%	13%	29%	16%
	GraspQP (ours)	MALA*	**52%**	**47%**	**36%**	**16%**	20%	**2%**			**28%**	**18%**	**34%**	**21%**
precision	DexGraspNet	MALA	40%	32%	32%	15%	15%	_3%_			18%	10%	26%	15%
	DexGraspNet	MALA*	44%	36%	30%	14%	18%	3%			15%	10%	27%	16%
	GenDexGrasp	MALA*	51%	40%	40%	23%	27%	3%			_23%_	_13%_	35%	20%
	TDG	MALA*	_52%_	_42%_	_41%_	25%	_29%_	**4%**			20%	12%	_36%_	_21%_
	GraspQP (ours)	MALA*	**58%**	**48%**	**47%**	**28%**	**30%**	3%			**31%**	**20%**	**42%**	**25%**
5N														
power	DexGraspNet	MALA	68%	49%	37%	18%	35%	8%	61%	54%	19%	9%	44%	28%
	DexGraspNet	MALA*	71%	48%	42%	22%	42%	10%	69%	62%	19%	9%	49%	30%
	GenDexGrasp	MALA*	_76%_	_53%_	48%	28%	52%	13%	83%	73%	24%	12%	57%	36%
	TDG	MALA*	71%	49%	_49%_	_30%_	_55%_	14%	_85%_	_77%_	_25%_	_14%_	_57%_	_37%_
	GraspQP (ours)	MALA*	**79%**	**55%**	**65%**	**40%**	**59%**	**16%**	**88%**	**81%**	**36%**	**22%**	**65%**	**43%**
pinch	DexGraspNet	MALA	32%	24%	11%	2%	_4%_	**0%**			11%	6%	14%	8%
	DexGraspNet	MALA*	37%	30%	8%	1%	3%	_0%_			11%	5%	15%	9%
	GenDexGrasp	MALA*	_46%_	_39%_	_15%_	2%	4%	0%			_18%_	7%	_21%_	12%
	TDG	MALA*	44%	38%	14%	_3%_	**5%**	0%			17%	_8%_	20%	_12%_
	GraspQP (ours)	MALA*	**51%**	**45%**	**19%**	**3%**	4%	0%			**24%**	**12%**	**24%**	**15%**
precision	DexGraspNet	MALA	39%	30%	15%	3%	5%	0%			15%	7%	19%	10%
	DexGraspNet	MALA*	44%	35%	14%	2%	6%	**1%**			12%	6%	19%	11%
	GenDexGrasp	MALA*	50%	39%	_24%_	4%	_9%_	0%			20%	10%	26%	13%
	TDG	MALA*	_51%_	_41%_	24%	_6%_	8%	_1%_			17%	9%	25%	_14%_
	GraspQP (ours)	MALA*	**57%**	**47%**	**31%**	**8%**	**10%**	0%			**28%**	**15%**	**32%**	**18%**
10N														
power	DexGraspNet	MALA	67%	47%	24%	7%	22%	2%	60%	49%	17%	8%	38%	23%
	DexGraspNet	MALA*	70%	46%	29%	9%	29%	2%	68%	57%	18%	8%	43%	24%
	GenDexGrasp	MALA*	_75%_	_51%_	35%	12%	38%	_4%_	81%	69%	22%	10%	50%	29%
	TDG	MALA*	69%	47%	_37%_	_13%_	_43%_	3%	_84%_	_72%_	_24%_	_13%_	_51%_	_30%_
	GraspQP (ours)	MALA*	**78%**	**54%**	**50%**	**20%**	**47%**	**5%**	**87%**	**76%**	**34%**	**21%**	**59%**	**35%**
pinch	DexGraspNet	MALA	31%	22%	2%	**0%**	**2%**	**0%**			8%	3%	11%	6%
	DexGraspNet	MALA*	36%	27%	1%	_0%_	1%	_0%_			8%	2%	12%	7%
	GenDexGrasp	MALA*	_44%_	_37%_	2%	0%	_2%_	0%			_15%_	5%	_16%_	_10%_
	TDG	MALA*	43%	34%	_3%_	0%	2%	0%			12%	_6%_	15%	10%
	GraspQP (ours)	MALA*	**50%**	**41%**	**3%**	0%	2%	0%			**19%**	**6%**	**18%**	**12%**
precision	DexGraspNet	MALA	39%	30%	3%	_0%_	2%	**0%**			12%	4%	14%	8%
	DexGraspNet	MALA*	43%	33%	2%	0%	2%	_0%_			10%	4%	14%	9%
	GenDexGrasp	MALA*	48%	37%	_6%_	0%	**3%**	0%			_16%_	_7%_	_18%_	11%
	TDG	MALA*	_50%_	_39%_	6%	0%	2%	0%			14%	6%	18%	_11%_
	GraspQP (ours)	MALA*	**57%**	**45%**	**8%**	**1%**	_3%_	0%			**24%**	**11%**	**23%**	**14%**

8.4.1 Contact Maps

We additionally provide heatmaps that visualize the contact regions for each gripper and grasp type. To generate these maps, we compute the distance from each object vertex $v \in V(O)$ to the gripper surface for every grasp pose $G \in \mathcal{G}$. The color intensity $c(v)$ at each vertex is then calculated as the normalized sum of the exponentiated negative distances across all grasp poses: $c(v) = \frac{1}{|\mathcal{G}|} \sum_{G \in \mathcal{G}} \left(\frac{\exp(-10 \cdot d(v,G))}{\sum_{\vartheta \in V(O)} \exp(-10 \cdot d(\vartheta,G))} \right)$. Finally, the resulting intensities are visualized using the viridis colormap, where high-intensity regions appear yellow and low-intensity regions appear blue.
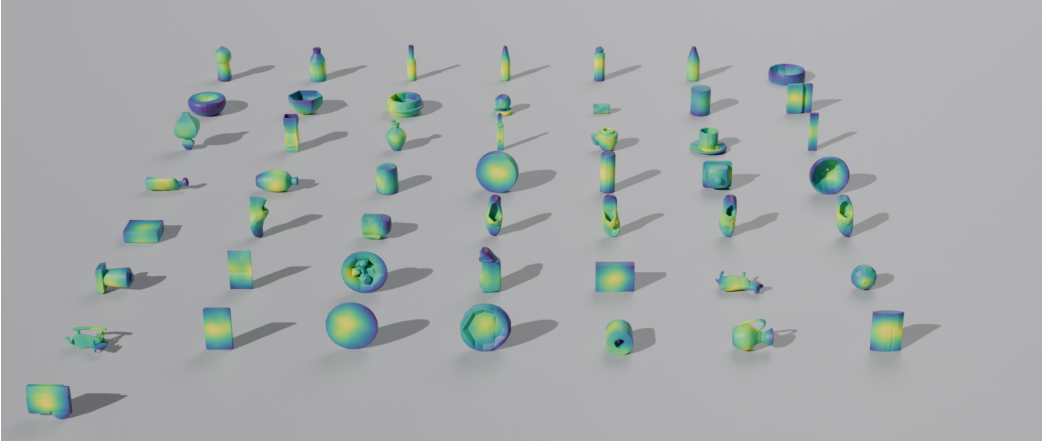
(a) Ability Hand - default
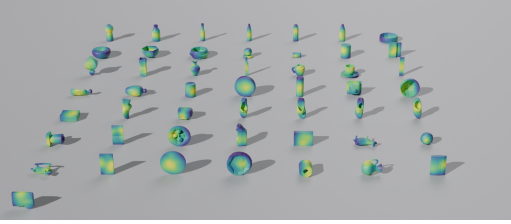
(b) Ability Hand - precision
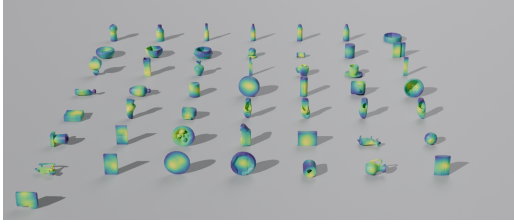
(c) Ability Hand - pinch

Figure 5: Contact heatmaps for various objects using the Ability Hand. The color intensity (mapped with the viridis colormap) indicates regions of frequent contact across all grasp poses: yellow represents high-contact areas, while blue indicates low-contact regions. Notably, many objects show concentrated contact regions near their center of gravity.
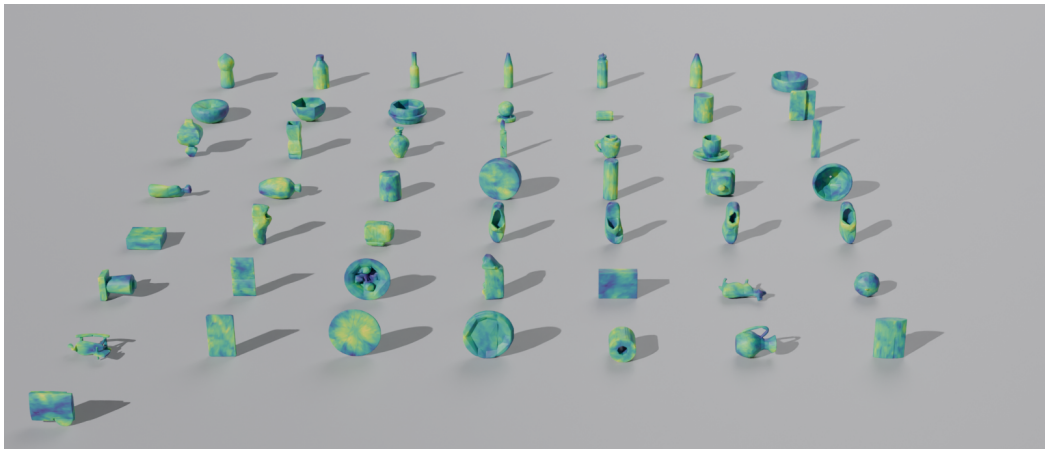
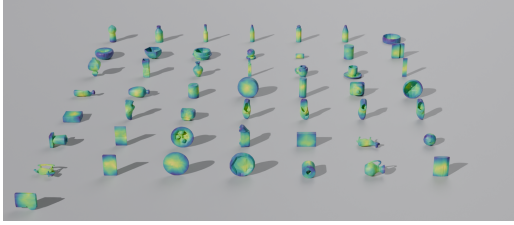(a) Roboti 3F - default
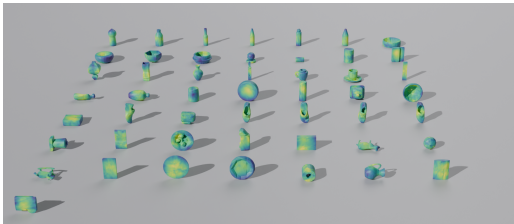
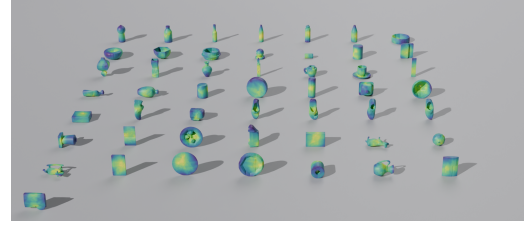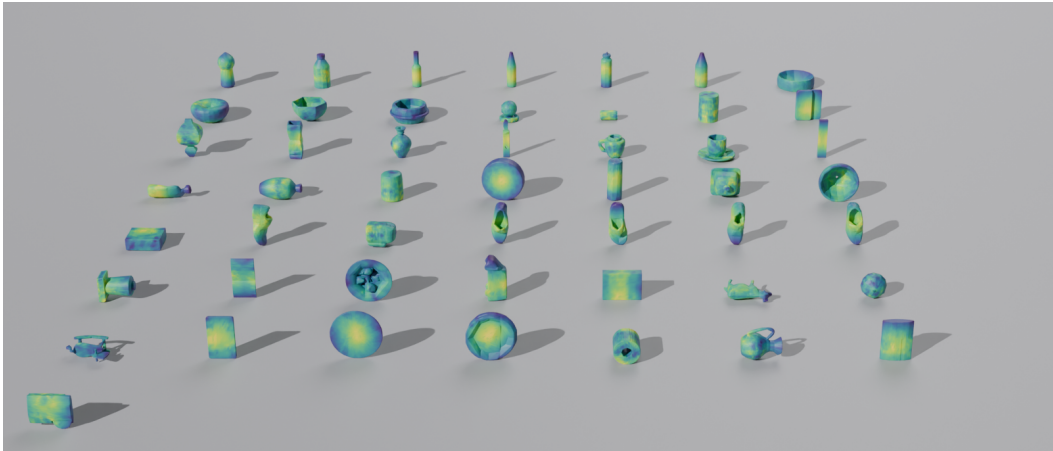(b) Roboti 3F - precision
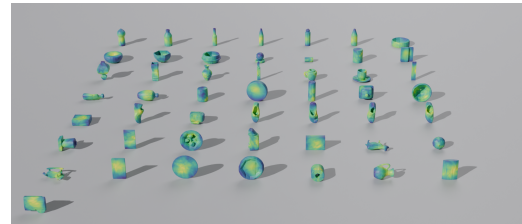
(c) Roboti 3F - pinch

Figure 6: Contact heatmaps for various objects using the Robotiq 3F. The color intensity (mapped with the viridis colormap) indicates regions of frequent contact across all grasp poses: yellow represents high-contact areas, while blue indicates low-contact regions.
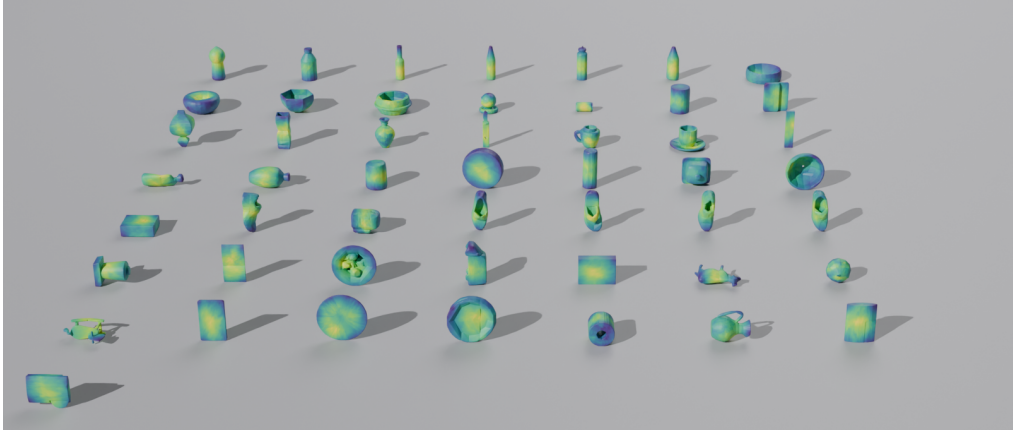
(a) Shadow Hand- default

(b) Shadow Hand- precision

(c) Shadow Hand- pinch

Figure 7: Contact heatmaps for various objects using the Shadow Hand. The color intensity (mapped with the viridis colormap) indicates regions of frequent contact across all grasp poses: yellow represents high-contact areas, while blue indicates low-contact regions. Notably, many objects show concentrated contact regions near their center of gravity.
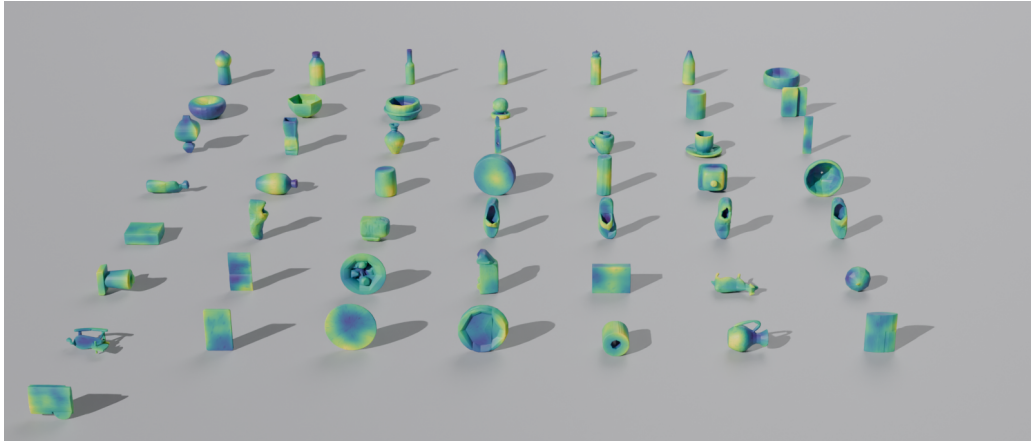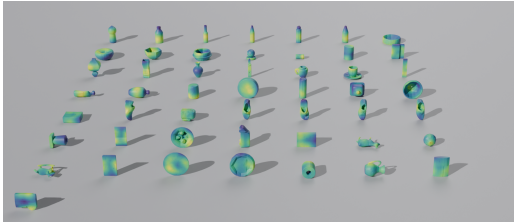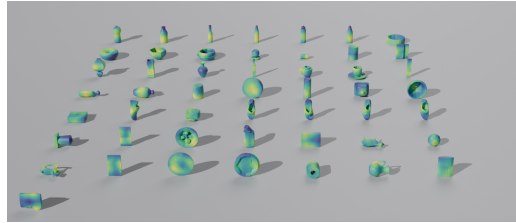

Figure 8: Robotiq 2f140

Figure 9: Contact heatmaps for various objects using the Robotiq 2f140 Gripper. The color intensity (mapped with the viridis colormap) indicates regions of frequent contact across all grasp poses: yellow represents high-contact areas, while blue indicates low-contact regions. Notably, many objects show concentrated contact regions near their center of gravity.

(a) Allegro - default

(b) Allegro - precision

(c) Allegro - pinch

Figure 10: Contact heatmaps for various objects using the Allegro hand. The color intensity (mapped with the viridis colormap) indicates regions of frequent contact across all grasp poses: yellow represents high-contact areas, while blue indicates low-contact regions. Notably, many objects show concentrated contact regions near their center of gravity.