Energy-Bounded Caging: Formal Definition and 2D Energy Lower Bound Algorithm Based on Weighted Alpha Shapes

Jeffrey Mahler[1], Florian T. Pokorny[1], Zoe McCarthy[1], A. Frank van der Stappen[3], Ken Goldberg[1,2]

***Abstract*— Caging grasps are valuable as they can be robust to bounded variations in object shape and pose, do not depend on friction, and enable transport of an object without full immobilization. Complete caging of an object is useful but may not be necessary in cases where forces such as gravity are present. This paper extends caging theory by defining energy-bounded cages with respect to an energy field such as gravity. This paper also introduces Energy-Bounded-Cage-Analysis-2D (EBCA-2D), a sampling-based algorithm for planar analysis that takes as input an energy function over poses, a polygonal object, and a configuration of rigid fixed polygonal obstacles, e.g. a gripper, and returns a lower bound on the minimum escape energy. In the special case when the object is completely caged, our approach is independent of the energy and can provably verify the cage. EBCA-2D builds on recent results in collision detection and the computational geometric theory of weighted α-shapes and runs in $O(s^2 + sn^2)$ time where s is the number of samples, n is the total number of object and obstacle vertices, and typically $n << s$. We implemented EBCA-2D and evaluated it with nine parallel-jaw gripper configurations and four nonconvex obstacle configurations across six nonconvex polygonal objects. We found that the lower bounds returned by EBCA-2D are consistent with intuition, and we verified the algorithm experimentally with Box2D simulations and RRT* motion planning experiments that were unable to find escape paths with lower energy. EBCA-2D required an average of 3 minutes per problem on a single-core processor but has potential to be parallelized in a cloud-based implementation. Additional proofs, data, and code are available at: `http://berkeleyautomation.github.io/caging/`.**

I. INTRODUCTION

Consider a single movable object and a configuration of fixed obstacles that define a set of gripper jaws or fingers. The object is caged if it cannot escape [22], [31]. Caging grasps are valuable as they can be robust to bounded variations in object shape and pose [3], [35], [45], do not depend on friction, and are sufficient in applications that do not require complete immobilization such as nonprehensile manipulation [26].

This paper extends caging theory by defining energy-bounded cages under an energy field such as gravity based on the minimum energy required to escape. For example, the blue objects in Fig. 1 would have to overcome gravitational forces to escape by moving above the fixed black obstacles. However, computing the minimum escape energy may be challenging for arbitrary nonconvex objects and obstacles because there can be an uncountable number of paths that an object can take to escape a cage. Currently, complete cages in 2D can only be verified under assumptions on the number of gripper fingers [3], [29], [40], the geometry of obstacles [35], or the geometry of the object [16], [41].

[1]Department of Electrical Engineering and Computer Sciences; {jmahler, ftpokorny, zmccarthy}@berkeley.edu
[2]Department of Industrial Engineering and Operations Research; goldberg@berkeley.edu
[1−2] University of California, Berkeley, USA
[3]Department of Information and Computing Sciences, Utrecht University, The Netherlands; A.F.vanderStappen@uu.nl

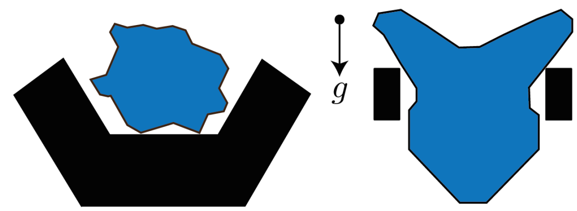

Fig. 1: Two energy-bounded cages of industrial parts (blue) by robotic grippers (black) under a gravitational field indicated by the center arrow. Neither object is completely caged, but both are energy-bounded caged.

This paper formally defines energy-bounded cages and introduces Energy-Bounded-Cage-Analysis-2D (EBCA-2D), a sampling-based algorithm that can verify cages and energy-bounded cages for general 2D polygonal objects and an arbitrary number of fixed polygonal obstacles under an energy field. The algorithm computes a lower-bound on the minimum escape energy using weighted α-shapes [13], [14], a well-studied data structure from computational geometry that facilitates checking the connectivity of an approximation to the configuration space and has been used for proving path non-existence in motion planning [4], [28], [47]. We use weighted α-shapes to discretize the object configuration space into cells from a set of sampled object poses and a conservative approximation of the penetration depth between the object and obstacles. We then mark cells that lie strictly within the collision space or above an energy threshold as forbidden and examine the connectivity of the free cells to prove the non-existence of object escape paths [28]. Finally, we lower bound the minimum escape energy by performing a binary search over energy levels, querying the connectivity of the free cells for each threshold. When the returned lower bound is arbitrarily large, the object is provably caged.

We evaluated EBCA-2D on a set of nine parallel-jaw gripper configurations and four configurations of obstacles across six polygonal objects under gravity. In each case, neither 120 seconds of RRT* optimal path planning nor 1,000 trials of dynamic simulation in Box2D generated an escape path with lower energy than the estimated lower bound.

II. Related Work

For surveys of the substantial literature on grasping, see Bicchi and Kumar [5] or Prattichizzo and Trinkle [30]. Many methods for grasp analysis use either the wrench space or configuration space. Methods based on the wrench space measure the ability of a grasp to resist external forces and torques applied to a grasped object [17], [21], [25]. While wrench space metrics depend on local properties of an object, cages depends on the configuration space, which concerns the global geometry of an object and gripper. Early research defined a cage as a configuration of a gripper defined by n isolated points in the plane such that a planar object could not be moved arbitrarily far away from the gripper [22], [33]. Rimon and Blake [31] later characterized the space of caging configurations for a 1-parameter two-fingered gripping system with convex fingers. Rimon and Blake [32] also developed an algorithm to determine the maximal set of two-finger gripper configurations that cage an object, which was later extended to three fingers by Davidson and Blake [11]. Other research has presented algorithms for computing the set of caging configurations for grasps with two or three disc fingers on convex polygons [16], non-convex polygons [29], [40] and grasps on 3D polyhedra with two point fingers [3]. Recently, Allen et al. [2] proposed an algorithm for identifying all cages of polygonal shapes by two point-fingers using a low-dimensional contact-space formulation. In contrast, our work focuses on a sampling-based approach for verifying a particular cage with an arbitrary number of polygonal fingers and with an additional energy constraint.

Several works have studied the relation of caging configurations to uncertainty and to form closure grasps. Vahedi and van der Stappen [40] developed the concepts of squeezing and stretching cages for two-finger grippers and showed that two-finger cages in the plane can always lead to a form closure grasp by either opening or closing the fingers. Rodriguez and Mason [34] extended this property to two finger cages of compact and contractible objects in arbitrary dimensions, and later generalized the link between caging and grasping to more than two fingers, showing that cages can be a useful waypoint to a form closure grasp of a polygonal object when the gripper stays in a sub- or super-level set of a gripper shape function [35]. Cages have also been shown experimentally to offer robustness to shape and pose uncertainty. Diankov et al. [12] found that caging grasps were empirically more successful than those ranked by local force closure metrics when manipulating articulated objects with handles. Other work has studied the robustness of caging grasps to object pose uncertainty [45] or uncertainty in object shape due to vision [37].

Due to the complexity of caging in 3D, recent research [36] has focused on specific object families that exhibit holes or handles to determine cages with complex robot hands. In 2D, Makpunyo et al. [27] introduced the concept of partial cage quality, arguing that configurations that allow only rare escape motions may be successful in practice. The authors proposed a heuristic metric based on the length and curvature of escape paths generated by a motion planner. Wan et al. [44] determined cages for 2D polygons by mapping out the configurations in collision in a voxelized representation of the configuration space and checking connectivity. In comparison, we present a formal definition and metric of energy-bounded cages and formally prove that a cell discretization of the 3D configuration space can be used to verify cages and energy-bounded cages.

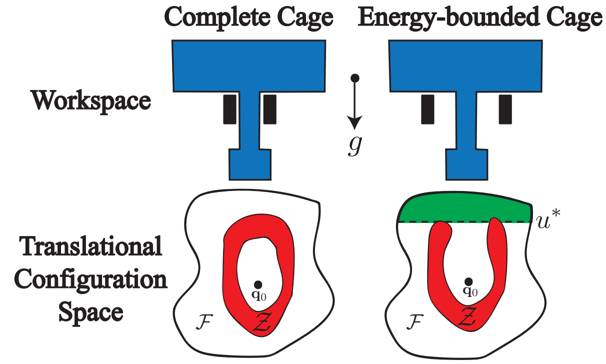

Fig. 2: A complete cage and an energy-bounded cage illustrated in the workspace (top) and translational configuration space (bottom) for a gravitational potential field. (Left) An object (blue) is caged by obstacles (black) if it cannot reach poses arbitrarily far away from its initial pose $\mathbf{q}_0$. In configuration space this corresponds to a disconnection between the exterior free space $\mathcal{F}$ and the object initial pose $\mathbf{q}_0$ which is completely enclosed by the collision space $\mathcal{Z}$. (b) A gripper forms an energy-bounded cage of an object with minimum escape energy u^* if all escape paths cross the u^*-superlevel set of a potential energy function $U : SE(2) \to \mathbb{R}$ (yellow).

Our work is also related to the problem of proving path existence and non-existence in the field of motion planning. When the free configuration space can be described by semi-algebraic functions, the free space can be analytically discretized into cells to answer path existence queries [23]. However, such a semi-algebraic description might be prohibitively expensive to compute, motivating alternative methods. Basch et al. [4] provided a quadratic-time algorithm to prove path non-existence of a polygon through a polygonal hole in an infinte wall. Zhang et al. [47] developed a method for approximately decomposing the free space and obstacle space for a robot into rectangular cells, labelling cells as being in collision using penetration depth computation, and searching for paths through cells in free space. We build upon the results of McCarthy et al. [28], which used configuration samples to approximate the collision space using α-shapes [13] and presented an algorithm that can verify path non-existence between two configurations.

III. Definitions

A. Problem

We consider the problem of caging a compact 2D polygonal object $\mathcal{O} \subset \mathbb{R}^2$ by a fixed configuration of compact polygonal obstacles $\mathcal{G} \subset \mathbb{R}^2$ and an energy function $U : SE(2) = \mathbb{R}^2 \times \mathbb{S}^1 \to \mathbb{R}$ that is convex when restricted to $\mathbb{R}^2$, such as gravity. We consider $\mathcal{G}$ to be fixed in the environment and denote the object polygon in pose $\mathbf{q} \in SE(2)$ relative

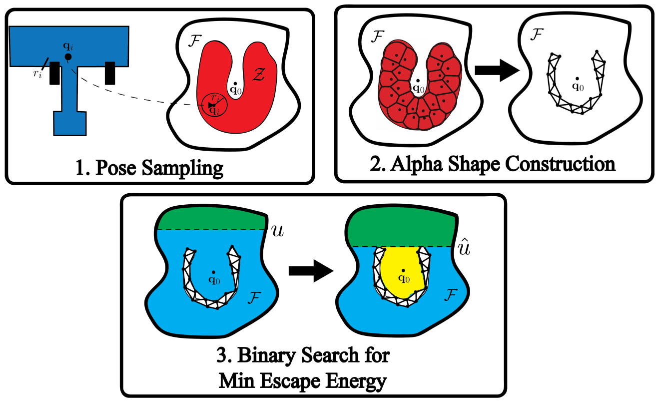

Fig. 3: Illustration of the steps of the EBCA-2D algorithm (best viewed in color). Given a polygonal object (blue), set of polygonal obstacles / fingers (black) and an energy function U, EBCA-2D finds a lower bound on the minimum escape energy or reports that the object is completely caged. 1) The first step is to conservatively approximate the collision space by uniformly sampling s object poses and keeping only poses where the object is in collision with the obstacles. For each collision pose, we compute the penetration depth r_i which defines a ball strictly inside the collision space $\mathcal{Z}$ (red). 2) The union of these balls conservatively approximates $\mathcal{Z}$. We then discretize the configuration space into cells by computing the weighted Delaunay triangulation D from the points and use the weighted α-shape $\mathcal{A}$ with $\alpha = 0$ to classify the cells belonging to $\mathcal{Z}$ (the white trianglular mesh). 3) Finally, we use binary search to find the maximum energy u for which no escape path exists by classifying the set of forbidden cells $\mathcal{V}_u$, classifying the connected components of $D \setminus \mathcal{V}_u$, and checking if the component containing the initial object pose $\mathbf{q}_0$ is bounded. Blue and yellow indicate connected components, while green indicates poses such that $U(\mathbf{q}) > u$.

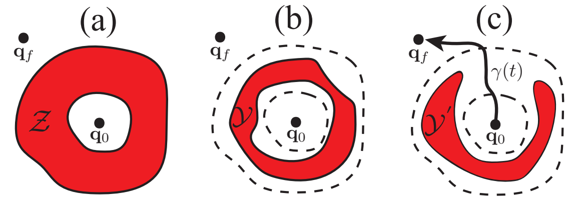

Fig. 4: Illustration of the subset property of cages. (a) An object in pose $\mathbf{q}_0$ is caged because it cannot reach poses $\mathbf{q}_f$ arbitrarily far away without crossing a forbidden region $\mathcal{Z}$. (b) If no path from $\mathbf{q}_0$ to $\mathbf{q}_f$ exists without passing through a subset $\mathcal{Y} \subseteq \mathcal{Z}$, then the object must be caged. (c) However, we are not guaranteed to verify a cage using any subset $\mathcal{Y}' \subseteq \mathcal{Z}$ because $\mathcal{Y}'$ may not block all paths from $\mathbf{q}_0$ to $\mathbf{q}_f$.

to its initial configuration $\mathbf{q}_0 \in SE(2)$ as $\mathcal{O}(\mathbf{q})$. Example obstacles $\mathcal{G}$ include the end-effectors of a robotic gripper or parts of the environment such as walls or support surfaces.

B. Complete Caging

Consider a bounded subset $\mathcal{Z}$ within the the configuration space $\mathcal{C}$ of a object:

Definition 3.1: Let $\mathcal{C}$ be a subset of $SE(2)$, the set of rigid transformations in the plane, and $\mathcal{Z} \subset \mathcal{C}$. We call a point $\mathbf{x} \in \mathcal{C} \setminus \mathcal{Z}$ *completely caged by* $\mathcal{Z}$ *in* $\mathcal{C}$ if $\mathbf{x}$ lies in a bounded path-component of $\mathcal{C} \setminus \mathcal{Z}$.

This definition is illustrated in the left panel of Fig. 2. Note that a complete cage guarantees that no continuous path in $\mathcal{C} \setminus \mathcal{Z}$ exists from $\mathbf{x}$ to a point arbitrarily far away. We can verify complete cages using a sufficient condition:

Lemma 3.1: Let $\mathcal{Y} \subseteq \mathcal{Z} \subset \mathcal{C}$. If $\mathbf{x} \in \mathcal{C} \setminus \mathcal{Z}$ is completely caged by $\mathcal{Y}$, then $\mathbf{x}$ is completely caged by $\mathcal{Z}$.

Proof: $\mathcal{C} \setminus \mathcal{Z} \subseteq \mathcal{C} \setminus \mathcal{Y}$, which implies that any path in $\mathcal{C} \setminus \mathcal{Z}$ can be restricted to $\mathcal{C} \setminus \mathcal{Y}$. ■

This property is illustrated in Fig. 4. Thus if we can prove a complete cage condition for a subset $\mathcal{Y}$ of the true set of interest $\mathcal{Z}$, then the result holds for $\mathcal{Z}$.

In this work, we are interested in the case where $\mathcal{Z} \subseteq SE(2)$ is the *collision space* of $\mathcal{O}$ relative to $\mathcal{G}$ [23]:

$$\mathcal{Z} = \left\{ \mathbf{q} \in SE(2) \middle| \operatorname{int}(\mathcal{O}(\mathbf{q})) \cap \mathcal{G} \neq \varnothing \right\}.$$

We denote by $\mathcal{F} = SE(2) \setminus \mathcal{Z}$ the *free configuration space*.

C. Energy-Bounded Caging

When the object can escape, we seek to quantify the energy required for the object to escape. Define $U^{-1}(X) = \{\mathbf{q} \in SE(2) \mid U(\mathbf{q}) \in X\}$ for any subset $X \subseteq \mathbb{R}$. Given an energy threshold $u \in \mathbb{R}$, we denote by $\mathcal{Z}_u = \mathcal{Z} \cup U^{-1}([u, \infty))$ the *u-energy forbidden space* and by $\mathcal{F}_u = SE(2) \setminus \mathcal{Z}_u$ the *u-energy admissible space*. Using the previous definitions, we formally introduce a the notion of an energy-bounded cage:

Definition 3.2: We call $\mathcal{G}$ a *u-energy-bounded cage* of $\mathcal{O}$ with respect to U if the initial configuration $\mathbf{q}_0 \in SE(2)$ of $\mathcal{O}$ lies in a bounded path-connected component of $\mathcal{F}_u$.

When u can be arbitrarily large, we obtain the standard notion of complete caging of a polygonal object $\mathcal{O}$ relative to $\mathcal{G}$ [22]. Fig. 2 illustrates both complete caging and a u-energy-bounded cage with respect to the gravitational potential energy $U(\mathbf{q}) = Mg(y - y_0)$, where M is the mass of the object and g is the acceleration due to gravity. To measure energy-bounded cages, we introduce the notion of the minimum escape energy:

Definition 3.3: The *minimum escape energy*, denoted u^*, is the infimum over u such that $\mathcal{G}$ is not a u-energy-bounded cage of $\mathcal{O}$ when such an infimum exists, and otherwise $u^* = \infty$.

The rest of this work is dedicated to computing a lower bound on u^* for a fixed configuration.

IV. Methodology

The EBCA-2D algorithm takes as input an object $\mathcal{O}$, obstacle configuration $\mathcal{G}$, and energy function U, and outputs a lower bound on the minimum escape energy or reports that the object is completely caged. Fig. 3 illustrates EBCA-2D.

We first generate s samples of object poses $\mathcal{Q} = \{\mathbf{q}_1, ..., \mathbf{q}_s\} \subset SE(2)$ in collision and lift the samples into $\mathbb{R}^3$ to form a set $\mathcal{X}$. Next, we compute a conservative estimate of the penetration depth for each lifted pose to form a set $\mathcal{R}$, and we prove that $\mathcal{R}$ can be used to construct a subset of the collision space. To prove that the object is caged up to an energy threshold u, we must show that the initial object pose is completely enclosed by the u-energy forbidden space $\mathcal{Z}_u$. To do so, we utilize weighted α-shapes with parameter $\alpha = 0$ to conservatively approximate $\mathcal{Z}_u$ with cells constructed from $\mathcal{X}$ and $\mathcal{R}$. We then use connectivity checking to prove non-existence of escape paths based on recent results of [28]. Finally, we use binary search to determine a lower bound of u for which no path exists in our cell decomposition.

A. Verifying Cages in SE(2)

Given a set of sampled poses in collision $\mathcal{Q} = \{\mathbf{q}_1, ..., \mathbf{q}_s\}$ where each $\mathbf{q}_i \in SE(2)$, the first step of our algorithm is to embed the samples in $\mathbb{R}^3$. Let $\mathbf{z}$ be the center of mass of $\mathcal{O}$ and $\rho = \max_{\mathbf{v} \in \mathcal{O}} \|\mathbf{v} - \mathbf{z}\|_2$ be the maximum moment arm of $\mathcal{O}$. Then let $\pi : \mathbb{R}^3 \to SE(2)$ be the covering map defined by $\pi(x, y, z) = (x, y, (z/\rho) \bmod 2\pi)$, for $(x, y, z) \in \mathbb{R}^3$. We map from poses to the covering space with an inverse map $\pi_k^{-1} : SE(2) \to \mathbb{R}^3$ defined by $\pi_k^{-1}(x, y, \theta) = (x, y, \rho\theta + 2\pi k)$ for $k \in \mathbb{Z}$ [9]. Given $v \in \mathbb{Z}$, a fixed number of rotations to embed, our lifted set of pose samples is $\mathcal{X} = \left\{\hat{\mathbf{q}}_{i,k} = \pi_k^{-1}(\mathbf{q}_i) \mid \mathbf{q}_i \in \mathcal{Q}, k \in \{-v, ..., 0, ..., v\}\right\}$.

We relate path existence in the covering space to cages in the configuration space by means of the following result [19]:

Theorem 4.1: Let $\mathcal{Y} \subset \mathbb{R}^3$ be a bounded subset, let $\text{Conv}(\mathcal{Y})$ denote the convex hull of $\mathcal{Y}$, and let $\overline{\text{Conv}}(\mathcal{Y})$ denote the closure of $\text{Conv}(\mathcal{Y})$. Let $\mathbf{q}_0 \in SE(2)$ such that $\mathbf{q}_0 \in \pi(\text{Conv}(\mathcal{Y})) \setminus \pi(\mathcal{Y})$ and let $\hat{\mathbf{q}}_0$ be any point such that $\pi(\hat{\mathbf{q}}_0) = \mathbf{q}_0$. If there exists no continuous path from $\hat{\mathbf{q}}_0 \in \mathbb{R}^3$ to $\partial\overline{\text{Conv}}(\mathcal{Y}) \subset \mathbb{R}^3$ in $\mathbb{R}^3 \setminus \mathcal{Y}$, then $\mathbf{q}_0 \in SE(2)$ is caged by $\pi(\mathcal{Y})$ in $SE(2)$.

Proof: Suppose the contrary. Since $\mathbf{q}_0$ is not caged by $\pi(\mathcal{Y})$ in $SE(2)$ and $\mathbb{S}^1$ is compact, there exists a continuous escaping path $\gamma(t) : [0, 1] \to SE(2) \setminus \pi(\mathcal{Y})$ such that $\gamma(0) = \mathbf{q}_0 = (x_0, y_0, \theta_0)$ and $\gamma(1) = (x_1, y_1, \theta_1)$ where $\|(x_0, y_0) - (x_1, y_1)\|_2 > \text{diam}(\overline{\text{Conv}}(\mathcal{Y}))$. By the properties of the covering map π, there exists a lifting of γ to a covering path $\hat{\gamma} : [0, 1] \to \mathbb{R}^3 \setminus \mathcal{Y}$ with $\hat{\gamma}(0) = \hat{\mathbf{q}}_0$ and $\pi(\hat{\gamma}(t)) = \gamma(t)$ for all $t \in [0, 1]$, where $\|\hat{\gamma}(0) - \hat{\gamma}(1)\|_2 > \text{diam}(\overline{\text{Conv}}(\mathcal{Y}))$. Hence by the continuity of $\hat{\gamma}(t)$ there exists a smallest $t_0 \in [0, 1]$ such that $\hat{\gamma}(t_0) \in \partial\overline{\text{Conv}}(\mathcal{Y})$ and $\hat{\gamma}([0, t_0]) \subseteq \mathbb{R}^3 \setminus \mathcal{Y}$. This contradicts our supposition that no continuous path exists from $\hat{\mathbf{q}}_0$ to $\partial\overline{\text{Conv}}(\mathcal{Y})$. ■

This result implies that a lifting of the u-energy forbidden space $\hat{\mathcal{Z}}_u \subset \mathbb{R}^3$ such that $\pi(\hat{\mathcal{Z}}_u) = \mathcal{Z}_u$ can be used to check the existence of energy-bounded cages.

B. Approximating the u-Energy Forbidden Space $\mathcal{Z}_u$

It remains to construct a conservative approximation of the lifted u-energy forbidden space $\mathcal{V}_u \subseteq \hat{\mathcal{Z}}_u$ and to computationally prove path non-existence in the lifted space, which would prove an energy-bounded cage by Lemma 3.1 and Theorem 4.1. We first approximate the lifted collision space $\hat{\mathcal{Z}}$ by a set $\mathcal{B}$ using a conservative estimate of penetration depth, then discretize the convex hull of $\mathcal{B}$ into cells using weighted α-shapes, and finally form $\mathcal{V}_u$ from cells lying strictly within $\hat{\mathcal{Z}}_u$.

1) Approximating the Collision Space Using Penetration Depth : The 2D generalized penetration depth (GPD) $p : SE(2) \to \mathbb{R}$ between an object $\mathcal{O}(\mathbf{q}_i)$ in pose $\mathbf{q}_i = (x_i, y_i, \theta_i)$ and obstacle $\mathcal{G}$ is defined as [48]:

$$p(\mathbf{q}_i) = \min_{\mathbf{q}_j \in SE(2)} \left\{ d(\mathbf{q}_i, \mathbf{q}_j) \middle| int\,(\mathcal{O}(\mathbf{q}_j)) \cap \mathcal{G} = \varnothing \right\}.$$

where $d : SE(2) \times SE(2) \to \mathbb{R}$ is a distance metric between poses. Following Zhang et. al [48], we use $d(\mathbf{q}_i, \mathbf{q}_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} + \min_{m \in \mathbb{Z}} \rho|\theta_i - (\theta_j + 2\pi m)|$, which has the following important property:

Lemma 4.1: Let $r_i = r(\mathbf{q}_i) : SE(2) \to \mathbb{R}$ be an approximate solution to the above equation such that $r_i \leqslant p(\mathbf{q}_i)$ for all $\mathbf{q}_i \in \mathcal{C}$ and let $\mathbb{B}_r(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^3 : \|\mathbf{x} - \mathbf{y}\| \leqslant r\}$ be a standard Euclidean ball of radius r centered at $\mathbf{x} \in \mathbb{R}^3$. For any lifted pose $\hat{\mathbf{q}}_i$, if $\pi(\hat{\mathbf{q}}_i) \in \mathcal{Z}$, then any lifted pose $\hat{\mathbf{q}}_j \in \mathbb{B}_{r_i}(\hat{\mathbf{q}}_i)$ also satisfies $\pi(\hat{\mathbf{q}}_j) \in \mathcal{Z}$.

A detailed version of the proof is given in the supplemental file at `http://berkeleyautomation.github.io/caging/`, and a similar proof is given by Zhang et al. [47]. For our set of pose samples $\mathcal{Q} \subset SE(2)$ with an associated lifting $\mathcal{X} \subset \mathbb{R}^3$ and associated GPD values $\mathcal{R} = \{r_{i,k} = r(\mathbf{q}_i) \mid \mathbf{q}_i \in \mathcal{Q}, k \in \{-v..., 0, ..., v\}\}$, define

$$\mathcal{B}(\mathcal{X}, \mathcal{R}) = \bigcup_{\mathcal{X}, \mathcal{R}} \mathbb{B}_{r_{i,k}}(\hat{\mathbf{q}}_{i,k}).$$

It follows from Lemma 4.1 that $\pi(\mathcal{B}(\mathcal{X}, \mathcal{R})) \subseteq \mathcal{Z}$.

In order to satisfy $r_i \leqslant p(\mathbf{q}_i)$, we use an algorithm by Zhang et. al [48] to lower bound the GPD between any two objects. The algorithm assumes a given convex decomposition of the two bodies [24], then computes the exact GPD between all possible pairs of convex pieces and takes the maximum over the GPD values between the pieces. In the worst case, we can decompose the polygons into $O(n)$ triangles, where n is the total number of vertices between $\mathcal{O}$ and $\mathcal{G}$. Then we can compute the exact GPD between two convex bodies using the Gilbert-Johnson-Keerthi Expanding Polytope Algorithm (GJK-EPA) developed by van den Bergen [42] and implemented in libccd [18], which takes $O(1)$ time for triangles [8]. There are up to $O(n^2)$ pairs of triangles to check, and therefore the total complexity of computing GPD is $O(n^2)$.

2) Weighted α-Shapes: Weighted α-shapes [13], [14], [15], illustrated in Fig. 5, are a well-studied tool from computational geometry that facilitate computational checks of complete cages and energy-bounded cages by discretizing the configuration space into cells. We use weighted α-shapes with $\alpha = 0$ to determine which cells belong strictly to the lifted collision space $\hat{\mathcal{Z}}$.

Weighted α-shapes are a type of simplicial complex [14], a key data-structure to represent a large collection of geometrically interesting spaces that generalize the notion of a graph and a triangulation. Let $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_s\} \subset \mathbb{R}^3$ be a point set and $\mathcal{R} = \{r_1, ..., r_s\}$ be positive scalars for each element of $\mathcal{X}$ such that any subset of 4 points of $\mathcal{X}$ are affinely independent. This is a weak condition since for uniformly sampled points this occurs with probability one [15]. The weighted Delaunay triangulation (WDT) of $\mathcal{X}$ and $\mathcal{R}$ is $D(\mathcal{X}, \mathcal{R}) = \{\sigma = \{\mathbf{x}_0, \ldots, \mathbf{x}_k\} \mid \cap_{i=1}^k V_{\mathbf{x}_i}(\mathcal{X}, \mathcal{R}) \neq \emptyset \text{ and } 0 \leqslant k \leqslant 3\}$, where $V_{\mathbf{x}_i}(\mathcal{X}, \mathcal{R}) = \{\mathbf{y} \mid \|\mathbf{x}_i - \mathbf{y}\|^2 - r_i^2 \leqslant \|\mathbf{x}_j - \mathbf{y}\|^2 - r_j^2, \forall j \in \{1, \ldots, P\}\}$ is the weighted Voronoi region for $\mathbf{x}_i$. The union of all simplices in $D(\mathcal{X}, \mathcal{R})$ is $\text{Conv}(\mathcal{X})$ and when $r_i = 0$ for all i this reduces to the standard Delaunay triangulation of $\mathcal{X}$. The weighted α-shape $\mathcal{A} = \mathcal{A}(\mathcal{X}, \mathcal{R})$ at $\alpha = 0$ is a particular simplicial subcomplex of $D(\mathcal{X}, \mathcal{R})$ with several important properties:

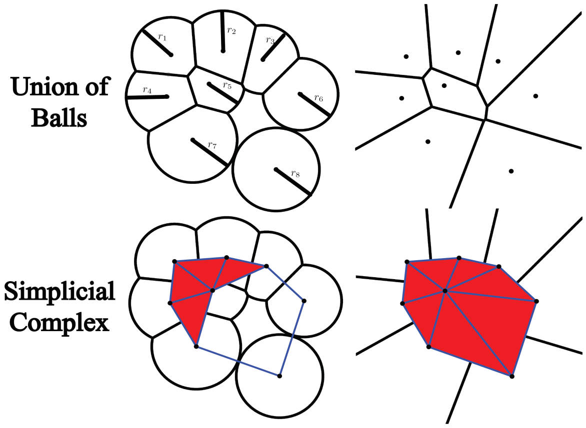

Fig. 5: Weighted α-shape construction and representation. (Top-left) The α-shape $\mathcal{A}$ with $\alpha = 0$ is constructed from a set of Euclidean balls centered at points $\mathcal{X}$ with radii $\mathcal{R} = \{r_1, .., r_s\}$ (Bottom-left) $\mathcal{A}$ contains edges and triangles between the pairs and triplets of with a common intersection, respectively (bottom). (Top-right) As we increase the ball radius towards ∞ the set of balls becomes the power diagram of the point set, a generalization of the Voronoi diagram. (Bottom-right) The triangulation of the power diagram is the weighted Delaunay triangulation $D(\mathcal{X}, \mathcal{R})$ of the points, which contains the convex hull.

Theorem 4.2 (Edelsbrunner et. al): Let $\mathcal{B}(\mathcal{X}, \mathcal{R}) = \bigcup_{i=1}^{s} \mathbb{B}_{r_i}(\mathbf{x}_i)$. Then any k-simplex $\sigma = \{\mathbf{x}_{i_0}, \dots, \mathbf{x}_{i_k}\}$ in $\mathcal{A}$ such that $i_k \in \{1, .., s\}$ and $0 \leqslant k \leqslant 3$ is completely contained in the union of balls $\bigcup_{j=0}^{k} \mathbb{B}_{r_{i_j}}(\mathbf{x}_{i_j})$.
This implies that if we use the set of lifted poses $\mathcal{X}$ with radii given by the penetration depths $\mathcal{R}$, then the weighted alpha shape $\mathcal{A}(\mathcal{X}, \mathcal{R}) \subset \hat{\mathcal{Z}}$ and can be used to verify cages by Lemma 3.1 and Theorem 4.1.

3) Approximating the Potential Energy Superlevel Set: It remains to find a subcomplex of $D(\mathcal{X}, \mathcal{R})$ such that $\pi(\mathbf{x}) \in U^{-1}([u, \infty))$ for any $\mathbf{x}$ in the subcomplex.

Lemma 4.2: For any k-simplex $\sigma \in D(\mathcal{X}, \mathcal{R})$, let $U(\sigma) = \min_{\mathbf{x} \in \sigma} U(\pi(\mathbf{x}))$. Furthermore, let $\mathcal{P}_u(\mathcal{X}, \mathcal{R}) = \{\sigma \in D(\mathcal{X}, \mathcal{R}) \mid U(\sigma) > u\}$. Then $\mathcal{P}_u(\mathcal{X}, \mathcal{R})$ is a subcomplex of $D(\mathcal{X}, \mathcal{R})$ and $\mathcal{P}_u(\mathcal{X}, \mathcal{R}) \subseteq U^{-1}([u, \infty))$.

Proof: Fix a simplex $\sigma_k \in \mathcal{P}_u(\mathcal{X}, \mathcal{R})$. Any face σ_j of σ_k is also a member of $\mathcal{P}_u(\mathcal{X}, \mathcal{R})$ by the minimum over the energy function. Furthermore, by the convexity of U when restriced to $\mathbb{R}^2$, any point $\mathbf{y}$ in $\mathrm{Conv}(\sigma_k)$ satisfies $U(\pi(\mathbf{y})) \geqslant U(\sigma_k) \geqslant u$, and therefore $\mathbf{y} \in U^{-1}([u, \infty))$. ■
A result of this Lemma and Theorem 4.2 is that the *u-energy forbidden subcomplex* $\mathcal{V}_u$ satisfies

$$\mathcal{V}_u(\mathcal{X}, \mathcal{R}) = \mathcal{A}(\mathcal{X}, \mathcal{R}) \cup \mathcal{P}_u(\mathcal{X}, \mathcal{R}) \subset \hat{\mathcal{Z}}_u.$$

C. Verifying Path Non-Existence

We can now verify u-energy-bounded cages by showing that no path exists from the lifting of the object pose $\hat{\mathbf{q}}_0$ to $\partial D(\mathcal{X}, \mathcal{R})$ in $D(\mathcal{X}, \mathcal{R}) \setminus \mathcal{V}_u(\mathcal{X}, \mathcal{R})$ by Theorem 4.1. We use Algorithm 1, a modified version of the algorithm by McCarthy et al. [28], to verify that no escape paths exist. As shown by McCarthy et al. [28], the worst-case runtime to verify path non-existence is $O(s^2)$ but is often $O(s)$ in practice, where s is the number of sampled points. The runtime is dominated by the construction of the weighted Delaunay triangulation $D(\mathcal{X}, \mathcal{R})$. Given $D(\mathcal{X}, \mathcal{R})$, Algorithm 1 takes $O(s)$ time in the worst case because each simplex in $D(\mathcal{X}, \mathcal{R})$ must be classified to construct a disjoint set structure.

Theorem 4.3: If $\mathcal{V}_u$ is any subcomplex of $D(\mathcal{X}, \mathcal{R})$ in $\mathbb{R}^3$ such that $\hat{\mathbf{q}}_0 \in \mathrm{Conv}(\mathcal{X}) \setminus \mathcal{V}_u$ and Algorithm 1 returns True, then there exists no continuous path from $\hat{\mathbf{q}}_0$ to $\partial\,\mathrm{Conv}(\mathcal{X})$ in $D(\mathcal{X}, \mathcal{R}) \setminus \mathcal{V}_u$.

The proof is a slight modification of the main Theorem of [28] and is given in the supplemental file. Therefore, if Algorithm 1 returns True when run with $\mathcal{V}_u$ as defined in Section IV-B.3, then we are guaranteed that $\mathcal{V}_u$ forms a u-energy bounded cage of $\hat{\mathbf{q}}_0$. If Algorithm 1 returns False then we cannot determine whether or not $\mathcal{V}_u$ forms a u-energy bounded cage of $\hat{\mathbf{q}}_0$.

```
Input: Lifted initial pose q̂0, weighted Delaunay triangulation
  D(X, R), u-Energy Forbidden Subcomplex V_u
Result: True if we verify that V_u cages O in pose π(q̂0),
        False if undecided
// Init free subcomplex and boundary
U = {σ_i | σ_i ∈ D(X, R) \ V_u, |σ_i| = 3, };
W = {σ_j | σ_j ∈ ∂D(X, R) \ V_u, |σ_j| = 2};
// Compute connected components
Q =DisjointSetStructure(U ∪ W);
for σ_i ∈ U ∪ W do
    for σ_j ∈Neighbors(σ_i, D(X, R) \ V_u) do
        if σ_i ∩ σ_j ∉ V_u then
            Q.UnionSets(σ_i, σ_j);
    end
end
// Check connectivity
σ_0 = Locate(q̂0, D(X, R));
for σ_i ∈ W do
    if Q.SameSet(σ_0, σ_i) then
        return False;
end
return True;
```

Algorithm 1: Verifying u-Energy-Bounded Cages

D. Lower-Bounding the Minimum Escape Energy

We determine a lower bound to u^* by searching over values of u that form an energy-bounded cage. EBCA-2D, our full algorithm for computing the a lower bound, is given in Algorithm 2. EBCA-2D generates s samples of poses in collision $\mathcal{Q}$ with penetration depths $\mathcal{R}$ over the collision space using rejection sampling, embeds the poses in $\mathbb{R}^3$ using to form a set $\mathcal{X}$, constructs a weighted Delaunay triangulation $D(\mathcal{X}, \mathcal{R})$ and alpha shape $\mathcal{A}(\mathcal{X}, \mathcal{R})$ from the samples, and finds and approximation $\hat{u}$ to u^* using binary search, where on each iteration we check for an energy-bounded cage using Algorithm 1. The complexity of EBCA-2D is $O(s^2 + sn^2)$, where the $O(s^2)$ term is due to constructing $D(\mathcal{X}, \mathcal{R})$ and the $O(sn^2)$ term is due to the computation of the GPD for s pose samples.

Theorem 4.4: Let u^* denote the minimum escape energy for object $\mathcal{O}$ and obstacle configuration $\mathcal{G}$. Let $\hat{u}$ be the result of running Algorithm 2 with $\mathcal{O}$ and $\mathcal{G}$. Then $\hat{u} \leqslant u^*$.

Proof: By Lemma 3.1 and the subset properties of $\mathcal{A}(\mathcal{X}, \mathcal{R})$ from Lemma 4.1 and Theorem 4.2 we are guaranteed that if our algorithm terminates when checking $u = \infty$, then the object is completely caged. It remains to show that for all iterations of the binary search, the gripper configuration $\mathcal{G}$ is a u_ℓ-energy-bounded cage. This is true for iteration 0, as the initial value satisfies $u_\ell \leqslant U(\mathbf{q}_0)$. Furthermore, if the lower bound is updated to $u_\ell = u_m$ then $\mathcal{G}$ is a u_m-energy-bounded cage of $\mathcal{O}$ by Theorem 4.3, Lemma 4.1, and Lemma 4.2. ■

Input: Polygonal obstacles $\mathcal{G}$, Polygonal object $\mathcal{O}$, Number of pose samples s, Number of rotations v for $SE(2)$ lifting, Binary search resolution Δ
Result: $\hat{u}$, a lower bound on the minimum escape energy u^*
`// Sample poses in collision`
$\mathcal{Q} = \varnothing, \mathcal{R} = \varnothing, \ell = diam(\mathcal{G}) + diam(\mathcal{O})$;
$\mathcal{W} = [-\ell, \ell] \times [-\ell, \ell] \times [0, 2\pi)$;
for $i \in \{1, ..., s\}$ **do**
  $\mathbf{q}_i$ = RejectionSample($\mathcal{W}$);
  r_i = LowerBoundPenDepth($\mathbf{q}_i, \mathcal{O}, \mathcal{G}$);
  if $r_i > 0$ **then**
    $\mathcal{Q} = \mathcal{Q} \cup \{\mathbf{q}_i\}$, $\mathcal{R} = \mathcal{R} \cup \{r_i\}$;
end
$\mathcal{X} = \{\pi_k^{-1}(\mathbf{q}_i) \mid \mathbf{q}_i \in \mathcal{Q}, k \in \{-v, ..., v\}\}$;
`// Create alpha shape`
$D(\mathcal{X}, \mathcal{R})$ = WeightedDelaunayTriangulation($\mathcal{X}, \mathcal{R}$);
$\mathcal{A}(\mathcal{X}, \mathcal{R})$ = WeightedAlphaShape($D(\mathcal{X}, \mathcal{R}), \alpha = 0$);
`// Binary search for min escape energy`
if *EnergyBoundedCage*$(\mathbf{q}_0, D(\mathcal{X}, \mathcal{R}), \mathcal{A}(\mathcal{X}, \mathcal{R}))$ **then**
  return ∞;
u_ℓ = min $U(\sigma_k)$ such that $\sigma_k \in D(\mathcal{X}, \mathcal{R})$;
u_u = max $U(\sigma_k)$ such that $\sigma_k \in D(\mathcal{X}, \mathcal{R})$;
while $|u_u - u_\ell| > \Delta$ **do**
  $u_m = 0.5(u_\ell + u_u)$;
  $\mathcal{V}_{u_m}$ = ForbiddenSubcomplex($D(\mathcal{X}, \mathcal{R}), \mathcal{A}(\mathcal{X}, \mathcal{R}), u_m$);
  if *EnergyBoundedCage*$(\mathbf{q}_0, D(\mathcal{X}, \mathcal{R}), \mathcal{V}_{u_m})$ **then**
    $u_\ell = u_m$;
  else
    $u_u = u_m$;
end
return u_ℓ;

Algorithm 2: Energy-Bounded-Cage-Analysis-2D

V. Experiments

To test our methods, we implemented EBCA-2D in C++ and evaluated the performance on a set of polygonal objects under a gravitational potential energy field. We used the CGAL library [10] to compute triangulations and α-shapes. For GPD computation we performed a convex decomposition of polygons using the algorithm of Lien et al. [24] and libccd [18] for the GJK-EPA algorithm. All experiments ran on an Intel Core i7-4770K 350 GHz processor with 6 cores.

A. Energy-Bounded Cages Under Gravity

We ran EBCA-2D with $s = 200,000$ pose samples for varying obstacle configurations on a set of six polygonal parts. The parts were created by projecting 3D models from the YCB dataset [7] and 3DNet [46] onto a plane and triangulating the projection. We computed the mass M for each object using a uniform mass density of $0.01 kg/cm^2$. Each run of EBCA-2D took approximately 180 seconds, and more details on runtime can be found in Section V-B.

Fig. 6 shows the estimated normalized minimum escape energy $\hat{u}_n = \hat{u}/(Mg)$, or distance that the center of mass must rise to escape, for three parallel-jaw gripper configurations on each of three objects. To aid in visualization, we used RRT* implemented in OMPL [38] to plan an escape path to directly below the initial object pose, and we rendered the object in the pose along the solution path with maximum energy. The ranking of grasps by $\hat{u}_n$ matches our intuition, and appears to also match the ranking of grasps by the maximum energy reached along the RRT* visualization path. To evaluate the lower bound of Theorem 4.4, we used RRT* to attempt to plan an object escape path over the set of collision-free poses with energy less than $\hat{u}$. We also ran dynamic simulations of object motions under gravity and Gaussian force and torque perturbations in Box2D [1] and checked if the object ever escaped with energy less than $\hat{u}$. For every configuration, the RRT* planner was not able to find an escape path with energy less than $\hat{u}$ within 120 seconds and in Box2D the object never violated $\hat{u}$ over 1,000 trials.

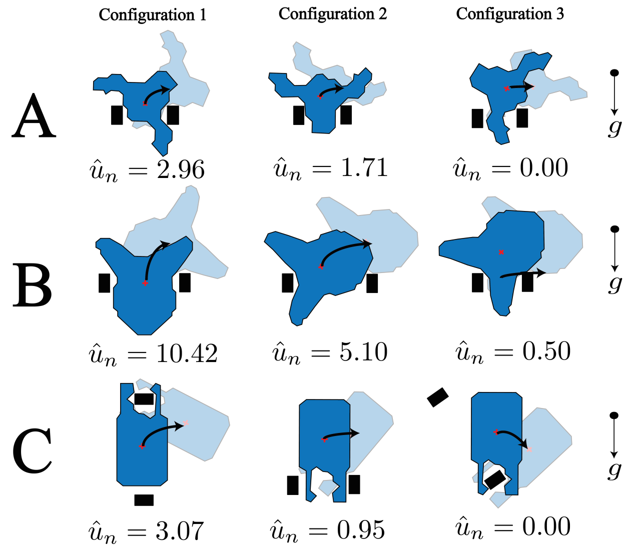

Fig. 6: Three example polygonal parts (blue) with three parallel-jaw configurations (black) for each object. Below each configuration is normalized minimum escape energy $\hat{u}_n = \hat{u}/Mg$ estimated by EBCA-2D with $s = 200,000$ pose samples under gravity, where M is the mass of the part. To visualize the output of EBCA-2D, we render the object translucently at the the highest point of an escape path found by an RRT* planner, with an arrow to indicate direction. We see that $\hat{u}_n$, which is the estimated minimum height that must be reached to escape, ranks the configurations for each object in the same order as the maximum height reached along the RRT* escape path.

We also ran our algorithm on a set of configurations with more than two nonconvex obstacles. Fig. 7 displays $\hat{u}_n$ for four examples: capturing an object using a single rectangular jaw and ramp, bowl-shaped jaws pinning an object against a vertical wall, three rectangular jaws, and a robotic gripper on a doorknob inspired by [12]. Our algorithm is able to prove cages for configurations 3 and 4, and the ranking of configurations 1 and 2 by $\hat{u}_n$ matches our intution. Again, both RRT* and Box2D did not generate an escape path with energy less than $\hat{u}$.

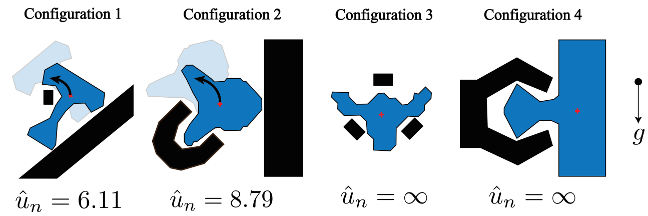

Fig. 7: Four example configurations of polygonal parts (blue) and obstacle configurations (black) with varying shape and number of components. Under each configuration is the normalized minimum escape energy $\hat{u}_n$ estimated by EBCA-2D with $s = 200,000$. We see that EBCA-2D verifies that configurations 3 and 4 are cages, both of which are challenging due to the nonconvexity of the parts and obstacles.

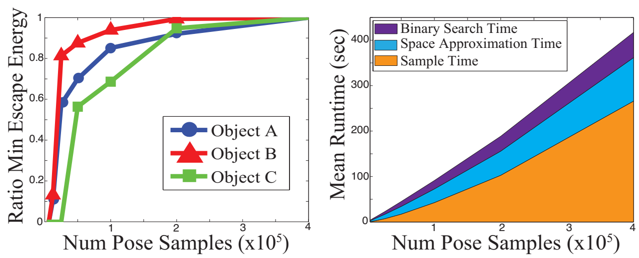

Fig. 8: (Left) The ratio of the minimum escape energy $\hat{u}$ estimated by EBCA-2D for s pose samples to $\hat{u}$ at $s = 400,000$ for $s = \{6.25, 12.5, 25, 50, 100, 200, 400\} \times 10^3$. The values plotted are for configuration 1 of each object in Fig. 6 and are averaged over 5 independent trials per value of s. We see that the object B, the "fattest" object, converges the fastest and object C, the object with the "thinnest" pieces, converges the slowest. (Right) The runtime of EBCA-2D in seconds broken down by component of the algorithm versus the number of pose samples $s = \{6.25, 12.5, 25, 50, 100, 200, 400\} \times 10^3$. Each datapoint is averaged over 5 independent trials per value of s and configuration 1 of the objects in Fig. 6. The scaling of the average runtime is approximately linear in s, and the runtime becomes dominated by the time to generate pose samples for large s.

B. Sensitivity to Number of Pose Samples

We also studied the sensitivity of $\hat{u}$ and the total runtime to the number of pose samples s used to approximate the collision space. The left panel of Fig. 8 shows the ratio of $\hat{u}$ at $s = \{6.25, 12.5, 25, 50, 100, 200, 400\} \times 10^3$ to $\hat{u}$ at $s = 400,000$ for configuration 1 for each of the objects in Fig. 6. Each ratio is averaged over 5 independent trials per value of s to smooth the effects of random initializations. We see that for less than about 25,000 samples the output tends to be $\hat{u} \approx 0$ because the collision space is not well-approximated, leading to "holes" in the algorithm's representation of the collision space for lower y-coordinates. However, as s becomes large, $\hat{u}$ converges towards a nonzero value. Interestingly, object B, which is the "fattest" [43] converges the fastest, taking only about $50,000$ samples to converge to within 90% of its value at $s = 400,000$. On the other hand, object C takes nearly $200,000$ to converge to within 90% of its value at $s = 400,000$. This is possibly because u^* for object C depends on a very thin part of the collision space, which requires more samples to approximate.

The right panel of Fig. 8 shows the scaling of the runtime in seconds versus the number of pose samples s averaged over 5 independent trials for configuration 1 of the objects in Fig. 6. The runtime is broken down by component of the algorithm: pose sampling, approximating the configuration space using α-shapes, and the binary search over energies. We see that the total runtime for these shapes and obstacles is approximately linear in the number of pose samples s, with pose sampling taking the largest portion of the runtime. However, the amount of time to sample poses and the time to construct an approximation to the configuration space both appear to be slightly superlinear in s. These results suggest that runtime remains well below the worst case s^2 scaling in practice.

VI. Discussion and Future Work

We defined energy-bounded caging configurations and the minimum escape energy, or the minimum energy that external perturbations must exert on an object for it to escape a set of obstacles. We also developed Energy-Bounded-Cage-Analysis-2D (EBCA-2D), an algorithm to compute a lower bound on the minimum escape energy for 2D polygonal objects and obstacles using weighted α-shapes. Our experiments demonstrate that we are able to verify cages and suggest that the lower bound from EBCA-2D matches our intuition on a set of nonconvex objects and obstacles.

Future work will explore tighter bounds on the escape energy using optimal planners such as RRT*. We will also investigate extensions of EBCA-2D to synthesize obstacle configurations that form energy-bounded cages and to analyze 3D objects and obstacles. One barrier to using EBCA-2D for synthesis is the runtime for analyzing a single configuration, which is largely dominated by pose sampling. Future work will study parallel cloud-based implementations of sampling and adaptive sampling procedures to bias samples towards thin parts of the collision space such as Gaussian sampling from motion planning [6]. Additionally, we will explore nonzero values of α and their relation to grasp robustness.

While in principle the theory behind our approach can be generalized to 3D, a challenge for synthesizing and analyzing configurations in 3D is the increase in dimensionality of the configuration space from 3D to 6D. This increases the computational load to construct dense α-shapes [15] and may also increase the number of samples needed to approximate the configuration space. Another difficulty is that scaling to 3D would require an embedding of $SE(3)$ into $\mathbb{R}^6$, which is more challenging due to the topology of $SE(3)$ [9] and because no implementation of higher dimensional weighted α-shapes exists in common software such as CGAL [39]. In future work, we will investigate alternative representations of the forbidden space such as Vietoris-Rips complexes [20], a sparser simplicial complex representation of point samples, or precomputed simplicial complexes that cover the configuration space [44].

VII. Acknowledgments

This work is supported in part by the U.S. National Science Foundation under Award IIS-1227536, grants from Google, the Knut and Alice Wallenberg Foundation, and the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program. We thank our colleagues who gave feedback and suggestions: Animesh Garg, David Gealy, James Kuffner, Michael Laskey, Sanjay Krishnan, Stephen McKinley, Jia Pan, Sachin Patil, Nan Tian, and Jur van den Berg.

REFERENCES

[1] "Box2d physics engine," http://box2d.org.

[2] T. F. Allen, J. W. Burdick, and E. Rimon, "Two-finger caging of polygonal objects using contact space search," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1164–1179, 2015.

[3] T. F. Allen, E. Rimon, and J. W. Burdick, "Two-finger caging of 3d polyhedra using contact space search," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2014, pp. 2005–2012.

[4] J. Basch, L. J. Guibas, D. Hsu, and A. T. Nguyen, "Disconnection proofs for motion planning," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, vol. 2. IEEE, 2001, pp. 1765–1772.

[5] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Citeseer, 2000, pp. 348–353.

[6] V. Boor, M. H. Overmars, and A. F. van der Stappen, "The gaussian sampling strategy for probabilistic roadmap planners," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, vol. 2. IEEE, 1999, pp. 1018–1023.

[7] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols," *arXiv preprint arXiv:1502.03143*, 2015.

[8] S. Cameron, "A comparison of two fast algorithms for computing the distance between convex polyhedra," 1996.

[9] M. Caroli and M. Teillaud, "Computing 3d periodic triangulations," in *Algorithms-ESA 2009*. Springer, 2009, pp. 59–70.

[10] T. K. F. Da, S. Loriot, and M. Yvinec, "3D alpha shapes," in *CGAL User and Reference Manual*, 4.6.2 ed. CGAL Editorial Board, 2015. [Online]. Available: http://doc.cgal.org/4.6.2/Manual/packages.html#PkgAlphaShapes3Summary

[11] C. Davidson and A. Blake, "Caging planar objects with a three-finger one-parameter gripper," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 1998, pp. 2722–2727.

[12] R. Diankov, S. S. Srinivasa, D. Ferguson, and J. Kuffner, "Manipulation planning with caging grasps," in *IEEE Conf. on Humanoid Robots*. IEEE, 2008, pp. 285–292.

[13] H. Edelsbrunner, *Weighted alpha shapes*. University of Illinois at Urbana-Champaign, Department of Computer Science, 1992.

[14] ——, "The union of balls and its dual shape," in *Proceedings of the Ninth Annual Symposium on Computational Geometry*. ACM, 1993, pp. 218–231.

[15] H. Edelsbrunner and E. P. Mücke, "Three-dimensional alpha shapes," *ACM Transactions on Graphics (TOG)*, vol. 13, no. 1, pp. 43–72, 1994.

[16] J. Erickson, S. Thite, F. Rothganger, and J. Ponce, "Capturing a convex object with three discs," *IEEE Trans. Robotics*, vol. 23, no. 6, pp. 1133–1140, 2007.

[17] C. Ferrari and J. Canny, "Planning optimal grasps," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 1992, pp. 2290–2295.

[18] D. Fiser, "libccd - collision detection between convex shapes," http://libccd.danfis.cz/.

[19] A. Hatcher, *Algebraic topology*. Cambridge University Press, 2002.

[20] J.-C. Hausmann *et al.*, "On the vietoris-rips complexes and a cohomology theory for metric spaces," *Ann. Math. Studies*, vol. 138, pp. 175–188, 1995.

[21] H. Kruger, E. Rimon, and A. F. van der Stappen, "Local force closure," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 4176–4182.

[22] W. Kuperberg, "Problems on polytopes and convex sets," in *DIMACS Workshop on polytopes*, 1990, pp. 584–589.

[23] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.

[24] J.-M. Lien and N. M. Amato, "Approximate convex decomposition of polygons," in *Proceedings of the Twentieth Annual Symposium on Computational Geometry*. ACM, 2004, pp. 17–26.

[25] S. Liu and S. Carpin, "A fast algorithm for grasp quality evaluation using the object wrench space," 2015.

[26] K. M. Lynch and M. T. Mason, "Dynamic nonprehensile manipulation: Controllability, planning, and experiments," *The International Journal of Robotics Research*, vol. 18, no. 1, pp. 64–92, 1999.

[27] T. Makapunyo, T. Phoka, P. Pipattanasomporn, N. Niparnan, and A. Sudsang, "Measurement framework of partial cage quality based on probabilistic motion planning," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2013, pp. 1574–1579.

[28] Z. McCarthy, T. Bretl, and S. Hutchinson, "Proving path non-existence using sampling and alpha shapes," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 2563–2569.

[29] P. Pipattanasomporn and A. Sudsang, "Two-finger caging of nonconvex polytopes," *IEEE Trans. Robotics*, vol. 27, no. 2, pp. 324–333, 2011.

[30] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer handbook of robotics*. Springer, 2008, pp. 671–700.

[31] E. Rimon and A. Blake, "Caging 2d bodies by 1-parameter two-fingered gripping systems," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 1996, pp. 1458–1464.

[32] ——, "Caging planar bodies by one-parameter two-fingered gripping systems," *Int. J. Robotics Research (IJRR)*, vol. 18, no. 3, pp. 299–318, 1999.

[33] E. Rimon and J. W. Burdick, "Mobility of bodies in contact. ii. how forces are generated by curvature effects," *Robotics and Automation, IEEE Transactions on*, vol. 14, no. 5, pp. 709–717, 1998.

[34] A. Rodriguez and M. T. Mason, "Two finger caging: squeezing and stretching," *Algorithmic Foundation of Robotics VIII*, vol. 57, 2010.

[35] A. Rodriguez, M. T. Mason, and S. Ferry, "From caging to grasping," *Int. J. Robotics Research (IJRR)*, pp. 886–900, 2012.

[36] J. A. Stork, F. T. Pokorny, and D. Kragic, "Integrated motion and clasp planning with virtual linking," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2013, pp. 3007–3014.

[37] J. Su, H. Qiao, Z. Ou, and Z.-Y. Liu, "Vision-based caging grasps of polyhedron-like workpieces with a binary industrial gripper," vol. 12, 2015.

[38] I. A. Şucan, M. Moll, and L. E. Kavraki, "The Open Motion Planning Library," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, December 2012, http://ompl.kavrakilab.org.

[39] The CGAL Project, *CGAL User and Reference Manual*, 4th ed. CGAL Editorial Board, 2015.

[40] M. Vahedi and A. F. van der Stappen, "Caging polygons with two and three fingers," *Int. J. Robotics Research (IJRR)*, vol. 27, no. 11-12, pp. 1308–1324, 2008.

[41] ——, "On the complexity of the set of three-finger caging grasps of convex polygons." in *Proc. Robotics: Science and Systems (RSS)*, 2009.

[42] G. Van Den Bergen, "Proximity queries and penetration depth computation on 3d game objects," in *Game developers conference*, vol. 170, 2001.

[43] A. F. van der Stappen and M. H. Overmars, "Motion planning amidst fat obstacles," in *Proceedings of the Tenth Annual Symposium on Computational Geometry*. ACM, 1994, pp. 31–40.

[44] W. Wan, R. Fukui, M. Shimosaka, T. Sato, and Y. Kuniyoshi, "A new grasping by caging solution by using eigen-shapes and space mapping," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2013, pp. 1566–1573.

[45] ——, "Grasping by caging: A promising tool to deal with uncertainty," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 5142–5149.

[46] W. Wohlkinger, A. Aldoma, R. B. Rusu, and M. Vincze, "3dnet: Large-scale object class recognition from cad models," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 5384–5391.

[47] L. Zhang, Y. J. Kim, and D. Manocha, "Efficient cell labelling and path non-existence computation using c-obstacle query," *Int. J. Robotics Research (IJRR)*, vol. 27, no. 11-12, pp. 1246–1257, 2008.

[48] L. Zhang, Y. J. Kim, G. Varadhan, and D. Manocha, "Generalized penetration depth computation," *Computer-Aided Design*, vol. 39, no. 8, pp. 625–638, 2007.