Exploratory Grasping: Asymptotically Optimal Algorithms for Grasping Challenging Polyhedral Objects

Michael Danielczuk*, Ashwin Balakrishna*,
Daniel S. Brown, Shivin Devgon, Ken Goldberg
* equal contribution
{mdanielczuk, ashwin_balakrishna}@berkeley.edu

Abstract: There has been significant recent work on data-driven algorithms for learning general-purpose grasping policies. However, these policies can consistently fail to grasp challenging objects which are significantly out of the distribution of objects in the training data or which have very few high quality grasps. Motivated by such objects, we propose a novel problem setting, Exploratory Grasping, for efficiently discovering reliable grasps on an unknown polyhedral object via sequential grasping, releasing, and toppling. We formalize Exploratory Grasping as a Markov Decision Process where we assume that the robot can (1) distinguish stable poses of a polyhedral object of unknown geometry, (2) generate grasp candidates on these poses and execute them, (3) determine whether each grasp is successful, and (4) release the object into a random new pose after a grasp success or topple the object after a grasp failure. We study the theoretical complexity of Exploratory Grasping in the context of reinforcement learning and present an efficient bandit-style algorithm, Bandits for Online Rapid Grasp Exploration Strategy (BORGES), which leverages the structure of the problem to efficiently discover high performing grasps for each object stable pose. BORGES can be used to complement any general-purpose grasping algorithm with any grasp modality (parallel-jaw, suction, multi-fingered, etc) to learn policies for objects in which they exhibit persistent failures. Simulation experiments suggest that BORGES can significantly outperform both general-purpose grasping pipelines and two other online learning algorithms and achieves performance within 5% of the optimal policy within 1000 and 8000 timesteps on average across 46 challenging objects from the Dex-Net adversarial and EGAD! object datasets, respectively. Initial physical experiments suggesting that BORGES can improve grasp success rate on average over two challenging 3D printed objects by 45% over a Dex-Net baseline. See https://tinyurl.com/exp-grasping for supplementary material and videos.

Keywords: Grasping, Online Learning

1 Introduction

Robot grasping systems have a broad array of applications in manufacturing, warehousing, assistive robotics, and household automation [1, 2, 3, 4]. There has been significant recent work in analytic grasp planning algorithms [5, 6, 7, 8], but these methods can often be difficult to apply when object geometry is unknown. In recent years, robot learning has shown exciting promise by utilizing large-scale datasets of previously attempted grasps both in simulation [4, 9, 1, 10] and in physical experiments [11, 3, 1, 12] to learn general-purpose grasping policies that can generalize to objects of varying geometries. To further enable generalization, there has also been work on applying reinforcement learning to learn grasping policies [2, 3, 13, 14]. However, while these techniques have shown significant success in practice, their generality comes at a cost: a policy which aims to grasp every object may fail to generalize to specific challenging objects [15, 16].

The AUTOLab at UC Berrkeley (automation@berkeley.edu)

4th Conference on Robot Learning (CoRL 2020), Cambridge MA, USA.

Inspired by infants that repeatedly attempt to grasp a toy until they can learn reliable ways to grasp it, we consider a novel problem: Exploratory Grasping, where a robot is presented with an unknown object and learns to reliably grasp it by repeatedly attempting grasps and allowing the object pose to evolve based on grasp outcomes. The objective is for the robot to explore grasps across different object poses to reliably grasp the object from any of its stable resting poses. We then present an algorithm for Exploratory Grasping motivated by polyhedral objects with sparse grasps and adversarial geometries, which can cause persistent failures in general-purpose grasping systems [16, 15]. The intuition is that while a general-purpose grasping policy can be broadly applied to a large set of objects, it can be complemented by learned policies for specific challenging objects.

This paper contributes (1) Exploratory Grasping: an MDP formulation of the problem of discovering robust grasps across object poses through online interaction that is independent of the grasping modality (parallel-jaw, suction, multi-fingered, etc), (2) parameter-dependent performance bounds on a family of existing tabular reinforcement learning algorithms in this setting, (3) an efficient bandit-inspired algorithm, Bandits for Online Rapid Grasp Exploration Strategy (BORGES), for Exploratory Grasping, and associated no-regret guarantees, (4) simulation experiments suggesting that BORGES can significantly outperform baseline algorithms which explore via reinforcement learning or select actions using general-purpose grasping policies across 46 objects in both the Dex-Net adversarial and EGAD! object datasets, and (5) initial physical results suggesting that BORGES can improve grasp success rate on average over two challenging objects by 45% over a Dex-Net baseline.

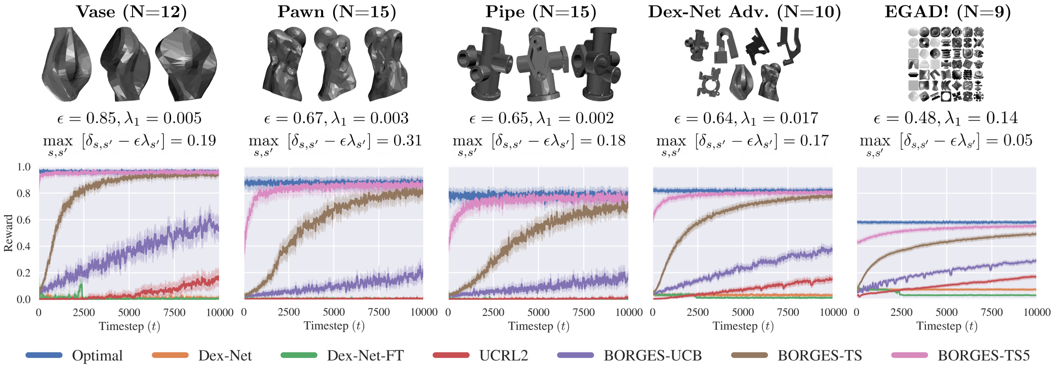

Figure 1: **Simulation Experiments:** Performance of each policy across the vase, pawn and pipe objects from the Dex-Net 2.0 adversarial object set [17] (first three columns), as well as aggregated performance over 7 Dex-Net objects (fourth column) and 39 EGAD! objects (fifth column). We report the number of distinguishable stable poses (N) with the first three shown in the second row, as well as ϵ (ground truth value for the lowest quality best grasp across poses), λ_1 (least likely stable pose probability) and $\max_{s,s'} [\delta_{s,s'} - \epsilon\lambda_{s'}]$ (maximum difference in transitioning via toppling and transitioning via grasping to a new pose) values. For the datasets, we give mean values and show the most likely stable pose of each object. We visualize policy performance in the learning curves in the lowest row. BORGES quickly converges to near-optimal performance even when Assumption 5.1 is violated while the other algorithms fail to reach optimal performance.

2 Related Work

Work in analytic robot grasping assumes knowledge of object geometry and pose to design geometric grasp planning algorithms [5, 7, 8, 6]. Recently, learning-based general-purpose algorithms have emerged for planning grasps robust to uncertainty in object geometry and sensing on a wide range of objects with data-driven strategies [4, 11, 10, 1, 18, 9] and online exploration through reinforcement learning [2, 3]. The latter approaches have been very effective in learning end-to-end policies for grasp planning for a variety of objects. In contrast, we focus on learning policies for the specific challenging objects that pose problems for these systems [15, 16].

Significant prior work studies multi-armed bandit frameworks for online grasp exploration [19, 20, 21, 22, 23, 24, 25], but primarily focus on settings when some geometric knowledge is known or grasp exploration is limited to a single object pose. In contrast, we consider a formulation where the robot must learn grasps across all poses of the object without human supervision. Laskey et al. [20] consider the setting where some prior geometric knowledge is known, but present an algorithm for 2D objects that does not use visual inputs. Li et al. [22] and Oberlin and Tellex [21] relax these assumptions by exploring grasps for a single stable pose of 3D object with RGB or depth observations.

We extend these ideas by repeatedly dropping the object and exploring grasps in all encountered stable poses. Thus, BORGES naturally explores grasps over the distribution of possible stable poses [26] to learn a robust policy which can reliably grasp the object when randomly dropped in the workspace.

A key requirement for successful exploration of grasps on an object is exploring the space of its resting stable poses, since the object will necessarily be in one of these poses during grasp attempts. There has been significant prior work on orienting parts into specific stable poses through a series of parallel jaw gripper movements [27], toppling actions [28], and squeezing actions [29]. However, these approaches require knowledge of object geometry apriori. When object geometry is not known, but assumed to be polyhedral, prior work [26] has established that repeatedly dropping the object from a known initial distribution of poses onto a flat workspace results in a stationary distribution over stable poses. This dropping procedure provides a useful primitive for reaching new stable poses of an object. We leverage this insight to formulate the Exploratory Grasping problem and design algorithms that address this setting to discover high-quality grasps for different object stable poses.

3 Exploratory Grasping: Problem Statement

Given a single unknown polyhedral object on a planar workspace, the objective is to learn a grasping policy that maximizes the likelihood of grasp success over all stable poses of the object [26, 30]. We formulate Exploratory Grasping as a Markov Decision Process (Section 3.1), define assumptions on the environment (Section 3.2) and formulate the policy learning objective (Section 3.3).

3.1 Exploratory Grasping as an MDP

We consider exploring grasps on an unknown, rigid, polyhedral object $\mathcal{O}$ which rests in one of a finite set of N distinguishable stable poses with associated landing probabilities $\{\lambda_s\}_{s=1}^N$. We consider polyhedral objects since they admit a finite number of stable resting poses, as does any object defined with a triangular mesh [31]. We assume that the robot does not initially know any of these stable poses or their count N. The robot must discover new stable poses from experience by attempting grasps on the object in its current stable pose and lifting and then releasing the object when grasps are successful. We assume an overhead depth camera with known camera intrinsics that cannot reliably determine the 3D shape of the object, but can be used to recognize distinguishable 3D poses by performing planar translations and rotations of the image into a canonical orientation and translation. We also assume that the camera can be used after grasp attempts to determine if a grasp succeeds.

We model Exploratory Grasping as an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R)$ as follows. We first define a one-to-many mapping from the set of object stable poses Σ to the set of overhead point clouds $\mathcal{I}$ that are scale-invariant, translation-invariant, and rotationally-invariant about the vertical axis. Then, we define the state space as the set of *distinguishable* stable poses $\mathcal{S}$, where $\mathcal{S}$ is the set of equivalence classes within Σ, where two poses are equivalent if they map to the same set of overhead point clouds $\mathcal{I}$. For example, all poses of a cube would map to the same overhead point cloud, so a cube only has 1 distinguishable stable pose. We assume point cloud $\mathcal{I}$ is obtained by deprojecting a depth image observation $o \in \mathbb{R}_{H\times W}^{+}$ taken from the overhead camera with known intrinsics. We define a set $\mathcal{A}_s$ of K grasp actions, such as parallel-jaw or suction grasps, for each stable pose $s \in \mathcal{S}$ of the object. The K grasps are sampled from o for each stable pose as in Mahler et al. [17]. The full action space is the union of the grasp actions available on each pose: $\mathcal{A} = \bigcup_{s\in\mathcal{S}} \mathcal{A}_s$. We assume that the robot acts in an environment with unknown transition probability distribution $P(s' \mid s, a)$, which denotes the probability of the object transitioning to pose s' if grasp action a is executed in pose s. If a is a successful grasp, then the object is released onto the workspace to sample a new stable pose s' based on unknown landing probabilities $\{\lambda_s\}_{s=1}^N$, while if a is a failed grasp, the object topples into some new pose s' with unknown probability $\delta_{s,s'}$. The robot receives reward $R(s, a)$ for selecting grasp action a on stable pose s, so $R(s, a) = 1$ if executing a in stable pose s results in the object being successfully grasped and lifted, and 0 otherwise. We assume that rewards are drawn from a Bernoulli distribution with unknown parameter $\phi_{s,a}$: $R(s, a) \sim \text{Ber}(\phi_{s,a})$ for $a \in \mathcal{A}_s$.

3.2 Assumptions

To study Exploratory Grasping, we first establish assumptions on the system dynamics to ensure that all poses are reachable and which describe how the object pose can evolve when the object is (1) released onto the workspace or (2) when a grasp is attempted. We make no assumptions on the grasping modality (e.g., parallel jaw, suction, multi-fingered, etc).

Assumption 3.1. Grasp Dynamics: If a grasp succeeds, we assume that the robot can randomize the pose of the object before releasing it to sample subsequent stable poses from the associated unknown stable pose landing probabilities $\{\lambda_s\}_{s=1}^N$ for $\mathcal{O}$. If a grasp fails, we assume that the object's pose will either remain unchanged or topple into some other pose s' with unknown probability $\delta_{s,s'}$.

Assumption 3.2. Release Dynamics: The categorical distribution over stable poses defined by landing probabilities $\{\lambda_s\}_{s=1}^N$ is stationary and independent of prior actions and poses when $\mathcal{O}$ is released from a fixed height with its orientation randomized as in Goldberg et al. [26].

Assumption 3.3. Irreducibility: We assume that there exists a policy π such that the Markov chain over stable poses induced by executing π in $\mathcal{M}$ is irreducible, and thus can reach all stable poses with nonzero probability for any initialization.

Note that Assumption 3.3 is satisfied if, for all poses $s \in \mathcal{S}$, $\lambda_s > 0$ and there exists a grasp with success probability $\epsilon > 0$. We assume that these conditions hold for analysis, but since object toppling is also possible, note that these conditions are sufficient but not necessary for ensuring irreducibility.

3.3 Learning Objective

The objective is to learn a policy $\pi : \mathcal{S} \to \mathcal{A}$ that maximizes the expected average reward over an infinite time horizon under the state distribution induced by π. Let $\tau = \{(s_t, \pi(s_t))\}_{t=1}^T$ be a trajectory of all states and actions when executing π for time T and let $r(\tau) = \sum_{t=1}^T r(s_t, \pi(s_t))$ be the sum of rewards for all states and actions in τ, and let $p(\tau|\pi)$ be the trajectory distribution induced by policy π. Then the expected average reward obtained from policy π in $\mathcal{M}$ is given as:

$$J(\mathcal{M}, \pi, T) = \frac{1}{T}\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[r(\tau)\right] \tag{1}$$

The objective is to find the policy which maximizes expected reward over an infinite time horizon:

$$\pi^* = \operatorname*{argmax}_{\pi} \lim_{T \to \infty} J(\mathcal{M}, \pi, T) \tag{2}$$

4 Reinforcement Learning for Exploratory Grasping

We first consider the performance of reinforcement learning algorithms for Exploratory Grasping and leverage the structure of the MDP described in Section 3 to establish a bound on the cumulative regret for a family of existing tabular reinforcement learning algorithms when applied to Exploratory Grasping. See Section A of the supplementary material for all proofs.

4.1 Analyzing the Exploratory Grasping MDP

A common metric to measure policy performance in online-learning settings is *regret*, which has been analyzed in the reinforcement learning setting by a variety of prior work [32, 33, 34, 35]. Intuitively, regret quantifies the difference in accumulated reward within T timesteps between a given policy π and optimal infinite horizon policy π^* for MDP $\mathcal{M}$. More precisely, we define average regret based on the definition in [32]:

$$\text{Regret}(\mathcal{M}, \pi, T) = \max_{\pi'}\left[\lim_{T \to \infty} J(\mathcal{M}, \pi', T)\right] - J(\mathcal{M}, \pi, T) \tag{3}$$

Recent theoretical work by Jaksh et al. [32] on reinforcement learning for tabular MDPs yielded algorithms which can attain average regret proportional to the *diameter* of the MDP, a measure of the furthest distance between pairs of states under an appropriate policy. However, for general MDPs, this diameter can be arbitrarily large, making these regret bounds difficult to interpret in practical settings. We leverage the structure of the MDP to derive an upper bound on the MDP diameter, which precisely quantifies the difficulty of grasp exploration based on the parameters of $\mathcal{M}$.

We begin by defining the Markov chain over $\mathcal{S}$ induced by a stationary deterministic policy π.

Definition 4.1. Pose Evolution under π: Given stationary policy π, the transitions between pairs of states in $\mathcal{M}$ is defined by a Markov chain. Precisely, the transition probabilities under π, denoted by P^π where $P^\pi[s, s'] = P(s' \mid s, a = \pi(s))$, are given as follows:

$$P^\pi[s, s'] = \phi_{s,\pi(s)}\lambda_{s'} + (1 - \phi_{s,\pi(s)})\delta_{s,s'} \tag{4}$$

Given this Markov Chain over poses for π, we can now analyze the diameter of the MDP, denoted $D(\mathcal{M})$, by considering the hitting time between stable poses in $\mathcal{M}$ as defined in [36].

Definition 4.2. Let $T^{\pi}_{s\to s'}$ denote the expected hitting time between states s and s' under policy π under the Markov chain defined in Definition 4.1. Then the diameter of $\mathcal{M}$ is defined as follows [32]:

$$D(\mathcal{M}) = \max_{s\neq s'} \min_{\pi} \mathbb{E}\left[T^{\pi}_{s\to s'}\right] \tag{5}$$

Intuitively, $D(\mathcal{M})$ measures the temporal distance between the most distant states in an MDP under the policy that minimizes this distance. We now leverage the structure of $\mathcal{M}$ to establish an upper bound on $D(\mathcal{M})$.

Lemma 4.1. *The diameter of $\mathcal{M}$ can be bounded above as follows:*

$$D(\mathcal{M}) \leq \frac{1}{\epsilon\lambda_1}, \tag{6}$$

where ϵ is a lower bound on the success probability of the highest quality grasp over all stable poses and λ_1 is the landing probability for the least likely stable pose.

Lemma 4.1 captures the intuition that the diameter of the MDP is large if the best grasp in each stable pose has a low success probability (ϵ is small), or if there exists a stable pose with very low landing probability (λ_1 is small).

Now we can establish regret bounds for 4 tabular reinforcement learning algorithms (UCRL2 [32], KL-UCRL [34], PSRL [37], and UCRLV [33]) when applied to $\mathcal{M}$ by combining diameter dependent regret bounds from prior work and the bound on the diameter of $\mathcal{M}$ established in Lemma 4.1.

Theorem 4.1. *UCRL2 [32], KL-UCRL [34] and PSRL [37] achieve average regret given by Regret$(\mathcal{M}, \pi, T) \sim \tilde{O}\left(\frac{N}{\epsilon\lambda_1}\sqrt{\frac{K}{T}}\right)$ for any Exploratory Grasping MDP $\mathcal{M}$ while UCRLV [33] achieves average regret given by Regret$(\mathcal{M}, \pi, T) \sim \tilde{O}\left(\sqrt{\frac{NK}{\epsilon\lambda_1 T}}\right)$. Here N, K, T, ϵ and λ_1 are defined as in Section 3.*

Theorem 4.1 leverages the specific structure of the MDP in Exploratory Grasping to relate the accumulated regret of 4 tabular RL algorithms to intuitive parameters of the MDP, providing insight into the theoretical complexity of Exploratory Grasping in the context of reinforcement learning.

5 Bandits for Online Rapid Grasp Exploration Strategy (BORGES)

While the reinforcement learning algorithms in Section 4 provide a method and formal guarantees for Exploratory Grasping, the algorithms themselves do not exploit any specific structure in MDP $\mathcal{M}$. One key feature of Exploratory Grasping is that in practice, objects have a small, finite set of stable poses [26]. This motivates learning a set of N policies, each of which explore grasps in a particular object stable pose. However, the grasp exploration problem in each pose is coupled. For example, there may exist a pose s with no available high quality grasps but a high likelihood of a failed grasp causing the object to topple into another pose with high quality grasps. Then, the optimal policy may deliberately fail to grasp the object in pose s in order to obtain access to grasps in the more favorable stable pose, leading it to avoid grasp exploration in poses without high quality grasps. To avoid this behavior, we introduce Assumption 5.1.

Assumption 5.1. We assume that $\delta_{s,s'} \leq \epsilon\lambda_{s'}$ for all $s \neq s'$ where $\delta_{s,s'}$ is the probability of toppling into pose s' given a failed grasp in pose s, ϵ is a lower bound on the success probability of the highest quality grasp over all stable poses as defined above, and $\lambda_{s'}$ is the landing probability of pose s'.

Assumption 5.1 ensures that there exists a grasp in all stable poses s such that the probability of transitioning to new pose s' via a grasp attempt is higher than that of toppling from pose s to pose s'. Given this assumption, the optimal grasp exploration policy in $\mathcal{M}$ reduces to selecting the grasp with highest success probability in each encountered pose, as this policy maximizes both reward at the current timestep and exploration of other stable poses when the object is released. In other words, the global optimal policy is the *greedy* policy. Given this structure, we can view the grasp exploration problem as N independent multi-armed bandit problems corresponding to grasp exploration in each pose. However, although grasp exploration can be performed independently in each pose, the success of a grasp exploration policy in one pose affects the time available to explore grasps in another pose, thereby coupling each bandit problem.

We propose an algorithm that takes advantage of this structure to enable rapid online grasp exploration: Bandits for Online Rapid Grasp Exploration Strategy (BORGES)[1]. BORGES works by maintaining the parameters of N independent bandit policies $(\pi_s^{\mathcal{B}})_{s=1}^N$, where $\pi_s^{\mathcal{B}} : s \to \mathcal{A}_s$ and $\pi_s^{\mathcal{B}}$ is only *active* in pose s. Let $\pi^{\mathcal{B}}$ denote the meta-policy induced by executing $\pi_s^{\mathcal{B}}$ in pose s and assume that $\pi_s^{\mathcal{B}}$ is learned by running a no-regret online learning algorithm $\mathcal{B}$ for grasp exploration in s. No-regret algorithms for the stochastic multi-armed bandit problem include the UCB-1 algorithm [38] and Thompson Sampling [39]. We then formulate a new notion of regret capturing the gap between $\pi^{\mathcal{B}}$ and the optimal policy on their respective distributions and show that BORGES achieves vanishing average regret despite the interdependence between pose exploration times.

Let $p_T^{\mathcal{B}}$ denote the distribution of poses seen under the sequence of policies $\pi_{1:T}^{\mathcal{B}}$ at each round of learning up to time T and let p_T^* denote the distribution of poses seen when executing the optimal policy (π^*) in $\mathcal{M}$ up to time T. We define the average regret accrued by BORGES in MDP $\mathcal{M}$ after T rounds as the difference in accumulated reward of the optimal policy on pose distribution p_T^* and the accumulated reward of $\pi^{\mathcal{B}}$ on pose distribution $p_T^{\mathcal{B}}$.

Definition 5.1. The average regret accumulated by running $\mathcal{B}$ in each stable pose is defined as the difference between the average regret on each pose visited by the optimal policy π^* weighted by the probability of it visiting each pose and the corresponding quantity for the executed policy $\pi^{\mathcal{B}}$:

$$\mathbb{E}\left[\mathcal{R}^{\mathcal{B}}(T)\right] = \sum_{s=1}^{N} p_T^*(s)\mathbb{E}\left[\frac{1}{T_s^*}\sum_{t=1}^{T_s^*} R(s, \pi^*(s))\right] - \sum_{s=1}^{N} p_T^{\mathcal{B}}(s)\mathbb{E}\left[\frac{1}{T_s^{\mathcal{B}}}\sum_{t=1}^{T_s^{\mathcal{B}}} R(s, \pi_t^{\mathcal{B}}(s))\right]$$

where T_s^* is the time spent by the optimal policy in pose s and $T_s^{\mathcal{B}}$ is the time spent by $\pi^{\mathcal{B}}$ in pose s.

In Section A of the supplementary material, we show that the average regret as defined in Definition 5.1 vanishes to 0 in the limit as $T \to \infty$ as stated in Theorem 5.1.

Theorem 5.1. *The average regret accrued by BORGES, when using any no-regret bandit algorithm $\mathcal{B}$ for grasp exploration in each encountered stable pose, vanishes in the limit:*

$$\lim_{T\to\infty} \mathbb{E}\left[\mathcal{R}^{\mathcal{B}}(T)\right] = 0$$

This result leverages the precise structure of $\mathcal{M}$, namely that the optimal policy is the greedy policy, to provide sublinear regret guarantees for BORGES.

A convenient property of BORGES is that while exploring grasps on an object, it naturally explores different object poses as well. Theorem 5.2 establishes an upper bound on the expected number of timesteps required for BORGES to reach all object stable poses.

Theorem 5.2. *Let T_{cover} denote the number of grasps BORGES executes until it has reached every stable pose. We can bound $\mathbb{E}[T_{cover}]$ above as follows where K, ϵ, and λ are defined as in Section 3.*

$$\mathbb{E}\left[T_{cover}\right] \leq \frac{K}{\epsilon}\sum_{j=1}^{N}\frac{1}{1-\sum_{i=2}^{j}\lambda_{i-1}}$$

We note that there can be pathological objects for which Assumption 3.3 is violated due to a sink state from which the object can neither be grasped nor toppled into a new pose. In this case, no grasp exploration strategy will be successful, because there will exist poses from which the robot cannot recover. However, one benefit of BORGES is that it can detect these cases, as it maintains estimates of the success probabilities of each grasp during learning and thus could request external intervention from a human supervisor when it becomes clear that a given pose is a sink state in $\mathcal{M}$. We also acknowledge that many objects violate Assumption 5.1 as they can be easily toppled between poses, including several of the objects we use for evaluation. However, we find that even when Assumption 5.1 is violated, BORGES still achieves good performance in practice (Figure 1).

[1] Jorge Luis Borges (1899-1986) was a brilliant writer whose short stories considered geometry, time, and combinatorics.

6 Experiments

6.1 Simulation Experiments

In simulation experiments, we study whether BORGES can discover high quality grasps on objects for which general-purpose grasping policies, such as Dex-Net 4.0 [4] and GG-CNN [1], perform poorly and whether learning different policies for each stable pose accelerates grasp exploration.

Simulation Details: To evaluate each policy, we choose 7 objects from the set of Dex-Net 2.0 adversarial objects [17] and 39 evaluation objects from the EGAD! dataset [40] for which general-purpose parallel-jaw grasping policies (Dex-Net 4.0 [4] and GG-CNN [1]) perform poorly, despite there existing high-quality grasps in multiple stable poses. We sample a set of K parallel-jaw grasps on the image observation of pose s of each object as in [17], and calculate the ground-truth quality of each grasp, $\phi_{s,a}$, using a wrench resistance metric that measures the ability of the grasp to resist gravity [41]. We randomize the initial pose of each object and execute BORGES and baselines, sampling rewards from $\text{Ber}(\phi_{s,a})$. If the grasp succeeds, we randomize the pose, choosing a stable pose according to the stable pose distribution of the object. Otherwise, we assume that the object remains in the same stable pose. We execute each policy for 10 rollouts of 10 trials, where new grasps are sampled for each trial and each rollout evaluates the policy over 10,000 timesteps of grasp exploration. Since there is stochasticity in both the grasp sampling and the policies themselves, we average policy performance across the 10 rollouts and 10 trials. In addition, we smooth policy performance across a sliding window of 20 timesteps and report average reward for each timestep.

To model object toppling when grasps are unsuccessful (Section 3.2), we use the toppling analysis from Correa et al. [28] to determine the toppling transition matrix for a given object. Specifically, we generate the distribution of next states from a given state by sampling non-colliding pushes across vertices on the object, finding their distribution of next states given perturbations around the nominal push point, and average the distribution from all of the pushes. Then if a grasp fails, we choose the next state according to the corresponding topple transition probabilities. When evaluating policies, we remove poses that have no grasps with nonzero ground-truth quality and renormalize the stable pose distribution. We emphasize that we perform this step for all policies and do this only to ensure that no poses act as sink states from which the robot can never change the pose in the future (and thus ensure that Assumption 3.3 is satisfied). For the 7 objects from the Dex-Net 2.0 adversarial object dataset, 11% of poses (an average of 1.3 poses per object) were removed. We present additional experiments in the supplementary material where this pose removal step is not performed, and find that while all policies perform more poorly (Section C), BORGES still outperforms all baseline policies. We also include further details on the experimental setup in the supplementary material.

Policies: We compare 3 variants of BORGES against 3 baselines to evaluate whether BORGES is able to (1) substantially outperform general-purpose grasping policies [4] on challenging objects and (2) learn more efficiently than other online learning algorithms which update a grasp quality convolutional network (GQCNN) online or explore grasps via reinforcement learning. We also instantiate BORGES with different algorithms for $\mathcal{B}$ to study how this choice affects grasp exploration efficiency and implement an oracle baseline to study whether BORGES is able to converge to the optimal policy. We compare the following baselines and Bandit Grasping variants: **Dex-Net**, which selects grasps greedily with respect to the Grasp Quality Convolutional Network from Dex-Net 4.0 [4] with probability 0.95 and selects a grasp uniformly at random with probability 0.05, **Dex-Net-FT**, which additionally fine tunes the GQCNN policy online from agent experience, an implementation of **UCRL2** from [42], a tabular RL algorithm discussed in Section 4, and instantiations of BORGES with the UCB-1 algorithm [38] (**BORGES-UCB**), Thompson sampling (**BORGES-TS**) with a uniform Beta distribution prior, and Thompson sampling with a prior from the Dex-Net policy of strength 5 (**BORGES-TS5**) as in [22]. Finally, we implement an oracle baseline that chooses grasps with the best ground-truth metric at each timestep to establish an upper bound on performance.

Simulation Results: Results for baselines and BORGES variants are in Figure 1. Above the learning curves in Figure 1, we report the maximum violation of Assumption 5.1 across all pose pairs for each object ($\max_{s,s'}\left[\delta_{s,s'} - \epsilon\lambda_{s'}\right]$). This value captures how much more likely an object is to topple between poses s and s' than be successfully grasped in s and released into s' for pose pair (s, s') which most violates Assumption 5.1. BORGES is robust to large violations of Assumption 5.1, and substantially outperforms prior methods even when there exist pose pairs where it is 30% more likely to topple between the poses than transition between them with a successful grasp.

As shown in Figure 1, the Dex-Net policy typically performs very poorly, achieving an average reward of less than 0.1 per timestep. While the online learning policies also start poorly, they quickly improve, and BORGES-TS eventually converges to the optimal policy. We also find that BORGES-TS5, which leverages Dex-Net as a prior using the method presented in [22], further speeds convergence to the optimal policy. This result is promising, as it suggests that the exploration strategy in BORGES can be flexibly combined with general-purpose grasping policies to significantly accelerate grasp exploration on unknown objects. We find that the Dex-Net-FT policy performs very poorly even though it continues to update the weights of the network online with the results of each grasp attempt and samples a random grasp with probability 0.05, which aids exploration. We hypothesize that this is due to the initial poor performance of Dex-Net—since the vast majority of fine-tuning grasps attain zero reward, the network is unable to explore enough high-quality grasps on the object. Overall, these results suggest that BORGES can greatly increase grasp success rates on objects for which Dex-Net performs poorly. We also find that BORGES policies greatly outperform tabular RL policy UCRL2 by leveraging the structure of the MDP. Both the UCB and Thompson sampling implementations maintain separate policies in each pose, leveraging the fact that the optimal policy is the greedy policy. This allows the BORGES policies to not waste timesteps exploring possible transitions to poses with low rewards. Thompson sampling additionally leverages the fact that the rewards are distributed as Bernoulli random variables, which may explain the significant performance gap between the Thompson sampling and UCB implementations. Thus, BORGES policies quickly learn to choose high-quality grasps in each pose to transition quickly to new poses.

We perform sensitivity analysis of BORGES to ϵ and λ_1 and find that BORGES quickly converges to the optimal policy unless ϵ or λ_1 is low. We additionally evaluate BORGES for different numbers of sampled grasps (K) on each pose, and find that with decreasing values of K, all policies perform worse since the likelihood of sampling a high quality grasp decreases, but BORGES policies continue to outperform prior methods. See Section D of the supplementary material for more details.

6.2 Initial Physical Experiments

In physical experiments, we evaluate BORGES on two challenging objects on an ABB YuMi robot and compare performance with the Dex-Net policy from Section 6.1. Section E of the supplementary material contains further details on experimental setup. Figure 5 shows learning curves for both BORGES and Dex-Net; results suggest that BORGES quickly finds high-quality grasps across all object poses, converging to near-perfect performance. Dex-Net correctly predicts high-quality grasps on some of the poses, but attempts suboptimal grasps in others, resulting in high variance performance across the rollout since it is unable to adapt to its successes and failures. Across the final 100 timesteps for each object (i.e., after BORGES begins to explore all poses), BORGES reaches 0.89 and 0.87 success rates on the clamp and pipe, respectively, as compared to 0.49 and 0.37 for the Dex-Net policy. This effect highlights the importance of a policy that is able to learn online from successful and failed grasp attempts; Dex-Net does not learn online so continuing to attempt grasps will lead to the same high-variance behavior over time, but BORGES continues to stabilize as it approaches optimal performance across all stable poses.

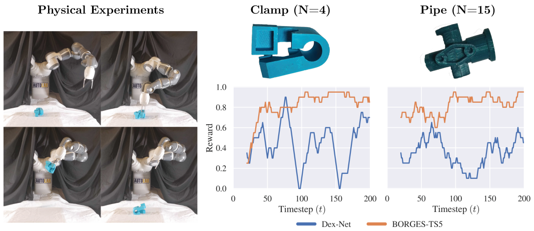

Figure 2: Experiment setup and learning curves for the Dex-Net and BORGES-TS5 policies for two objects across 200 grasp attempts (smoothed with a running average of 20 attempts). The robot attempts to grasp the object at each timestep and, if it succeeds, rotates and drops the object to sample from the stable pose distribution (left). BORGES-TS5 quickly converges within 100 attempts on both objects, indicating that it finds grasps that succeed nearly every time for each pose. Dex-Net's performance remains uneven, indicating that it finds high-quality grasps for some poses, but not others.

7 Future Work

In future work, we plan to evaluate BORGES in extensive physical trials with different grasping modalities such as suction grasps and with priors from different grasp planners. One property of BORGES is that in the process of exploring grasps, it also explores different object stable poses. Thus, we are excited to explore further applications of BORGES; for example, exploring different object poses to construct accurate 3D models of unknown objects or inspect parts for defects.

Acknowledgments

This research was performed at the AUTOLAB at UC Berkeley in affiliation with the Berkeley AI Research (BAIR) Lab, and the CITRIS ”People and Robots” (CPAR) Initiative. This research was supported in part by the Scalable Collaborative Human-Robot Learning (SCHooL) Project, NSF National Robotics Initiative Award 1734633, by donations from Google, Toyota Research Institute, Siemens, Autodesk, Honda, Intel, Hewlett-Packard and by equipment grants from PhotoNeo, NVidia, and Intuitive Surgical. Ashwin Balakrishna and Michael Danielczuk are supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. 1752814. We thank our colleagues who provided helpful feedback, especially Brijen Thananjeyan and Jeffrey Ichnowski.

References

[1] D. Morrison, P. Corke, and J. Leitner. Learning robust, real-time, reactive robotic grasping. *Int. Journal of Robotics Research (IJRR)*, 39(2-3):183–201, 2020.

[2] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conf. on Robot Learning (CoRL)*, 2018.

[3] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. Journal of Robotics Research (IJRR)*, 37 (4-5):421–436, 2018.

[4] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg. Learning ambidextrous robot grasping policies. *Science Robotics*, 4(26):4984, 2019.

[5] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, volume 1, pages 348–353, 2000.

[6] R. M. Murray. *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[7] E. Rimon and J. Burdick. *The Mechanics of Robot Grasping*. Cambridge University Press, 2019.

[8] J. Kim, K. Iwamoto, J. J. Kuffner, Y. Ota, and N. S. Pollard. Physically based grasp quality evaluation under pose uncertainty. *IEEE Trans. Robotics*, 29(6):1424–1439, 2013.

[9] D. Kappler, J. Bohg, and S. Schaal. Leveraging big data for grasp planning. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 4304–4311, 2015.

[10] U. Viereck, A. ten Pas, K. Saenko, and R. Platt. Learning a visuomotor controller for real world robotic grasping using simulated depth images. *arXiv preprint arXiv:1706.04652*, 2017.

[11] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 3406–3413, 2016.

[12] C. Choi, W. Schwarting, J. DelPreto, and D. Rus. Learning object grasping for soft robot hands. *IEEE Robotics & Automation Letters*, 3(3):2370–2377, 2018.

[13] M. Breyer, F. Furrer, T. Novkovic, R. Siegwart, and J. Nieto. Comparing task simplifications to learn closed-loop object picking using deep reinforcement learning. *IEEE Robotics & Automation Letters*, 4(2): 1549–1556, 2019.

[14] O. Kroemer, S. Niekum, and G. Konidaris. A review of robot learning for manipulation: Challenges, representations, and algorithms. *arXiv preprint arXiv:1907.03146*, 2019.

[15] D. Wang, D. Tseng, P. Li, Y. Jiang, M. Guo, M. Danielczuk, J. Mahler, J. Ichnowski, and K. Goldberg. Adversarial grasp objects. In *Proc. IEEE Conf. on Automation Science and Engineering (CASE)*, pages 241–248, 2019.

[16] K. Sanders, M. Danielczuk, J. Mahler, A. Tanwani, and K. Goldberg. Non-markov policies to reduce sequential failures in robot bin picking. In *Proc. IEEE Conf. on Automation Science and Engineering (CASE)*, 2020.

[17] 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. In *Proc. Robotics: Science and Systems (RSS)*, 2018.

[18] I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *Int. Journal of Robotics Research (IJRR)*, 34(4-5):705–724, 2015.

[19] M. Laskey, Z. McCarthy, J. Mahler, F. T. Pokorny, S. Patil, J. Van Den Berg, D. Kragic, P. Abbeel, and K. Goldberg. Budgeted multi-armed bandit models for sample-based grasp planning in the presence of uncertainty. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2015.

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

[21] J. Oberlin and S. Tellex. Autonomously acquiring instance-based object models from experience. In *Int. S. Robotics Research (ISRR)*, pages 73–90. Springer, 2015.

[22] K. Li, M. Danielczuk, A. Balakrishna, V. Satish, and K. Goldberg. Accelerating grasp exploration by leveraging learned priors. In *Proc. IEEE Conf. on Automation Science and Engineering (CASE)*, 2020.

[23] O. B. Kroemer, R. Detry, J. Piater, and J. Peters. Combining active learning and reactive control for robot grasping. *Robotics and Autonomous systems*, 58(9):1105–1116, 2010.

[24] C. Eppner and O. Brock. Visual detection of opportunities to exploit contact in grasping using contextual multi-armed bandits. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[25] Q. Lu, M. Van der Merwe, and T. Hermans. Multi-fingered active grasp learning. *arXiv preprint arXiv:2006.05264*, 2020.

[26] K. Goldberg, B. V. Mirtich, Y. Zhuang, J. Craig, B. R. Carlisle, and J. Canny. Part pose statistics: Estimators and experiments. *IEEE Trans. Robotics and Automation*, 15(5):849–857, 1999.

[27] K. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 1993.

[28] C. Correa, J. Mahler, M. Danielczuk, and K. Goldberg. Robust toppling for vacuum suction grasping. In *Proc. IEEE Conf. on Automation Science and Engineering (CASE)*, 2019.

[29] K. Y. Goldberg and M. T. Mason. Bayesian grasping. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1264–1269, 1990.

[30] M. Moll and M. A. Erdmann. Manipulation of pose distributions. *Int. Journal of Robotics Research (IJRR)*, 21(3):277–292, 2002.

[31] A. S. Rao and K. Y. Goldberg. Manipulating algebraic parts in the plane. *IEEE Transactions on Robotics and Automation*, 11(4):598–602, 1995.

[32] T. Jaksh, R. Ortner, and P. Auer. Near-optimal regret bounds for reinforcement learning. In *Journal of Machine Learning Research*, 2010.

[33] A. Tossou, D. Basu, and C. Dimitrakakis. Near-optimal optimistic reinforcement learning using empirical bernstein inequalities. In *Proceedings of Machine Learning Research*, 2019.

[34] S. Filippi, O. Cappé, and A. Garivier. Optimism in reinforcement learning and kullback-leibler divergence. In *Annual Allerton Conference on Communication, Control, and Computing*, 2010.

[35] I. Osband and B. Van Roy. Near-optimal reinforcement learning in factored mdps. In *Proc. Advances in Neural Information Processing Systems*, 2014.

[36] H. Chen and F. Zhang. The expected hitting times for finite markov chains. *Linear Algebra and its Applications*, 428(11-12):2730–2749, 2008.

[37] Z. Xu and A. Tewari. Near-optimal reinforcement learning in factored mdps: Oracle-efficient algorithms for the non-episodic setting. In *Proc. Int. Conf. on Machine Learning*, 2018.

[38] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[39] S. Agrawal and N. Goyal. Further optimal regret bounds for thompson sampling. In *Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, 2013.

[40] D. Morrison, P. Corke, and J. Leitner. Egad! an evolved grasping analysis dataset for diversity and reproducibility in robotic manipulation. *IEEE Robotics & Automation Letters*, 2020.

[41] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg. Dex-net 3.0: Computing robust robot vacuum suction grasp targets in point clouds using a new analytic model and deep learning. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018.

[42] R. Fruit. Exploration-exploitation in reinforcement learning. `https://github.com/RonanFR/UCRL`, 2018.

Exploratory Grasping: Asymptotically Optimal Algorithms for Grasping Challenging Polyhedral Objects Supplementary Material

The supplementary material is organized as follows. In Section A we provide proofs for all theoretical results in Sections 4 and 5. In Section B we provide additional experimental details, in Section C we include additional simulation experiments, and in Section D we perform sensitivity experiments. In Section 6.2 we provide further details on physical experiments.

A Proofs

Here we provide the proofs for all results in Section 4 and Section 5 in the main text.

A.1 Proof of Lemma 4.1

Consider Exploratory Grasping MDP $\mathcal{M}'$ for which the object stable pose does not change when a grasp fails. Thus, it must be the case that $\delta_{l,m} = 0$ when $l \neq m$ and $\delta_{l,l} = 1\ \forall l$. Note that for any Exploratory Grasping MDP $\mathcal{M}$, it must be the case that $D(\mathcal{M}) \leq D(\mathcal{M}')$ since the probability of transition between any pair of distinct states in $\mathcal{M}'$ is at most the probability of transition in $\mathcal{M}$. Now we establish a bound on $D(\mathcal{M})$ by bounding $D(\mathcal{M}')$.

Without loss of generality, let $\lambda_1 \leq \lambda_s\ \forall s \in \mathcal{S}$. Additionally, define π^* as the policy which selects the grasp with highest success probability on all poses, with associated probability transition matrix P^{π^*} and with hitting time $T^{\pi^*}_{s\to s'}$ defined as in Definition 4.1. Then, the diameter of $\mathcal{M}'$ can be computed as follows.

For MDP $\mathcal{M}'$, it must be the case that

$$\min_{\pi} T^{\pi}_{s\to s'} = T^{\pi^*}_{s\to s'} \tag{7}$$

since π^*, the policy which always picks the highest quality grasp on each pose, minimizes the hitting time between any pair of poses s, s'. Furthermore, note that

$$\max_{s\neq s'} T^{\pi^*}_{s\to s'} = \max_{s} T^{\pi^*}_{s\to 1} \tag{8}$$

since for any starting pose s, the hitting time between s and s' will always be highest for $s' = 1$ (the pose with lowest drop probability) for any policy π. Thus, we see that

$$D(\mathcal{M}') = \max_{s} T^{\pi^*}_{s\to 1} \tag{9}$$

Finally, we leverage equation 9 to compute an upper bound on $D(\mathcal{M}')$ as follows.

Let π_ϵ be any policy which selects a grasp with success probability ϵ on each stable pose l. Note that by Assumption 3.3 we assume that there exists a grasp with success probability at least ϵ on each pose. Without loss of generality, we can consider the case that there exists a grasp with success probability exactly ϵ on each pose and the policy which selects these grasps since the hitting times under this policy will only be greater than those under a policy which selects grasps with success probability greater than ϵ. Then, it follows that

$$\min_{\pi} T^{\pi}_{s\to s'} \leq T^{\pi_\epsilon}_{s\to s'}\ \forall s, s' \tag{10}$$

Now note that since we are considering pose evolution under π_ϵ, which selects a grasp of the same quality ϵ on any pose, the starting pose s does not affect the hitting time. Combining this with the fact that the hitting time to the least likely pose (pose 1) will always be the highest for any policy π and inequality (10) yields that

$$D(\mathcal{M}') \leq \max_{s\neq s'} T^{\pi_\epsilon}_{s\to s'} = T^{\pi_\epsilon}_{2\to 1} \tag{11}$$

Since the choice of s does not matter under π_ϵ, we use $s = 2$ above without loss of generality. Now, note that the hitting time to pose 1 under π_ϵ is distributed as a geometric random variable with parameter $\epsilon\lambda_1$, which has mean $\frac{1}{\epsilon\lambda_1}$, yielding the desired result. □

A.2 Proof of Theorem 4.1

The result immediately follows from combining the diameter bound from Lemma 4.1 and the regret bounds established for UCRL2 [32], KL-UCRL [34] and PSRL [37] (average regret $\tilde{O}(DS\sqrt{A/T})$) and for UCRLV [33] (average regret $\tilde{O}(\sqrt{DSA/T})$) where D is the MDP diameter and S and A are the cardinalities of the state space and action space respectively. □

A.3 Proof of Theorem 5.1

We bound the expected regret of BORGES by decomposing the regret into two terms, one which depends on the divergence between the distribution of poses seen by the optimal policy and $\pi^{\mathcal{B}}$, and the other of which depends on the difference in rewards attained by $\pi^{\mathcal{B}}$ and π^* when evaluated on the distribution of poses seen by $\pi^{\mathcal{B}}$. For simplicity, we refer to $\pi^{\mathcal{B}}$ as π for the proof.

$$\mathbb{E}\left[\mathcal{R}^{\mathcal{B}}(T)\right] = \sum_{s=1}^{S} p_T^*(s)\mathbb{E}\left[\frac{1}{T_s^*}\sum_{t=1}^{T_s^*} R(s,\pi^*(s))\right] - \sum_{s=1}^{S} p_T^\pi(s)\mathbb{E}\left[\frac{1}{T_s^\pi}\sum_{t=1}^{T_s^\pi} R(s,\pi_t(s))\right] \tag{12}$$

$$= \sum_{s=1}^{S}\left(p_T^*(s) - p_T^\pi(s)\right) g_s^*(T_s^*) + \sum_{s=1}^{S} p_T^\pi(s)\left(g_s^*(T_s^*) - g_s^\pi(T_s^\pi)\right) \tag{13}$$

$$= \mathbb{E}\left[\mathcal{R}_\pi^{\mathcal{B}}(T)\right] + \sum_{s=1}^{S}\left(p_T^*(s) - p_T^\pi(s)\right) g_s^*(T_s^\pi) \tag{14}$$

where (13) follows from letting $g_s^*(T_s^*) = \mathbb{E}\left[\frac{1}{T_s^*}\sum_{t=1}^{T_s^*} R(s,\pi^*(s))\right]$ and $g_s^\pi(T_s^\pi) = \mathbb{E}\left[\frac{1}{T_s^\pi}\sum_{t=1}^{T_s^\pi} R(s,\pi_t(s))\right]$ and (14) follows from letting $\mathbb{E}\left[\mathcal{R}_\pi^{\mathcal{B}}(T)\right]$ denote the expected regret on the distribution of poses visited by policy π and noting that the average reward for the optimal policy, g_s^*, is independent of the timesteps spent in the pose (i.e., $g_s^*(T_s^\pi) = g_s^*(T_s^*)\ \forall s$).

We first focus on the first term in (14). We know that $\mathbb{E}\left[\mathcal{R}_\pi^{\mathcal{B}}(T)\right]$ approaches zero if each pose is visited infinitely often in the limit as $T \to \infty$ provided that $\mathcal{B}$ is a no-regret online learning algorithm:

$$\lim_{T\to\infty} p_T^\pi(s) > 0, \forall s \in \{1, 2, \ldots, S\} \quad \Rightarrow \quad \lim_{T\to\infty} \mathbb{E}\left[\mathcal{R}_\pi^{\mathcal{B}}(T)\right] = 0 \tag{15}$$

Thus, it remains to be shown that under π, all poses are visited infinitely often in the limit as $T \to \infty$. Note that this statement is equivalent to showing that in the limit as $T \to \infty$, bandit algorithm $\mathcal{B}$ selects grasps on each pose with non-zero success probability with non-zero probability. Suppose that this was not the case (i.e., that as $T \to \infty$, $\mathcal{B}$ assigns zero grasp probability to all grasps with non-zero success probability). This would imply that $\mathcal{B}$ only selects grasps with zero success probability, and thus incurs constant regret on its own distribution ($\lim_{T\to\infty} \mathbb{E}\left[\mathcal{R}_\pi^{\mathcal{B}}(T)\right] > 0$). This contradicts the initial assumption that $\mathcal{B}$ is a no-regret online learning algorithm, showing that under π, all poses must be visited infinitely often in the limit as $T \to \infty$.

Now we shift our attention to the second term in (14). Given that $\mathcal{B}$ is a no-regret online learning algorithm, it must be the case that $g_s^\pi(T_s^\pi) \xrightarrow[T_s^\pi\to\infty]{} g_s^*(T_s^\pi)\ \forall s$. This implies that in the limit as $T \to \infty$, π and π^* have the same success rate on all stable poses. Two policies with the same success rate on all stable poses induce the same Markov chain over $\mathcal{S}$, and thus admit the same stable pose distribution. Thus, $p_T^\pi(s) \xrightarrow[T\to\infty]{} p_T^*(s)$, implying that the second term also approaches 0 as $T \to \infty$. □

A.4 Proof of Theorem 5.2

Let X denote the random variable that is the number of drops needed until we've reached every stable pose. We assume we have n distinct stable poses. Let X_k be the discrete random variable that represents the number of drops after visiting the $(k-1)$th distinct stable pose to visit the kth distinct stable pose. As a base case, $X_1 = 1$ since the first pose we drop into will always be distinct. By linearity of expectation we have

$$\mathbb{E}[X] = \sum_{j=1}^{N} \mathbb{E}[X_j]. \tag{16}$$

After we visit the $(j-1)$th distinct stable pose we have $N-j+1$ stable poses left that haven't been visited yet. As before, we order the stable poses in order from most likely to least likely: $\lambda_1 > \lambda_2 > \cdots > \lambda_N$. Because the expected number of drops to get into an unseen pose is a geometric random variable, we can upper bound the number of drops needed to visit all stable poses as

$$\mathbb{E}[X] = \sum_{j=1}^{N} \mathbb{E}[X_j] \leq \sum_{j=1}^{N} \frac{1}{1 - \sum_{i=2}^{j} \lambda_{i-1}}. \tag{17}$$

We derive this by first noting that the probability of visiting a new stable pose is $1 - \sum_{i \in \text{visited}} \lambda_i$ and the expected number of drops to successfully visit a new stable pose is a geometric random variable. We want to bound the number of drops to visit all stable poses, but we do not know what order we will visit these poses. However, the worst-case outcome (largest number of required drops) will be if we visit them in order from largest drop probability to smallest since this will reduce the probability as quickly as possible which will maximize $E[X]$ since the probabilities are in the denominator. The above analysis assumes we are repeatedly able to randomly drop the object, with no accounting for grasp success or failure. Rather than only bounding the number of drops, we want to bound the expected number of grasps to visit all stable poses at least once.

The grasping strategy will impact how often we need to attempt grasps before we get to drop again. For a worst-case analysis, we assume that there are no topples, $\delta_{s,s'} = 0 \ \forall s, s'$, and assume that each stable pose has only one grasp with non-zero success probability. We will also let ϵ be the lower bound on the success probability of this grasp across all stable poses. If we just pick grasps uniformly at random on each pose, we have a $1/K$ probability of selecting the best grasp (ie. the one grasp with non-zero success probability) out of K possible grasps. Note that this strategy can only yield strictly lower grasp success rate than BORGES since any no-regret algorithm (such as UCB or Thompson sampling) will cause BORGES to pick grasps with non-zero success probability with higher probability than grasps which never succeed. Thus, if we can upper bound the number of grasps under a uniform random grasp strategy this will upper bound the number of grasps for BORGES. The lower bound on the probability of a successful grasp which leads to a redrop is ϵ. Thus, in the worst-case, we have probability ϵ/K of getting to redrop and the worst-case expected number of grasps to get a successful grasp is K/ϵ since this is a geometric random variable. Thus, the expected number of grasps we need to visit all stable poses is no more than

$$\frac{K}{\epsilon} \sum_{j=1}^{N} \frac{1}{1 - \sum_{i=2}^{j} \lambda_{i-1}}. \tag{18}$$

□

B Experimental Details

Object Selection: We choose 7 objects from the set of adversarial objects in Dex-Net 2.0 [17] because these objects had empirically been shown to be difficult to grasp for the Dex-Net policy. Similarly, the recently-introduced EGAD! object dataset [40] was created to contain objects with few high-quality parallel-jaw grasps. For this dataset, we select all objects for which there exists at least one sampled grasp of quality $\epsilon = 0.1$ on at least one stable pose of the object. Of the 49 objects in the EGAD evaluation dataset, 39 met this criterion.

Pose Selection: For each of the objects, we remove stable poses from the distribution in simulation if they occur with less than a 0.1% chance or if they do not contain a sampled grasp with quality at least $\epsilon = 0.1$. When a pose is removed, the remaining stable pose distribution is renormalized.

Grasp Sampling: We sample a set of $K = 100$ parallel-jaw grasps on the image observation of each pose of each object as in [17]. This sampling process is done using the depth image grasp sampler from the GQCNN repository and is repeated for up to 10 iterations. At each iteration, the sampled grasps' ground truth qualities are calculated using the robust wrench resistance metric that measures the ability of the grasp to resist gravity [41]. If no grasps are found with quality of at least $\epsilon = 0.1$, then the sampling process is repeated for another iteration where more grasps are sampled. If a grasp is found with quality at least ϵ, then that grasp is selected along with 99 other grasps chosen at random from the sampled grasps. If the maximum number of iterations are exceeded without finding a grasp with quality ϵ, the stable pose is discarded.

C Additional Simulation Experiments

We also evaluate BORGES and baselines in a setting in which toppling is not possible ($\delta_{s,s'} = 0, \forall s \neq s$). This exacerbates the difficulty of grasp exploration since an object must be successfully grasped in a given pose for policies to be able to explore grasps in other poses. When evaluating policies, we remove poses that have no grasps with nonzero ground-truth quality and renormalize the stable pose distribution. We emphasize that we perform this step for all policies and do this only to ensure that no poses act as sink states from which the robot can never change the pose in the future (and thus ensure that Assumption 3.3 is satisfied). For the 7 objects from the Dex-Net 2.0 adversarial object dataset, 11% of poses (an average of 1.3 poses per object) were removed. These poses made up an average of 2.8% of the probability mass of the stable pose distribution. These results are shown in Figure 1. We find that BORGES again significantly outperforms baseline policies in this setting.

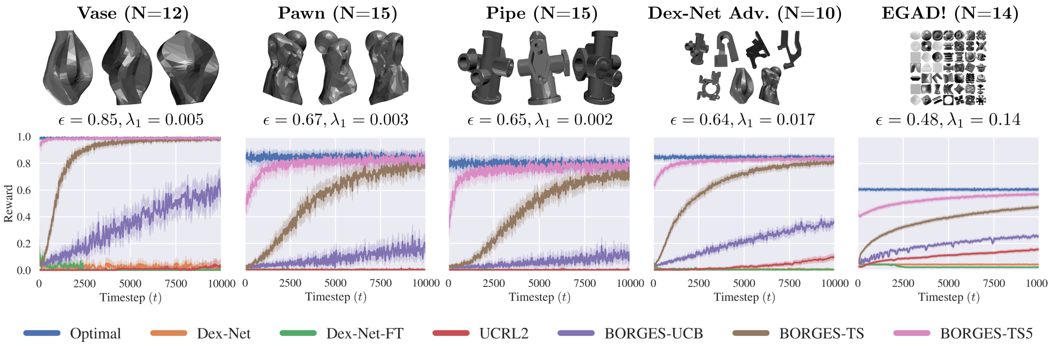

Figure 1: Reward plots for three Dex-Net Adversarial objects as well as aggregated rewards across all Dex-Net Adversarial objects and EGAD! objects in a scenario where no toppling is allowed between poses. Poses without any nonzero quality grasps are discarded. When transitions can only occur due to successful grasps, Exploratory Grasping policies converge to the optimal policy much more quickly than baselines.

We show the importance of Assumption 3.3 even when toppling is included by evaluating policies on all stable poses of the object, regardless of whether a grasp with nonzero ground-truth quality exists or toppling out of each pose is possible. Thus, it is possible that some poses may act as sink states. Figure 2 suggests that for objects that do not obey Assumption 3.3, such as the pawn, all policies find sink states and remain in them. For objects that do obey Assumption 3.3, but have stable poses that can only transition via toppling, such as the mount, performance decreases for all policies, but the order of convergence is consistent.

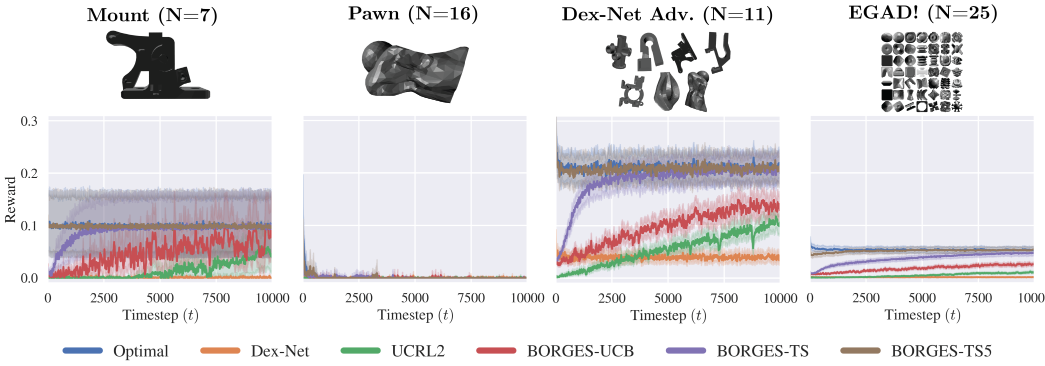

Figure 2: Reward plots for two Dex-Net adversarial objects shown alongside an aggregation of all Dex-Net adversarial objects and EGAD! objects in a scenario where no poses are removed from exploration. In such a case, there is a split between objects without sink states, such as the mount, and objects with sink states, such as the pawn, where no means of transitioning via toppling or grasping is available.

D Sensitivity Experiments

D.1 Sensitivity to Number of Grasp Samples

We vary K, the number of grasp samples per object pose, within the range $K = \{10, 20, 50, 100, 200\}$ across the 7 objects in the Dex-Net Adversarial object set. The results in Figure 3 suggest that as K increases, the likelihood of sampling a high-quality grasp on each pose also increases. With more grasps to consider, each policy takes longer to converge to the optimal policy, but the order of convergence between policies is consistent.

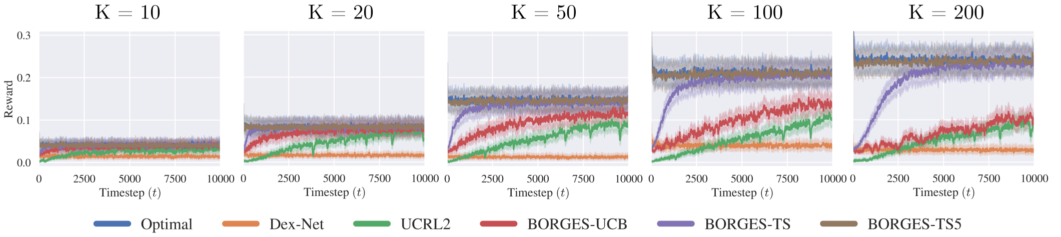

Figure 3: Sensitivity of BORGES to the number of grasp samples K across the 7 Dex-Net Adversarial objects. With more grasp samples, the likelihood of including a higher-quality grasp increases, but the rate of convergence decreases.

D.2 Sensitivity to Exploratory Grasping Parameters

We perform sensitivity analysis of BORGES to the Exploratory Grasping parameters ϵ and λ_1. For these experiments, we evaluate the policy using a set of synthetic objects with $\lambda_1 = \{0.001, 0.01, 0.1, 0.2\}$ and $\epsilon = \{0.1, 0.25, 0.5, 0.75, 1.0\}$ and for simplicity consider the case in which toppling is not possible. In each case, we choose a single grasp on each pose to have reliability ϵ with all other grasps having a mean parameter of 0. The results are shown in Figure 4. These results suggest that unless ϵ or λ_1 are low, BORGES quickly converges to the optimal policy. In particular, ϵ has an outsized effect on the accumulated reward; with $\epsilon \leq 0.10$, we observe that the policy fails to approach the optimal policy regardless of λ_1 through the first 10,000 timesteps.

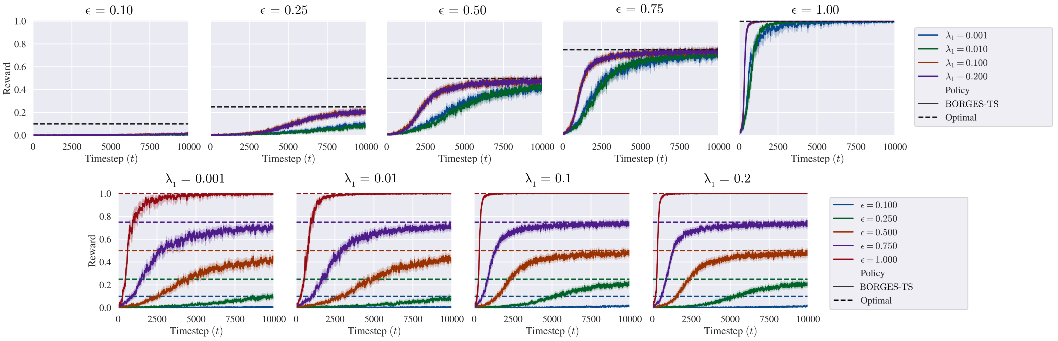

Figure 4: Sensitivity of BORGES to the Exploratory Grasping parameters ϵ and λ_1, as shown across 20 synthetic objects. Unless ϵ or λ is low, BORGES quickly converges to the optimal policy. However, when either is low, particularly ϵ, the policy converges much more slowly, taking even more than the 10,000 timesteps shown for some combinations.

E Initial Physical Experiments Details

In physical experiments, we evaluate BORGES on two challenging 3D printed objects and compare performance with the Dex-Net policy from Section 6.1. The object is placed in front of an ABB Yumi robot with a parallel-jaw gripper and overhead Photoneo PhoXi S camera. At each timestep, we capture a depth image observation from the camera and estimate if the current object stable pose has been seen previously. Specifically, we cache the first depth image seen for each new pose, and compare the current depth image at each timestep to the cached depth images. We segment out the background and project the depth image into a point cloud representation. We then mean-center the current point cloud, apply 7200 rotations around the z-axis, and measure the chamfer distance

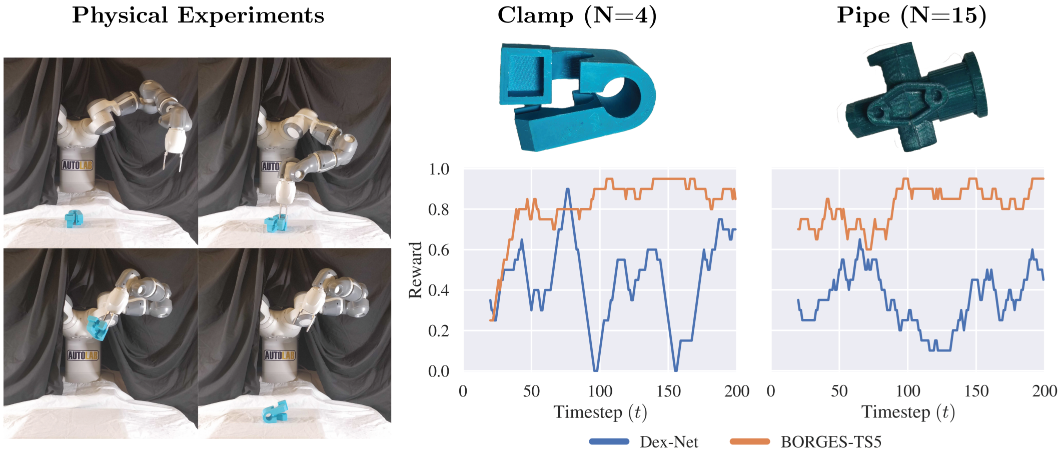

Figure 5: Experiment setup and learning curves for the Dex-Net and BORGES-TS5 policies for the clamp and pipe objects across 200 grasp attempts (smoothed with a running average of 20 attempts). The robot attempts to grasp the object at each timestep and, if it succeeds, rotates and drops the object to sample from the stable pose distribution (left). BORGES-TS5 quickly converges within 100 attempts on both objects, indicating that it finds grasps that succeed nearly every time for each pose. Dex-Net's performance remains uneven, indicating that it finds high-quality grasps for some poses, but not others.

between each resulting rotated point cloud and the point cloud representations of each of the cached depth images. We classify each point in the current point cloud as an inlier if the closest point in a previous point cloud is less than 0.02 mm away. If at least 80% of the points are inliers, we classify the two point clouds as belonging to the same stable pose. If the pose has not been seen previously, we add the depth image to the cache, sample a set of grasps, and calculate the Dex-Net predicted quality values for each pose. Otherwise, we recall the previously sampled grasps for the given pose and transform them by the relative transform between the cached and current images. A grasp is executed according to the Dex-Net or BORGES policy; if the grasp is successful, the object is lifted, rotated by 30 degrees around an axis sampled uniformly at random from 3D unit vectors, and released. If the grasp is unsuccessful, the object may topple into another pose or may remain in the same pose. This process repeats for 200 timesteps per object.

Figure 5 shows learning curves comparing BORGES and the Dex-Net policy on the two objects and suggests that BORGES quickly finds high-quality grasps across all poses of the objects, converging to near-perfect performance. Dex-Net correctly predicts high-quality grasps on some of the poses, but attempts suboptimal grasps in others, resulting in high variance performance across the rollout since it is unable to adapt to its successes and failures. Across the final 100 timesteps for each object (i.e., after BORGES begins to explore all poses), BORGES reaches 0.89 and 0.87 success rates on the clamp and pipe, respectively, as compared to 0.49 and 0.37 for the Dex-Net policy. This effect highlights the importance of a policy that is able to learn online from successful and failed grasp attempts; Dex-Net does not learn online so continuing to attempt grasps will lead to the same high-variance behavior over time, but BORGES continues to stabilize as it approaches optimal performance across all stable poses.