Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

DexEvolve: Evolutionary Optimization for Robust and Diverse Dexterous Grasp Synthesis

René Zurbrügg, Andrei Cramariuc, Marco Hutter
ETH Zürich

Abstract—**Dexterous grasping is fundamental to robotics, yet data-driven grasp prediction relies heavily on large and diverse datasets that are costly to generate and typically limited to a narrow set of gripper morphologies. Analytical grasp synthesis can scale data collection, but the required simplifying assumptions often yield physically infeasible grasps. These grasps must be filtered in high-fidelity simulators, substantially reducing both the number of retained grasps and their diversity.**

We propose a scalable generate-and-refine pipeline for synthesizing large-scale, diverse, and physically feasible grasps. Instead of using high-fidelity simulators solely for verification and filtering, we leverage them as an optimization stage that continuously improves grasp quality without discarding pre-computed candidates. Specifically, we initialize an evolutionary search with a seed set of analytically generated, potentially suboptimal grasps. We then refine these proposals directly in a high-fidelity simulator (Isaac Sim) using an asynchronous, gradient-free evolutionary algorithm that improves stability while maintaining diversity. In addition, this refinement stage can be guided by human preferences and/or domain-specific quality metrics without requiring differentiable objectives. We further distill the refined grasp distribution into a diffusion model for robust real-world deployment and highlight the role of diversity in both training and deployment. Experiments on our newly introduced Handles dataset and a DexGraspNet subset show that the proposed method produces over 120 distinct stable grasps per object, a 1.7–$6\times$ improvement over unrefined analytical seeds.
Project page: https://dexevolve.github.io

Fig. 1: **Visualization of grasps generated by our evolutionary refinement.** We visualize grasps for two *Handles* assets and one *Objects* asset. Our evolutionary approach generates diverse, physically stable grasps by refining candidates directly in high-fidelity simulation.

I. Introduction

Dexterous grasping is a fundamental capability in robotics, enabling a wide range of applications, including industrial automation, service robotics, and human-robot collaboration. Traditional approaches to grasp synthesis primarily rely on analytical methods that sample contact points on an object's surface and evaluate their force-closure properties by solving conical optimization programs [20]. While analytical optimization can generate diverse grasp candidates, many of them are not dynamically feasible due to simplifying assumptions in the underlying models, such as coarse contact dynamics, friction approximations, and simplified collision handling.

Two-stage approaches first generate grasp candidates and then filter them using high-fidelity, non-differentiable physics simulators such as Isaac Sim [27] or MuJoCo [34]. Classical methods such as GraspIt! [24] use shape-guided heuristics and quality metrics to generate and evaluate candidates. More recent methods [5, 6, 21, 36, 38, 45] relax or approximate traditional force- or form-closure conditions, enabling integration with gradient-based optimization and kinematic simulators. Despite these advances, both paradigms share a fundamental limitation: the separation between generation and validation. Grasps are proposed under simplified or relaxed assumptions and are only then verified in high-fidelity simulation. This post-selection strategy is inherently sample-inefficient, since most proposals are discarded during verification, and even those that pass are often suboptimal because the generation process cannot leverage direct physical feedback during optimization. We argue that, rather than treating high-fidelity simulation as a binary accept/reject filter, grasp proposals can benefit substantially from direct optimization within the physics simulator.

Direct simulator-based refinement is challenging for two reasons. First, evaluating grasp stability in a physics simulator is significantly more expensive than performing kinematic checks. It requires applying interaction forces over a predefined time horizon and verifying that the object remains stable. Second, most high-fidelity simulators are non-differentiable, which prevents the use of conventional gradient-based optimization methods. Therefore, the few works that investigate direct optimization in high-fidelity simulators are typically limited to wrist-pose optimization and do not consider the full grasp configuration in order to simplify the search space [12, 13].

To address these shortcomings, we propose a two-stage approach that uses high-fidelity simulators for grasp refinement without requiring differentiability. We start with a diverse set of analytically generated grasp proposals from [45]. In practice, our framework does not restrict the initialization strategy, and other methods are equally applicable for obtaining an initial pool of grasp candidates. We then refine these proposals directly in a high-fidelity simulator (Isaac Sim) using an asynchronous, gradient-free evolutionary algorithm. The asynchronous and parallel nature of the process allows many unpromising candidate grasps to be quickly discarded, thereby allocating computation efficiently toward promising regions of the grasp space.

Concretely, we reinterpret high-fidelity simulation not as a binary accept/reject filter, but as a black-box objective that can be optimized directly: instead of discarding the majority of seed grasps, we iteratively *repair* and *improve* them through evolutionary refinement. This yields a substantially larger number of stable, dynamically feasible grasps (Fig. 1) while preserving coverage of multiple grasp modes. Our contributions include density-aware selection and archive-based insertion rules that explicitly suppress redundant grasp clusters and promote exploration of underrepresented regions of the grasp space. Moreover, because our refinement is gradient-free, it naturally supports steering with non-differentiable, domain-specific metrics (e.g., interaction objectives for handles) and even human preference signals. Finally, we distill the refined grasp distribution into a conditional diffusion model to enable fast inference from partial observations, bridging large-scale dataset generation and deployable grasp prediction.

In summary, our contributions are:

- **Generate-and-refine evolutionary grasp synthesis:** A scalable pipeline that uses an asynchronous, gradient-free evolutionary optimizer to refine an initial seed set of dexterous grasp configurations directly in a high-fidelity, non-differentiable simulator, with density-aware selection and archive-based insertion to mitigate mode collapse and preserve multimodal grasp diversity.
- **Gradient-free guidance:** Our framework allows the refinement step to be guided toward human preferences and/or task-specific quality metrics without requiring differentiable objectives.
- **Real-world deployment:** We train a point-cloud-conditioned diffusion model on the refined dataset, yielding grasp predictions that are not only more diverse but also applicable to real-world deployment from partial point-cloud observations.

II. Related Work

A. Dexterous Grasping Datasets and Grasp Synthesis

A wide range of datasets [4, 8, 17, 36, 38, 42] provide synthesized grasp poses for rigid objects across different grippers. These datasets vary in scale, ranging from a few hundred thousand grasps [17, 19] to large-scale datasets containing one million or more grasp annotations [4, 8, 17, 38]. In terms of gripper diversity, many focus on parallel-jaw grippers [8, 31], while others incorporate dexterous hands such as the Shadow or Allegro Hand [4, 36, 45].

These datasets are typically generated through grasp synthesis methods. Traditional sampling-based approaches [4, 7, 23] explore possible hand poses to optimize grasp quality metrics, but are computationally expensive for high-DoF dexterous hands. Recent gradient-based methods [17, 21, 35, 36] leverage differentiable grasp quality metrics for faster convergence. However, classical metrics such as Q_1 [9] are not inherently differentiable, leading to relaxed formulations [19, 21] that may oversimplify grasp stability by neglecting frictional forces [21, 38]. Other approaches incorporate differentiable kinematic simulations with velocity-based metrics [35, 36].

Our dataset extends this diversity by contributing a new collection of door handle and knob assets with grasp annotations for the XHand [29]. By modeling these assets after products available in physical stores, our dataset enables real-world testing and supports reproducible evaluation and sim-to-real transfer across different works. While [37] provides synthetic door knobs for reinforcement learning, it does not include grasp predictions or corresponding physical assets.

B. Evolutionary Algorithms and Quality Diversity

Evolutionary algorithms [30] are gradient-free methods that improve a population of solutions through mutation and fitness-based selection. Genetic algorithms [1] encode solutions as genes and evolve them through *selection*, *crossover*, and *mutation*. They have been used for tasks such as system identification [2, 16] and hyperparameter tuning [40].

In grasp prediction, their use remains limited: Shukla *et al.* [32] optimize two-finger gripper poses, Huber *et al.* [12] optimize wrist placement for diverse hand poses, and Huber *et al.* [14] release a large-scale two-finger dataset generated with QDG-6DOF [13]. However, no recent work optimizes full dexterous grasp configurations beyond wrist placement or parallel-jaw setups.

C. Diffusion Models for Grasping

Recent work has explored diffusion and other generative models for sampling multimodal dexterous grasps conditioned on high-level intent and partial observations, including Transformer-based architectures with point-cloud conditioning [41] and approaches that use language as an explicit conditioning signal [22, 39]. Zhong *et al.* [43] target more universal dexterous grasping by incorporating physics-aware modeling into the grasp generation pipeline and serve as the core building block for our diffusion model. Additionally, [44] introduces direct policy optimization for grasp-pose diffusion to align grasp generation with human feedback.

A key commonality of these generative approaches is that they typically fit or distill a grasp distribution from existing data rather than synthesizing and validating new grasps through direct optimization in high-fidelity simulators. This distillation enables efficient inference and conditioning on partial observations, such as point clouds, for fast on-hardware deployment. We likewise distill a grasp distribution using an architecture similar to DexGraspAnything [43], but add geometric consistency constraints and diffuse the full wrist pose rather than only its position. Unlike pure

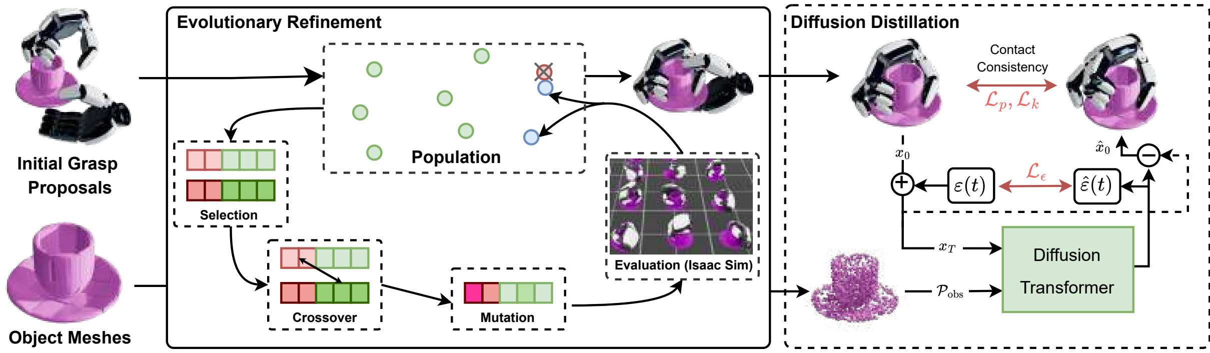

Fig. 2: **Method overview.** Given object meshes, we first generate an initial set of analytical grasp proposals. We then refine these candidates in a high-fidelity simulator using an asynchronous, gradient-free evolutionary loop comprising selection, crossover, mutation, and physics-based evaluation. This yields a diverse set of physically feasible grasps. Finally, we distill the refined grasp distribution into a diffusion model conditioned on noisy point-cloud observations, enabling efficient sampling at inference time under partial observability.

generative modeling, our framework uses simulator-in-the-loop evolutionary refinement to explicitly improve the physical feasibility and stability of grasp configurations and then distills the refined distribution for efficient point-cloud-conditioned inference.

III. Method

Given an accurate 3D object model and a dexterous hand, our goal is to synthesize a diverse set of feasible grasp configurations $\mathcal{G} = \{G_i\}_{i=1}^{N}$ that satisfy force closure, i.e., that can resist arbitrary bounded external wrenches by applying admissible contact forces subject to Coulomb friction constraints. We first generate a diverse seed set with an analytical optimizer (GraspQP [45]), although any grasp generation method could serve this purpose. We then refine these candidates using the evolutionary procedure detailed in Sec. III-B. For real-world deployment, we distill the resulting grasp dataset into a point-cloud-conditioned diffusion model, described in Sec. III-D. An overview of the full pipeline is shown in Fig. 2.

A. Grasp Representation

For a given robotic gripper with n_q joints, we parameterize each grasp as $G = (\chi, q, \Delta q_{cmd})$, where $\chi = (\chi_{pos}, \chi_{orient}) \in SE(3)$ denotes the wrist pose, $q \in \mathbb{R}^{n_q}$ denotes the joint states of the robot, and $\Delta q_{cmd} \in \mathbb{R}^{n_q}$ denotes the desired delta joint commands used to apply a gripping force. We denote the corresponding projections by $G_{\text{pos}} = \chi_{pos}$, $G_{\text{orient}} = \chi_{orient}$, and $G_{\text{joints}} = q$.

B. Evolutionary Refinement

In the following, we outline our evolutionary refinement procedure, which optimizes dexterous grasp candidates directly in Isaac Sim through massively parallel rollouts with early rejection. We introduce the simulator-derived fitness and the genetic operators, namely density-aware selection, mutation, and structured crossover, that improve stability while preserving multimodality. Following evolutionary computation conventions, we refer to each grasp as an individual and to the set of grasp candidates as the population or archive.

Evolutionary refinement in high-dimensional grasp spaces is prone to *mode collapse*, i.e., repeatedly producing minor variants of the same grasp. To maintain diversity, we treat the population as an *archive* and gate insertions using a novelty criterion, as done in quality-diversity methods [26]: a newly evaluated candidate is inserted only if it is sufficiently far from the current archive under a task-relevant metric $d(\cdot, \cdot)$; otherwise, it competes locally with its nearest neighbor. Additionally, to promote diversity during reproduction, we bias the selection process toward lower-density regions.

Insertion of New Grasps: This step updates the archive by inserting novel candidates and replacing redundant ones only when they improve fitness. Given a candidate G and the current population $\mathcal{G} = \{G_j\}_{j=1}^{|\mathcal{G}|}$, we compute its nearest neighbor

$$j^\star = \arg\min_j d(G, G_j), \tag{1}$$

where $d(\cdot, \cdot)$ is a weighted distance metric over position, orientation, and joint configuration:

$$\begin{aligned} d(G, G') = \lambda_{\text{pos}} \left\| G_{\text{pos}} - G'_{\text{pos}} \right\|_2^2 + \lambda_{\text{orient}} \left\| G_{\text{orient}} - G'_{\text{orient}} \right\|_2^2 \\ + \lambda_{\text{joints}} \left\| G_{\text{joints}} - G'_{\text{joints}} \right\|_2^2 . \end{aligned} \tag{2}$$

If $d(G, G_{j^\star}) \geq \tau$, the candidate is considered novel and inserted: $\mathcal{G} \leftarrow \mathcal{G} \cup \{G\}$. If $d(G, G_{j^\star}) < \tau$, we do not grow the archive. Instead, the candidate competes with its nearest neighbor and replaces it only if it improves fitness:

$$G_{j^\star} \leftarrow G \quad \text{if} \quad \mathcal{F}(G) > \mathcal{F}(G_{j^\star}). \tag{3}$$

This replacement rule preserves diversity by preventing redundant insertions while still allowing local improvements to update existing individuals.

Fitness Function: The base fitness function for the evolutionary algorithm measures grasp quality and is defined as a weighted sum of three components: an environment-lifetime score, a contact-distance penalty, and a penetration penalty.

$$\mathcal{F}(G) = E_{lifetime} - w_{dis} E_{dis} - w_{pen} E_{pen}, \tag{4}$$

where w_{dis} and w_{pen} denote the respective weights. $E_{lifetime}$ measures robustness under a fixed disturbance protocol: forces are applied sequentially along each canonical axis ($\pm x, \pm y, \pm z$) over a predefined horizon, and $E_{lifetime}$ counts the number of simulation steps survived before failure. Partial stability is rewarded, and full-horizon survival yields the maximum score. In practice, $E_{lifetime}$ dominates the fitness. E_{dis} is a weak shaping term that penalizes grasps far from contact-feasible configurations; it is computed only over the $n_c = 12$ closest contact candidates, so fingers intentionally far from the object are not penalized. E_{pen} penalizes hand–object interpenetration, and we terminate evaluations early if it exceeds a threshold. Eq. (4) is the base fitness; with preference guidance, an additional $+w_{\text{reward}} E_{\text{reward}}$ term is added (Supp. Mat. VIII-E). Full forms are provided in Supp. Mat. VIII-A.

Selection: We rely on tournament selection [11] to select parents for crossover. In each tournament, we randomly sample k individuals from the population and select the two highest-scoring candidates. As discussed at the beginning of Sec. III-B, selecting purely by raw fitness can accelerate mode collapse [15]. To preserve diversity, we therefore perform tournament selection using a density-aware reweighting of each individual's score $\mathcal{F}'(G_i)$ based on the current population:

$$\mathcal{F}'_i = \mathcal{F}(G_i) \cdot \left(\sum_{G_j \in \mathcal{B}_r(G_i)} \left(1 - \frac{d(G_i, G_j)^p}{r^p} \right) \right)^{-1}, \quad (5)$$

where r and p are tunable parameters that define how strongly clustering is suppressed. Intuitively, if an individual G_i has many similar grasp poses G_j within a ball of radius r, the probability of selecting G_i is reduced. In the extreme case in which G_i has no neighbors besides itself, i.e., $\mathcal{B}_r(G_i) = \{G_i\}$, its fitness remains unchanged ($\mathcal{F}' \to \mathcal{F}$). Conversely, if all grasps are identical, their fitness is reduced by a factor of $1/|\mathcal{G}|$ ($\mathcal{F} \to \frac{\mathcal{F}}{|\mathcal{G}|}$). While the selection strategy and archive-based insertion use the same metric and serve a similar purpose, the selection strategy steers reproduction toward promising samples, whereas the insertion strategy ensures that the final population remains diverse after the algorithm terminates.

Mutation: The mutation operator is applied to the first two parameters of the grasp configuration, namely pose and joint states, with probability p_{mutation}. The mutation step size is drawn from a zero-mean Gaussian distribution with standard deviation $\Sigma = \text{diag}(\Sigma_\chi, \Sigma_q)$. We use an Euler-angle representation for the wrist pose χ, allowing mutation to be applied independently to roll, pitch, and yaw.

Crossover: Genetic algorithms typically use a crossover operator to combine genes from two parents into a new individual. However, naively combining grasp configurations from two parents yields many infeasible grasps. In line with prior work [5, 45], we observe that grasps tend to follow distinct grasp taxonomies with similar finger configurations. Furthermore, many objects can be grasped from different wrist poses using similar finger configurations. We therefore exchange either the finger configuration or the full grasp-pose information as a whole, effectively fixing the crossover point.

Fig. 3: **Contact point selection for dexterous grasping.** Two views of a dexterous hand (XHand) grasping a cup from the Objects dataset. Light blue points indicate candidate contact points on the object surface that lie within the contact distance threshold, while dark blue points denote points outside this threshold. Green points represent the final active contact points selected via farthest point sampling (FPS), which are used to compute grasping commands through the contact Jacobian.

More formally, we select two parents G_i, G_j and form an offspring G_c by exchanging their joint vectors q while keeping χ fixed, with probability $p_{\text{crossover}}$.

Resampling of Contact Points and Grasping Commands: In contrast to analytical optimization approaches in which the active contact points $\mathcal{C} = \{c_0, ..., c_{n_c}\}$ are fixed [6] or randomly resampled at each iteration with a fixed probability [17], we recompute both the assigned contact points and the grasping commands for each new offspring grasp, as shown in Fig. 3. This adaptive resampling strategy improves efficiency by ensuring that valid grasps are not rejected due to suboptimal contact-point selection.

For each grasp proposal G_i, we first identify the set of all potential contact points $\mathcal{C}_i$ using a ball query, selecting all points on the object surface within a predefined distance threshold from the hand. From this candidate set, we select the n_c active contact points $\mathcal{C}'_i \subseteq \mathcal{C}_i$ using farthest point sampling (FPS), which ensures spatially diverse potential contact locations. Finally, we compute the grasping commands by solving for the joint commands $\Delta\tilde{q}$ that best achieve the desired normal forces at each contact point:

$$\Delta q_{cmd} = \arg\min_{\Delta\tilde{q}} ||J \cdot \Delta\tilde{q} - N_o||_2, \quad (6)$$

where $J = [J_0, ..., J_{n_c}]$ denotes the stacked contact Jacobians for the active contact points, N_o is the desired normal force vector for each contact point, and Δq_{cmd} represents the commanded change in joint states.

C. Evolutionary Steering and Preference Alignment

An advantage of our evolutionary approach is its flexibility in incorporating diverse steering signals through the fitness function. Unlike gradient-based methods that require differentiable objectives, our evolutionary framework can directly optimize non-differentiable metrics. The fitness function presented in Sec. III-B already demonstrates this capability: the lifetime metric and contact-distance calculation include contact-point resampling operations, both of which are not fully differentiable. In general, we can steer evolutionary refinement either by (i) modifying tournament selection to

Fig. 4: **Selection of our IKEA Handles dataset.** We provide 90 geometrically distinct handle assets with multiple texture variations, yielding 190 unique simulation-ready assets in total. Each handle includes high-resolution collision geometry, realistic textures, and articulated joints with compliant degrees of freedom to support stable grasp synthesis. All assets are based on commercially available products from ikea.com and can be used directly in Isaac Sim.

incorporate pairwise preferences, or (ii) augmenting the fitness with absolute, task-specific scores, such as grasp energies.

For human preference alignment, we train a PointNet++ [28] network to predict preference scores for grasp candidates based on hand keypoints and object point-cloud configurations, using 1,000 human pairwise annotations. This learned preference model is incorporated directly into the fitness function, biasing the evolutionary process toward grasps that align with human notions of natural or desirable grasping strategies while maintaining physical stability. Importantly, our framework is agnostic to the preference acquisition method: it can accommodate learned reward models such as our PointNet++ predictor, direct human annotation during evolution, or any other preference signal that can score or rank grasp candidates.

D. Diffusion Distillation

After generating our grasp-pose dataset, we train a diffusion model to enable real-world inference from noisy point clouds. We build on the DexGraspAnything [43] architecture, which diffuses only wrist position and joint states, but extend it to diffuse the complete grasp configuration, including end-effector orientation. We parameterize orientation using the standard 6-DoF representation given by the first two columns of the rotation matrix [3, 10]. We further incorporate a differentiable hand model [45] to compute hand keypoints and surface samples through forward kinematics. This enables geometric supervision during training by defining loss terms not only on noise magnitude, but also on keypoint distance and object penetration between ground-truth and predicted hand models.

Concretely, as illustrated in Algorithm 2, we (i) normalize a ground-truth grasp, (ii) sample a diffusion time step and corrupt the grasp with Gaussian noise, and (iii) train the network to predict the injected noise conditioned on the observed partial/noisy point cloud. Beyond the standard denoising objective, we use the hand model to map the predicted grasp back to 3D hand geometry and add two intuitive regularizers: a keypoint-consistency loss that keeps potential hand contact points aligned with those of the target grasp, and a penetration penalty that discourages hand–object interpenetration by penalizing hand surface samples that lie inside the object. Finally, we schedule the collision weight over diffusion steps so that feasibility is emphasized most strongly when the sample is close to the initial sampling distribution, effectively allowing early grasps in the diffusion schedule to be in collision.

IV. Handles Dataset

We provide a total of 90 distinct handle assets collected from the IKEA website (190 when including texture variations), each with detailed collision geometry and high-quality textures suitable for simulation. To ensure stable contact simulation, we re-mesh each collision model by recursively subdividing triangles until a sufficiently high resolution is reached, with a maximum edge length of 1cm per triangle. This is important for Isaac Sim's Signed Distance Field (SDF)-based collision handling: with coarse collision meshes, we observed noticeable interpenetration and unstable contacts. Each handle is attached to a prismatic joint that translates along the z-axis. To introduce mild compliance and encourage force-closure grasps, we additionally add two prismatic joints with small travel limits ($\pm 1.5\,\mathrm{cm}$, i.e., $3\,\mathrm{cm}$ total) along the x- and y-axes. This prevents weak grasps that only "hold" the handle by a few millimeters along the compliant direction from being evaluated as stable, and effectively pushes the solutions toward grasps with stronger force-closure properties. We release the final dataset as articulated `.usd` assets that can be used directly in Isaac Sim. An overview of all handles is shown in Fig. 4.

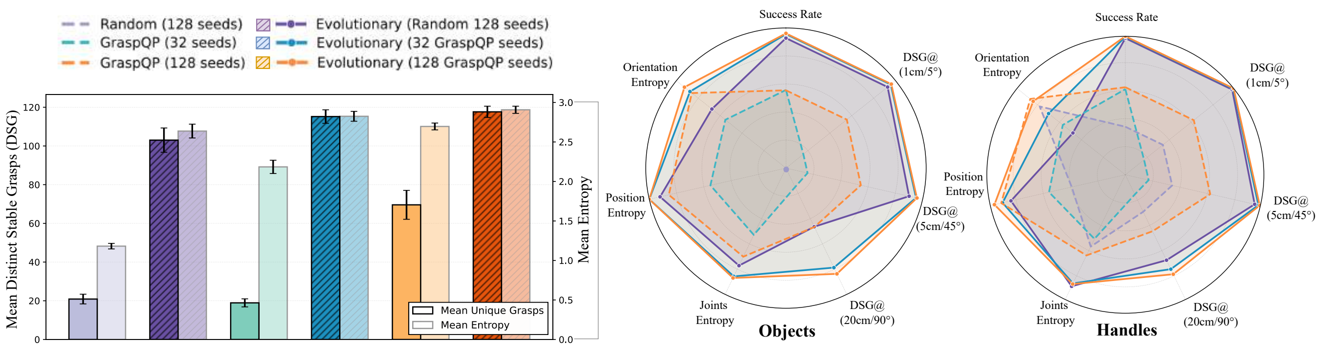

Fig. 5: **Grasp quality and diversity comparison.** Radar plots comparing synthesized grasp distributions across generation strategies, reporting success rate, DSG@2cm/DSG@20cm, and marginal entropies of position, orientation, and joints. Brighter colors indicate DSG, lighter bars indicate entropy; hatched bars denote evolutionary refinement, and non-hatched bars denote baselines. Refinement improves stability while preserving diversity across both datasets.

V. Experiments

A. Experimental Setup

Robotic Gripper: We evaluate our method using the XHand gripper from Robotera. However, our pipeline can be directly applied to any gripper supported in IsaacLab. For the XHand, we use a contact-point set focused on the fingertips, as illustrated in Fig. 3. To better match real-world interaction properties, we define two friction materials per asset: metallic, low-friction surfaces use $\mu = 0.2$, while higher-friction materials, such as silicone, use $\mu = 0.8$.

Simulation Environment: For the object evaluation, we use a subset of 50 assets from [38] and their corresponding IsaacLab integration from [45]. For the Handles dataset, we select 50 assets from the handle dataset described in Sec. IV, each initialized with the main prismatic joint translating along the z-axis.

Evolutionary Refinement: We initialize the evolutionary algorithm with a population size of $S = 32$ or $S = 128$ and run refinement for a total of 10,000 evolutionary steps. We use tournament selection with size $k = 4$, a mutation probability of $p_{\text{mutation}} = 0.75$, and a crossover probability of $p_{\text{crossover}} = 0.2$. Mutations are applied with Gaussian step sizes $\Sigma = \text{diag}(\Sigma_\chi, \Sigma_q)$, using $\Sigma_\chi = (0.025, 0.05)$ for the pose components (translation/orientation) and $\Sigma_q = 0.04$ for the joint states. For density-aware selection (Eq. 5), we set the neighborhood radius to $r = 0.65$ and the power to $p = 2.0$. Moreover, because PhysX does not reliably handle assets initialized in collision, we perform a few de-penetration steps after mutation to avoid spawning in-collision configurations and ensure valid initialization. Finally, we cap the archive size at $|\mathcal{P}| \leq 1024$ and, once this limit is exceeded, prune the archive by ranking grasps by score and applying farthest-point sampling.

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

Success Rate: The total ratio of grasps that successfully hold the object without dropping it during simulation.

DistinctStableGrasps: The unique-grasp metric captures the number of distinct, stable grasps the method can generate. We calculate this metric by discretizing successful grasp poses into bins defined by three resolution parameters: δ_r for position, δ_ϕ for orientation (Euler angles), and δ_q for joint states. We then count the number of unique grasps under this quantization and report DistinctStableGrasps (DSG) as the total number of unique, stable grasps at the specified resolution. We evaluate at three resolutions: DSG@(20cm, 90°, 90°), DSG@(2cm, 45°, 45°), and DSG@(1cm, 5°, 5°).

Entropy (H): For each object, we calculate the joint entropy of the distribution of successful grasps over position, orientation, and joint states, following [45]. We report either the average entropy over position, orientation, and joint states, or each value independently, depending on the analysis context.

Evaluation Procedure: We assess the stability of each grasp proposal using Isaac Lab [25] following prior evaluation protocols [38, 45]. For each asset, we reset the simulation, place the hand at the proposed wrist pose, execute the corresponding grasp-closing command, and test stability under external disturbances over a fixed rollout horizon. For the *Objects* dataset, we follow the standard disturbance test by sequentially applying forces along the canonical axes and declaring success if the object remains stably grasped, within the allowed pose thresholds, throughout at least one of the three axes. For the *Handles* dataset, we evaluate stability by applying a pulling force along the compliant direction (negative z-axis) and deem a grasp successful if the handle remains stably held throughout the rollout.

B. Influence of Evolutionary Refinement

To systematically assess the impact of evolutionary refinement on grasp quality and diversity, we analyze grasp distributions generated by different initialization strategies on representative test sets from both the *Objects* and *Handles* datasets (Fig. 5). Our investigation addresses two key questions: (1) whether evolutionary refinement improves grasp stability while maintaining diversity, and (2) how the choice of initial population affects the final grasp distribution. Because the Handles dataset is domain-specific, we also evaluate on a 50-asset subset of DexGraspNet [38] to verify that the gains extend beyond handles.

Our results reveal several important findings. First, evolutionary refinement consistently and substantially improves both stability and unique grasp coverage across all initialization strategies and both datasets, increasing success rates and DSG@(<20cm,<90°) metrics close to saturation. Critically, these gains in stability do not come at the expense of diversity.

TABLE I: **Cross-hand generalization on *Objects*.** Distinct stable grasps at 5 cm resolution (DSG@5cm) and average entropy H before and after evolutionary refinement, initialized with 32-seed GraspQP.

	Robotiq 2F	Robotiq 3F	Allegro	Shadow
DSG@5cm	$9 \rightarrow \mathbf{80}$	$25 \rightarrow \mathbf{86}$	$25 \rightarrow \mathbf{123}$	$17 \rightarrow \mathbf{110}$
H	$1.5 \rightarrow \mathbf{2.7}$	$2.3 \rightarrow \mathbf{2.7}$	$2.4 \rightarrow \mathbf{2.9}$	$2.1 \rightarrow \mathbf{2.8}$

The refined distributions preserve, and often increase, entropy across pose and joint spaces. We report DSG alongside entropy as the two metrics capture complementary properties: entropy measures distributional spread, whereas DSG counts the number of physically distinct stable grasps at a given resolution. Second, while increasing the number of analytically generated seeds from GraspQP from 32 to 128 improves the baseline, evolutionary refinement consistently outperforms unrefined methods at both seed counts, improving DSG from 19 to 114 for 32 seeds and from 69 to 118 for 128 seeds. This demonstrates that simulator-in-the-loop optimization is essential for generating physically feasible grasps beyond what analytical methods can achieve alone.

Perhaps most surprisingly, evolutionary refinement starting from random initialization achieves competitive success rates and unique-grasp counts, even when the initial random distribution yields zero successful grasps, as observed in the Objects dataset. This robustness underscores our method's capacity to synthesize diverse, stable grasps from highly suboptimal starting points. However, this comes with a notable trade-off: randomly initialized refinement exhibits markedly lower entropy, particularly in orientation and, to a lesser extent, joint configurations. This suggests that, while our evolutionary approach can recover stability from poor initializations, analytical seeding remains important for maintaining a rich, multimodal grasp distribution that spans the full space of feasible solutions.

Generalization across Robotic Hands: The framework is hand-agnostic. Table I reports DSG@5cm and entropy H on *Objects* for four additional grippers, using 32-seed GraspQP initialization followed by evolutionary refinement. The gains are consistent across both parallel-jaw and fully dexterous hands, indicating that the benefit is not specific to a single morphology.

Convergence Behavior: To characterize the sample efficiency and convergence properties of our evolutionary refinement approach, we visualize grasp quality and diversity metrics as a function of simulator environment steps in Fig. 6. We initialize the population with 32 analytical seeds and monitor both the mean number of unique successful grasps (blue, left axis) and the mean entropy across position, orientation, and joint-configuration spaces (red, right axis) throughout the optimization process.

This convergence behavior demonstrates two key properties of our approach. First, evolutionary refinement is highly sample-efficient, achieving near-optimal performance within a limited interaction budget of 10k steps. Second, the method reliably converges to distributions that simultaneously maximize grasp stability and preserve diversity, avoiding common failure modes such as premature convergence to a single grasp mode or uncontrolled diversity at the expense of physical feasibility.

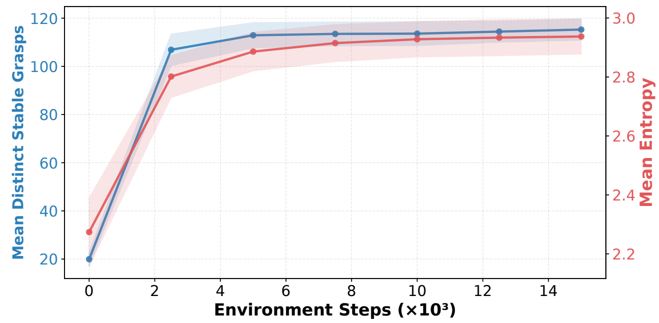

Fig. 6: **Convergence of evolutionary refinement.** Mean distinct stable grasps (blue, left axis) and mean entropy (red, right axis) as a function of simulator environment steps, initialized with 32 analytical seeds. Both metrics improve rapidly during the first 2.5k steps and then converge to a stable plateau by approximately 10k steps. The simultaneous saturation of both stability and diversity metrics shows that evolutionary refinement efficiently balances grasp quality and coverage within a limited interaction budget.

Computational Cost: Our pipeline trades *offline* simulation cost for dataset quality. On matched hardware, generating 128 grasps per object takes approximately 2.4 min/object ($\approx$1.1 s/grasp) for our method, compared with 3.0 min/object ($\approx$1.4 s/grasp) for GraspQP [45]. Increasing the number of analytical seeds does not close the gap. As shown in [45] (Fig. 4), the number of unique grasps exhibits diminishing returns and saturates at roughly 90 for [45] and 60 for [38]. This suggests that the main bottleneck is candidate *quality*, not candidate count. Fig. 6 further shows that more than 90% of the final quality is reached within the first 2,500 of 10,000 steps, indicating that early stopping can substantially reduce cost. About 50% of runtime is spent on explicit de-penetration, so improved collision handling could further reduce this overhead. Because refinement runs in a massively parallel simulator, it scales well with GPU parallelism.

C. Effect of Training a Diffusion Model

A natural alternative to simulator-in-the-loop evolutionary refinement is to leverage generative modeling: train a diffusion model on the analytically seeded grasp distribution and sample novel grasps from the learned implicit manifold. To evaluate this approach as a data-quality ablation, we fix the diffusion architecture and vary only the training data. We train a conditional diffusion model on the initial seed distribution, using both 32 and 128 seeds from the *Objects* dataset, and compare the quality and diversity of generated grasps against both the raw analytical seeds and evolutionary refinement. We additionally compare against GenDexGrasp [17] on *Objects*, where our method improves DSG@5cm from 62 to 121 and entropy from 2.7 to 3.0.

Fig. 7 presents mean unique successful grasps (solid bars, left) and mean entropy (hatched bars, right) for analytical, diffusion, and evolutionary grasp synthesis. The results reveal several key findings. First, diffusion improves upon the raw analytical initialization from GraspQP, increasing mean unique grasps from approximately 20 to 71 for 32 seeds and

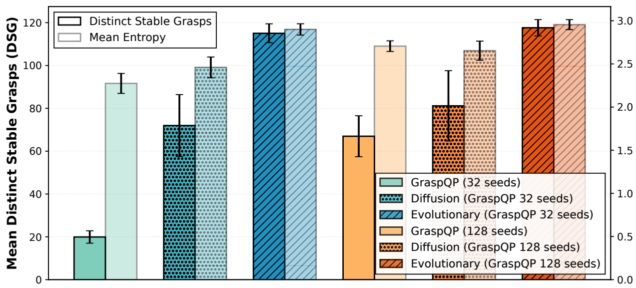

Fig. 7: **Effect of diffusion training vs. evolutionary refinement.** Mean distinct stable grasps for GraspQP, a diffusion model trained on GraspQP samples, and evolutionary refinement. Diffusion improves over the raw initialization but remains below evolutionary sampling.

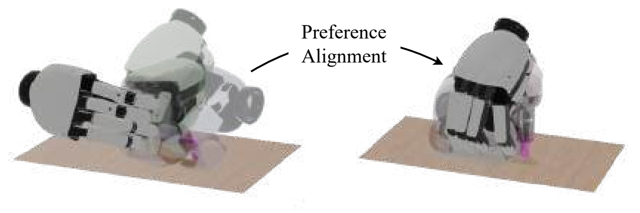

Fig. 8: **Effect of preference alignment.** Comparison of grasps without (left) and with (right) preference alignment from human feedback. The unguided evolutionary distribution produces diverse but unconventional grasps, while the preference-aligned version generates grasps that better match human intent and natural grasping patterns.

from 70 to 82 for 128 seeds, while maintaining or slightly increasing entropy. This demonstrates that the diffusion model successfully learns to interpolate within the analytical grasp manifold, generating configurations that expand coverage of the feasible grasp space.

However, diffusion trained on raw analytical seeds consistently underperforms evolutionary refinement across all settings. With 32 seeds, evolutionary refinement achieves approximately 115 unique grasps compared to 72 for diffusion, corresponding to a 60% improvement. With 128 seeds, the gap narrows but remains substantial (118 vs. 81 unique grasps, corresponding to a 46% improvement). Critically, evolutionary refinement also achieves higher average entropy, approximately 3.0, compared to diffusion, approximately 2.4–2.6, indicating that the performance gap manifests in both grasp stability and diversity. We interpret this narrowly: raw analytical seeds are not a sufficient substitute for refinement in our pipeline. Instead, the two components play complementary roles: refinement for *offline* data improvement and diffusion for *distillation* and inference from partial observations. The refined dataset can further benefit downstream tasks; for example, it greatly improves IK-reachable grasps (Appendix VIII-F) and provides richer demonstrations for RL or IL.

D. Preference-Guided Evolutionary Refinement

While evolutionary refinement generates diverse, stable grasps, the resulting configurations may not align with human preferences or task-specific requirements for natural grasping strategies. To address this, we incorporate human feedback by training a reward model on preference comparisons between grasp pairs and using it to bias the fitness evaluation during evolutionary refinement (see Sec. III-C).

We collect approximately 1,000 human pairwise preference annotations on grasp candidates, where annotators compare two grasps and select which appears more natural and socially appropriate. Preferences are based on intuitive human grasping behavior, for instance discouraging grasps that may be functional but unnatural, such as grasping a handle between the pinky and thumb or approaching from awkward angles. We train a PointNet++ network [28] that takes as input the object point cloud and hand keypoint configuration and outputs a preference score. The network is trained with a standard Bradley-Terry pairwise ranking loss to predict the probability that one grasp is preferred over another. During evolutionary refinement, we add the predicted preference score to the fitness function. Fig. 8 illustrates the effect of preference alignment. Without guidance (left), evolutionary refinement produces physically stable but unconventional grasps with awkward approach angles and unintuitive finger placements. With preference alignment (right), the method generates more natural configurations that align with typical human grasping patterns, approaching from conventional orientations with expected finger placements while maintaining physical stability. This demonstrates that preference guidance can effectively shape grasp distributions toward human-like behavior without sacrificing feasibility.

E. Real-World Robotic Deployment

To assess the real-world transferability of our approach, we deploy synthesized grasps on a physical robotic system consisting of a Franka Panda arm equipped with a dexterous XHand. We treat this evaluation as a feasibility study rather than a benchmark: it demonstrates sim-to-real transfer without real-world fine-tuning, but the deployment scale remains too small to support quantitative robustness claims.

Perception. For each target object, we capture an RGB-D sequence using an Intel RealSense L415 and select $N=7$ keyframes. Multi-view fusion improves robustness to noise, occlusion, and depth failures on reflective or transparent surfaces. We run DepthAnything-v3 [18] on each RGB frame, align predictions with sensor depth for metric scale, and apply outlier removal and normal estimation. The resulting point clouds are fused, followed by RANSAC plane removal and DBSCAN clustering to isolate the object. The final filtered cloud is used both as diffusion-model conditioning and as the planning scene representation.

Sampling and planning. Grasps are sampled from the refined diffusion model using collision-aware guidance. Motion trajectories are planned with cuRobo MotionGen [33] using voxelized scene geometry and simplified collision primitives for the table plane and cabinet backpanel.

Fig. 9 shows successful executions across diverse grasp modes and approach directions. Additional real-world results on handles, knobs, and object pickup tasks without fine-tuning are provided in the supplementary material.

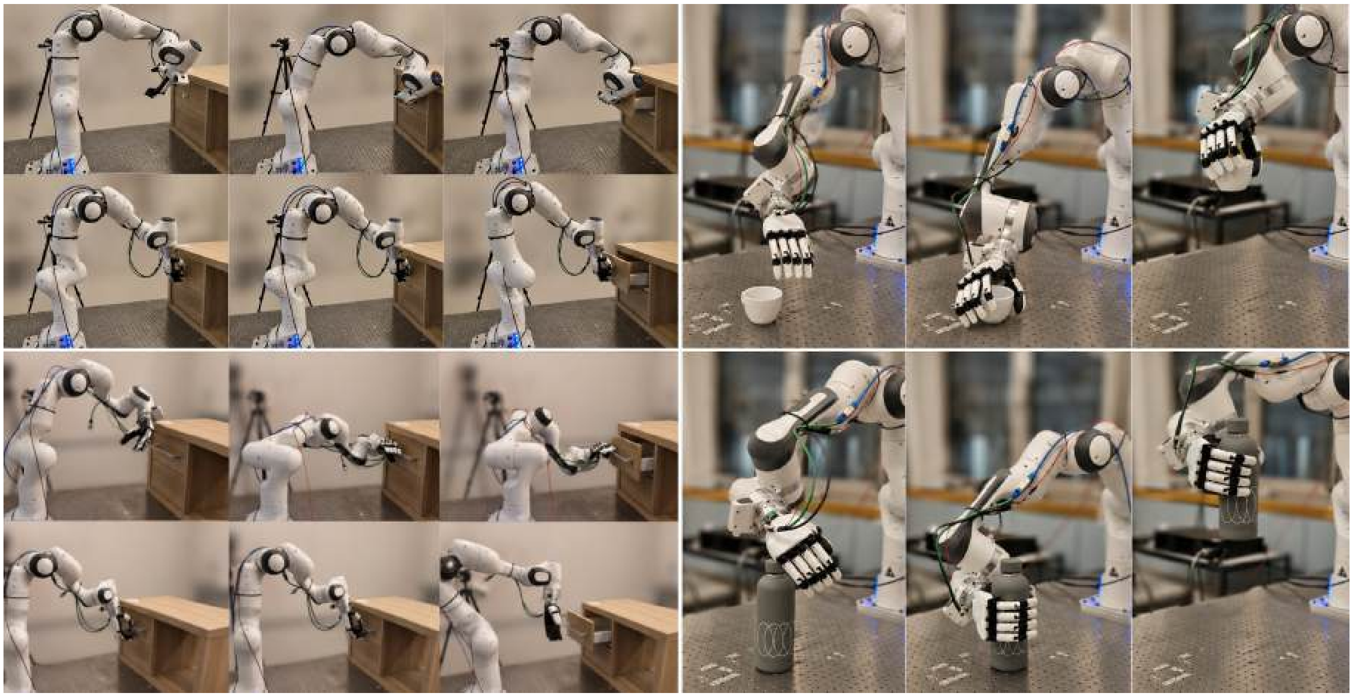

Fig. 9: **Real-world cabinet-handle rollout.** We sample handle grasps from a point-cloud-conditioned diffusion model trained on refined grasp distributions and then plan collision-aware trajectories using cuRobo. The trajectories are executed on a Franka Panda arm equipped with a dexterous hand. Three executions across different grasp modes and approach directions demonstrate successful sim-to-real transfer in contact-rich manipulation.

VI. Limitations

While our approach demonstrates significant improvements in grasp stability and diversity, several limitations remain. First, evolutionary refinement still benefits from a reasonably strong initialization: although the method can recover from poor seeds such as random grasps, runs initialized with structured or analytically seeded distributions yield higher final diversity and better coverage of the grasp manifold. This suggests that initialization quality still plays a meaningful role in the final solution set and convergence behavior.

Second, PhysX does not reliably resolve contacts when assets are initialized in penetration, requiring explicit depenetration after mutation. This step accounts for approximately $\sim 50\%$ of runtime, making it a major computational bottleneck; improved collision handling or early termination criteria (Fig. 6) could substantially reduce offline cost.

Third, the real-world deployment pipeline remains relatively slow due to its multi-stage design, including RGB-D capture, depth alignment, point-cloud processing, diffusion-based sampling, and motion planning. This limits end-to-end responsiveness and makes the system less suitable for time-critical or highly dynamic settings.

VII. Conclusion

We presented DexEvolve, a scalable generate-and-refine pipeline for synthesizing large-scale, diverse, and physically feasible dexterous grasps. Starting from analytically generated seeds, we refine grasp configurations directly in a high-fidelity, non-differentiable simulator using an asynchronous evolutionary algorithm with archive-based novelty insertion and density-aware selection, enabling efficient exploration while suppressing mode collapse. Across both the *Objects* and *Handles* datasets, evolutionary refinement substantially improves grasp stability and unique-grasp coverage, achieving over 120 distinct stable grasps per object, a 1.7–6$\times$ improvement over analytical seeds depending on the seed count (Fig. 5), while maintaining high entropy across position, orientation, and joint-configuration spaces. The refined distributions consistently outperform both unrefined analytical methods and diffusion-based alternatives in terms of physical feasibility and distributional diversity.

Beyond synthesis performance, our framework supports gradient-free steering through non-differentiable objectives, enabling preference alignment with human feedback and task-specific quality metrics. We validate real-world transfer through successful deployment on physical hardware and distill the refined grasp distribution into a point-cloud-conditioned diffusion model to enable fast inference for real-world deployment.

Acknowledgment

This research was primarily supported by the ETH AI Center. The project received funding from the European Union's Horizon Europe Framework Programme under Grant Agreement No. 101121321, the ETH Zurich Research Grant No. 22-2 ETH-47, and the Swiss National Science Foundation through the National Centre of Competence in Research (NCCR) Automation. Additional support was provided by NVIDIA, including access to NVIDIA A6000 GPUs. The authors thank Mayank Mittal for valuable discussions and assistance with Isaac Sim and PhysX.

References

[1] I.P. Androulakis and V. Venkatasubramanian. A genetic algorithmic framework for process design and optimization. *Computers & Chemical Engineering*, 15(4):217–228, 1991. ISSN 0098-1354. doi: https://doi.org/10.1016/0098-1354(91)85009-J. URL https://www.sciencedirect.com/science/article/pii/009813549185009J.

[2] Filip Bjelonic, Fabian Tischhauser, and Marco Hutter. Towards bridging the gap: Systematic sim-to-real transfer for diverse legged robots. *arXiv preprint arXiv:2509.06342*, 2025.

[3] Romain Brégier. Deep regression on manifolds: a 3d rotation case study. In *2021 International Conference on 3D Vision (3DV)*, pages 166–174. IEEE, 2021.

[4] Luis Felipe Casas, Ninad Khargonkar, Balakrishnan Prabhakaran, and Yu Xiang. Multigrippergrasp: A dataset for robotic grasping from parallel jaw grippers to dexterous hands, 2024. URL https://arxiv.org/abs/2403.09841.

[5] Jiayi Chen, Yubin Ke, Lin Peng, and He Wang. Dexonomy: Synthesizing all dexterous grasp types in a grasp taxonomy. *arXiv preprint arXiv:2504.18829*, 2025.

[6] Jiayi Chen, Yubin Ke, and He Wang. Bodex: Scalable and efficient robotic dexterous grasp synthesis using bilevel optimization. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 01–08. IEEE, 2025.

[7] Hongkai Dai, Anirudha Majumdar, and Russ 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.

[8] Clemens Eppner, Arsalan Mousavian, and Dieter 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.

[9] Carlo Ferrari and John 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.

[10] A René Geist, Jonas Frey, Mikel Zhobro, Anna Levina, and Georg Martius. Learning with 3d rotations, a hitchhiker's guide to so (3). *arXiv preprint arXiv:2404.11735*, 2024.

[11] David E. Goldberg and Kalyanmoy Deb. A comparative analysis of selection schemes used in genetic algorithms. volume 1 of *Foundations of Genetic Algorithms*, pages 69–93. Elsevier, 1991. doi: https://doi.org/10.1016/B978-0-08-050684-5.50008-2. URL https://www.sciencedirect.com/science/article/pii/B9780080506845500082.

[12] Johann Huber, François Hélénon, Mathilde Kappel, Elie Chelly, Mahdi Khoramshahi, Faïz Ben Amar, and Stéphane Doncieux. Speeding up 6-dof grasp sampling with quality-diversity. *arXiv preprint arXiv:2403.06173*, 2024.

[13] Johann Huber, François Hélénon, Hippolyte Watrelot, Faïz Ben Amar, and Stéphane Doncieux. Domain randomization for sim2real transfer of automatically generated grasping datasets. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4112–4118, 2024. doi: 10.1109/ICRA57147.2024.10610677.

[14] Johann Huber, François Hélénon, Mathilde Kappel, Ignacio de Loyola Páez-Ubieta, Santiago T Puente, Pablo Gil, Faïz Ben Amar, and Stéphane Doncieux. Qdgset: A large scale grasping dataset generated with quality-diversity. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 01–08. IEEE, 2025.

[15] Sourabh Katoch, Sumit Singh Chauhan, and Vijay Kumar. A review on genetic algorithm: past, present, and future. *Multimedia tools and applications*, 80:8091–8126, 2021.

[16] K. Kristinsson and G.A. Dumont. System identification and control using genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(5):1033–1046, 1992. doi: 10.1109/21.179842.

[17] Puhao Li, Tengyu Liu, Yuyang Li, Yixin Zhu, Yaodong Yang, and Siyuan Huang. Gendexgrasp: Generalizable dexterous grasping. *arXiv preprint arXiv:2210.00722*, 2022.

[18] Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

[19] Min Liu, Zherong Pan, Kai Xu, Kanishka Ganguly, and Dinesh Manocha. Deep differentiable grasp planner for high-dof grippers. *arXiv preprint arXiv:2002.01530*, 2020.

[20] Min Liu, Zherong Pan, Kai Xu, and Dinesh Manocha. New formulation of mixed-integer conic programming for globally optimal grasp planning. *IEEE Robotics and Automation Letters*, 5(3):4663–4670, 2020.

[21] Tengyu Liu, Zeyu Liu, Ziyuan Jiao, Yixin Zhu, and Song-Chun 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.

[22] Jiaxin Lu, Hao Kang, Haoxiang Li, Bo Liu, Yiding Yang, Qixing Huang, and Gang Hua. Ugg: Unified generative grasping. In *European Conference on Computer Vision*, pages 414–433. Springer, 2024.

[23] A.T. Miller and P.K. 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.

[24] A.T. Miller and P.K. Allen. Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, 11(4):110–122, 2004. doi: 10.1109/MRA.

2004.1371616.

[25] Mayank Mittal, Pascal Roth, James Tigue, Antoine Richard, Octi Zhang, Peter Du, Antonio Serrano-Muñoz, Xinjie Yao, René Zurbrügg, Nikita Rudin, et al. Isaac lab: A gpu-accelerated simulation framework for multi-modal robot learning. *arXiv preprint arXiv:2511.04831*, 2025.

[26] Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*, 2015.

[27] NVIDIA. Isaac Sim. URL https://github.com/isaac-sim/IsaacSim.

[28] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[29] ROBOTERA. Xhand1: Fully actuated, truly free, 2024. URL https://www.robotera.com/en/goods1/4.html.

[30] Jeffrey R Sampson. Adaptation in natural and artificial systems (john h. holland), 1976.

[31] Lin Shao, Fabio Ferreira, Mikael Jorda, Varun Nambiar, Jianlan Luo, Eugen Solowjow, Juan Aparicio Ojea, Oussama Khatib, and Jeannette Bohg. Unigrasp: Learning a unified model to grasp with multifingered robotic hands. *IEEE Robotics and Automation Letters*, 5(2):2286–2293, 2020.

[32] Priya Shukla, Hitesh Kumar, and Gora Chand Nandi. Robotic grasp manipulation using evolutionary computing and deep reinforcement learning. *Intelligent Service Robotics*, 14(1):61–77, 2021.

[33] Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, et al. curobo: Parallelized collision-free minimum-jerk robot motion generation. *arXiv preprint arXiv:2310.17274*, 2023.

[34] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. doi: 10.1109/IROS.2012.6386109.

[35] Dylan Turpin, Liquan Wang, Eric Heiden, Yun-Chun Chen, Miles Macklin, Stavros Tsogkas, Sven Dickinson, and Animesh Garg. Grasp'd: Differentiable contact-rich grasp synthesis for multi-fingered hands. In *European Conference on Computer Vision*, pages 201–221. Springer, 2022.

[36] Dylan Turpin, Tao Zhong, Shutong Zhang, Guanglei Zhu, Eric Heiden, Miles Macklin, Stavros Tsogkas, Sven Dickinson, and Animesh Garg. Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation. In *ICRA*, 2023.

[37] Yusuke Urakami, Alec Hodgkinson, Casey Carlin, Randall Leu, Luca Rigazio, and Pieter Abbeel. Doorgym: A scalable door opening environment and baseline agent. *arXiv preprint arXiv:1908.01887*, 2019.

[38] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He 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.

[39] Yi-Lin Wei, Jian-Jian Jiang, Chengyi Xing, Xian-Tuo Tan, Xiao-Ming Wu, Hao Li, Mark Cutkosky, and Wei-Shi Zheng. Grasp as you say: Language-guided dexterous grasp generation. *Advances in Neural Information Processing Systems*, 37:46881–46907, 2024.

[40] Xueli Xiao, Ming Yan, Sunitha Basodi, Chunyan Ji, and Yi Pan. Efficient hyperparameter optimization in deep learning using a variable length genetic algorithm. *arXiv preprint arXiv:2006.12703*, 2020.

[41] Guo-Hao Xu, Yi-Lin Wei, Dian Zheng, Xiao-Ming Wu, and Wei-Shi Zheng. Dexterous grasp transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17933–17942, 2024.

[42] Hui Zhang, Sammy Christen, Zicong Fan, Otmar Hilliges, and Jie Song. Graspxl: Generating grasping motions for diverse objects at scale. In *European Conference on Computer Vision*, pages 386–403. Springer, 2025.

[43] Yiming Zhong, Qi Jiang, Jingyi Yu, and Yuexin Ma. Dexgrasp anything: Towards universal robotic dexterous grasping with physics awareness. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22584–22594, 2025.

[44] Yufei Zhu, Yiming Zhong, Zemin Yang, Peishan Cong, Jingyi Yu, Xinge Zhu, and Yuexin Ma. Evolvinggrasp: Evolutionary grasp generation via efficient preference alignment. *arXiv preprint arXiv:2503.14329*, 2025.

[45] René Zurbrügg, Andrei Cramariuc, and Marco Hutter. Graspqp: Differentiable optimization of force closure for diverse and robust dexterous grasping. *arXiv preprint arXiv:2508.15002*, 2025.