Caging Polygonal Objects Using Equilateral Three-Finger Hands

Hallel A. Bunis, Elon D. Rimon, Yoav Golan, and Amir Shapiro

***Abstract*—Multifinger caging offers a robust object grasping approach. While efficient computation of two-finger caging grasps is well developed, the computation of three-finger caging grasps has remained a challenging open problem. This letter considers the caging of polygonal objects with three-finger hands which maintain an *equilateral triangle* formation during the grasping process. While the c-space of such hands is four-dimensional (4-D), their *contact space* which represents all two and three finger contacts along the grasped object's boundary forms a 2-D stratified manifold. The letter describes a *caging graph* that can be constructed in the hand's relatively simple contact space. Starting from a desired immobilizing grasp of the object, the caging graph is searched for the largest finger opening that maintains a three-finger cage about the object. This finger opening determines the *caging regions*, and any equilateral finger placement within the caging regions guarantees a robust object grasping. The technique is illustrated with a detailed example and a video showing caging experiments with an equilateral robot hand.**

***Index Terms*—Caging, grasping, grasp planning, multifingered hands.**

I. Introduction

MULTI-FINGER caging offers a robust object grasping approach by robot hands. To securely grasp an object, the fingers are first placed in a *cage* formation around the object, thus preventing it from escaping the hand, then the hand closes until the object is securely immobilized by the fingers. The *caging regions* that surround a desired immobilizing grasp allow huge position uncertainty in the initial placement of the robot fingers, while subsequent closing of the hand does not require any accurate positioning of the fingers relative to the grasped object.

This letter focuses on caging polygonal objects with three-finger hands that maintain an *equilateral triangle* formation during the grasping process. While the configuration space of such hands is four dimensional, the letter describes how the cage formations that surround an immobilizing grasp can be computed in the hand's two dimensional *contact space,* which represents all two and three finger contacts along the grasped object's boundary. The letter describes a *caging graph* that can be constructed in the hand's contact space. Starting from a desired immobilizing grasp of a given polygonal object, the caging graph can be readily searched for the critical grasp which allows the object to escape the hand. The corresponding caging regions which surround the immobilizing grasp can then be computed. Moreover, the entire search process can be intuitively observed as the motion of the fingers along the grasped object's boundary. The principal method of this letter has been recently submitted to WAFR. This letter offers a much improved search algorithm which is accompanied by a detailed three-finger caging synthesis example. The MATLAB files used to implement the search algorithm, and a video showing experiments conducted with a specially designed equilateral robot hand on physical objects, can be found in the accompanying media files.

Since robotic caging was introduced by Rimon and Blake [12], [13], the synthesis of caging grasps initially focused on two-finger hands. Notable examples are the comprehensive two-finger caging algorithms of Vahedi and Van der Stappen [17] and Pipattanasomporn and Sudsang [10], which compute all two-finger cage formations directly in the hand's four-dimensional configuration space. Allen, Rimon and Burdick [1] recently showed how all two-finger cage formations can be alternatively computed in the hand's two-dimensional *contact space,* which offers implementation simplicity and geometric verifiability along the grasped object's boundary. This letter extends the contact space approach to three-finger hands which maintain an equilateral triangle formation.

While two point or disc fingers can immobilize polygonal objects which possess concavities, three point or disc fingers can immobilize and hence securely grasp almost all polygonal objects. This insight motivated several three-finger caging algorithms. The first algorithm by Erickson *et al.* [5] attempted to generate small caging regions localized around three particular edges of the grasped polygon. Subsequent work by Vahedi and Van der Stappen [17] was able to compute all three-finger caging grasps provided that two fingers are held fixed as base fingers. Having three as well as higher number of fingers in mind, Pipattanasomporn and Sudsang [11] proposed to use convex functions of the finger locations, called *dispersion functions,* to compute the maximal finger dispersion that maintains a cage formation as a convex optimization problem. Wan considered the efficient numerical computation of three-finger cage formations with fixed base fingers [19], [20]. Wan recently proposed the topological enumeration of the hand's c-space boundary components, a technique that can efficiently report when a

Manuscript received September 15, 2016; accepted December 22, 2016. Date of publication January 10, 2017; date of current version May 16, 2017. This letter was recommended for publication by Associate Editor Y. Zheng and Editor H. Ding upon evaluation of the reviewers' comments. This work was supported by the Israel Science Foundation under Grant 1253/14.

H. A. Bunis and E. D. Rimon are with the Department of Mechanical Engineering, Technion—Israel Institute of Technology, Haifa 32587, Israel (e-mail: hallelb@technion.ac.il; rimon@technion.ac.il).

Y. Golan and A. Shapiro are with the Department of Mechanical Engineering, Ben-Gurion University of the Negev, Beer Sheva 84105, Israel (e-mail: yoavgo@post.bgu.ac.il; ashapiro@bgu.ac.il).

This letter has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LRA.2017.2651158

candidate three or higher number of fingers formation forms a cage about a given object [18].

The literature survey would be incomplete without mentioning advances in caging theory made by Rodriguez [15] and Allen [2] in the context of three and higher number of fingers cage formations, and the inclusion of gravity in partial cage formations [8]. Caging has also been considered for robust object manipulation [16], [21], as well as the robust transfer of warehouse objects by mobile robots [6], [9].

The three-finger hands considered in this letter are motivated by hand mechanism design. As discussed by Dollar *et al.* [3], [7], roboticists seek general purpose *minimalistic hand designs* that will posses the smallest number of fingers as well as the smallest number of actuators. Three-finger hands form an attractive minimalistic design, since three point or disc fingers can immobilize every polygonal object which does not possess opposing parallel edges [14]. Moreover, by limiting the three-finger hand to a particular one-parameter family of formations such as an equilateral triangle, a *single actuator* can simultaneously close the three fingers, thus allowing for extremely simple hand designs (Fig. 1). While this letter considers equilateral finger formations, we are currently working to extend the contact space approach to other one-parameter triangular formations.

The letter is structured as follows. Section II introduces the three-finger caging problem. Section III provides background on the representation of cage formations in the hand's four-dimensional configuration space. Section IV describes the three-finger hand *contact space*. Section V describes the *caging graph* construction in the hand's contact space. Section VI presents the caging graph search algorithm. Section VII provides a detailed synthesis example of how the caging graph is searched in order to determine the caging regions associated with an immobilizing grasp, and touches on practical considerations. The conclusion discusses using contact space for three-finger hands that maintain more general triangular finger formations.

II. The Three-Finger Caging Problem

Consider the caging of polygonal objects using a robot hand made of a rigid *palm* and three finger mechanisms attached to the palm (Fig. 1). The hand is modeled by a *palm frame*, $\mathcal{F}_B$, that can freely translate and rotate in $\mathbb{R}^2$, and three point or disc fingers that form an equilateral triangle fixed to the hand's palm frame (Fig. 2(a)). The distance between each pair of fingers is specified by a common scalar parameter $\sigma \geq 0$. The hand's configuration is thus specified by the pair (q, σ), where $q = (d, \theta) \in \mathbb{R}^2 \times \mathbb{S}$ is the position and orientation of the palm frame with respect to a fixed world frame, while σ determines the *size* of the equilateral triangle formed by the three fingers.

The object to be caged, denoted $\mathcal{B}$, is assumed to be a rigid polygon that can freely translate and rotate in $\mathbb{R}^2$. When the fingers are placed and held stationary around the object, the object is said to be *immobilized* by the fingers when no motion of $\mathcal{B}$ is possible relative to the stationary fingers. The object $\mathcal{B}$ is said to be *caged* by the fingers when it has some freedom of movement between the stationary fingers, but cannot escape to infinity (Fig. 2(a)). Equivalently, let the object be held stationary and let the three-finger hand move with a fixed inter-finger distance σ. The object is *caged* when the fixed-σ hand cannot move to infinity without one of its fingers penetrating the stationary object.

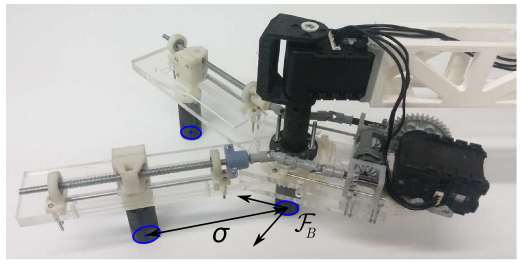

Fig. 1. The experimental system used to validate the caging algorithm (see also video file). The hand's configuration is specified by the palm frame $\mathcal{F}_B$ and the inter-finger distance σ.

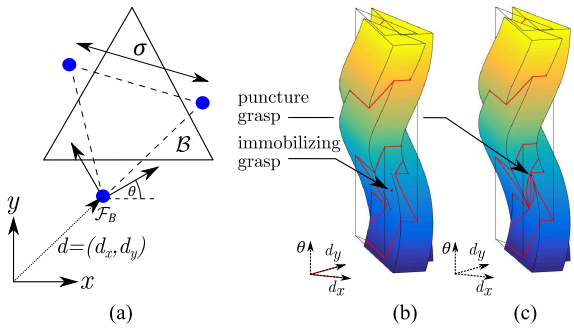

Fig. 2. (a) The three-finger hand is modeled by a palm frame and three point or disc fingers, which maintain an equilateral triangle formation fixed to the palm frame. (b) The c-space slice through $\sigma = \sigma_0$, where the c-obstacle slices completely surround the immobilizing configuration point. (c) The c-space slice through $\sigma = \sigma_1$, where a puncture point appears on the free c-space cavity's wall (best viewed in color).

In general, there are two types of immobilizing grasps: *squeezing grasps* where the fingers press inward on the grasped object, and *stretching grasps* where the fingers press outward from within a cavity of the grasped object. This letter focuses on the more common *squeezing grasps*.

Every immobilizing grasp of $\mathcal{B}$ is associated with *caging regions*. When the fingers are placed in these caging regions such that the inter-finger distance is kept under some critical value, $\sigma < \sigma_{\max}$, the object will be caged by the fingers. Starting at such a cage and monotonically decreasing σ while maintaining the equilateral finger formation will lead to an immobilizing grasp of $\mathcal{B}$, while the object remains caged throughout this process. Hence, given a geometric description of $\mathcal{B}$ and a desired two or three-finger immobilizing grasp of this object, the *three-finger caging problem* seeks to compute the critical caging grasp which allows the object to escape the hand. The value of σ at this grasp, $\sigma_{\max}$, can then be used to determine the caging regions which will form the output of the three-finger caging problem.

III. Representation of Cages in the Hand's Configuration Space

The Free C-Space Boundary: The configuration space or *c-space* of the three-finger hand, denoted $\mathcal{C}$, can be thought of as $(q, \sigma) \in I\!R^4$. When the object $\mathcal{B}$ lies stationary in $I\!R^2$, it forms an obstacle from the hand's perspective. The object induces a *c-space obstacle* in $\mathcal{C}$ for each of the three fingers (Fig. 2(b) and (c)). The union of the c-space obstacles is denoted $\mathcal{CB}$. The hand's *free c-space*, $\mathcal{F}$, is the complement of $\mathcal{CB}$'s interior:

$\mathcal{F} = \mathcal{C} - \mathsf{int}(\mathcal{CB})$, where int denotes set interior. The boundary of $\mathcal{F}$ (which is the same as the boundary of $\mathcal{CB}$) consists of all hand configurations at which one or more fingers touch the boundary of the stationary object $\mathcal{B}$.

The boundary consists of three types of submanifolds: manifolds formed by single-finger contacts, two-finger contacts, and three-finger contacts. The union of the two-finger contact submanifolds together with their intersection curves along the three-finger contact manifolds will be key for the contact space approach for computing cage formations, and is defined next.

Definition 1: The three-finger hand's **contact submanifold**, denoted $\mathcal{S}$, is the union of the submanifolds of the free c-space $\mathcal{F}$ associated with all hand configurations at which at least two fingers contact the stationary object $\mathcal{B}$.

C-Space Representation of Cage Formations: Equilibrium grasps play a key role in the c-space representation of cage formations. Let the hand's palm frame coincide with the fixed world frame. Let x_i denote the i'th finger contact along the object's boundary, and let f_i denote the i'th finger force applied on $\mathcal{B}$ along the object's *inward* normal at x_i, where $i = 1,2,3$. The *wrench* (i.e. force and torque) generated by a finger force f_i acting on $\mathcal{B}$ at x_i is given by $\boldsymbol{w}_i = (f_i, x_i \times f_i) \in I\!R^3$, where the torque component is given by the scalar $x_i \times f_i = x_i^T J f_i$, such that $J = \left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right]$. An object $\mathcal{B}$ is held in an *equilibrium grasp* when the finger contact forces can apply a zero net wrench on $\mathcal{B}$, as stated in the following definition.

Definition 2: A rigid object $\mathcal{B}$ is held via $k \geq 2$ finger contacts in a **feasible equilibrium grasp** (in the absence of external forces) when there exist finger forces satisfying:

$$\lambda_1 \boldsymbol{w}_1 + \cdots + \lambda_k \boldsymbol{w}_k = \vec{0} \tag{1}$$

such that $\lambda_1, \ldots, \lambda_k \geq 0$ are not all zero.

An *immobilizing grasp* of $\mathcal{B}$ is an equilibrium grasp at a hand configuration $(q_0, \sigma_0) \in \mathcal{S}$, at which the point q_0 is completely surrounded by the c-obstacles in the σ_0 slice of $\mathcal{C}$ (Fig. 2(b)). A small increase of σ above σ_0 (i.e. a small opening of the fingers) will cause the c-obstacle slices to move away from each other, forming a small *bounded cavity* in $\mathcal{F}$. This cavity allows the hand's palm frame to locally move in a bounded neighborhood of the immobilizing grasp, while the object is kept stationary. Further increasing σ would cause the cavity to expand until eventually a *puncture point* will appear on its boundary. At this instant, the puncture point might connect the cavity to an adjacent cavity associated with a different immobilizing grasp of $\mathcal{B}$, thus being an *intermediate puncture point*, or can connect to infinity, in which case it represents the *escape puncture point* with critical value $\sigma_{\max}$ (Fig. 2(c)), allowing the hand to escape to infinity. The latter puncture point is denoted (q_1, σ_1), where $\sigma_1 = \sigma_{\max}$. Importantly, every puncture point in $\mathcal{F}$ corresponds to an equilibrium grasp of $\mathcal{B}$ which involves two or three finger contacts [13].

When $\sigma = \sigma_{\max}$, the hand's palm frame moves in the maximal bounded region around the stationary object, and the three fingers move accordingly in three *caging regions* in $I\!R^2$. The union of fixed-σ cavities in $\mathcal{F}$ for $\sigma_0 \leq \sigma \leq \sigma_1$ is termed the *caging set* in $\mathcal{F}$. Thus, given a desired immobilizing grasp (q_0, σ_0), our objective is to compute the *escape puncture grasp* (q_1, σ_1) and then determine the caging regions surrounding the desired immobilizing grasp.

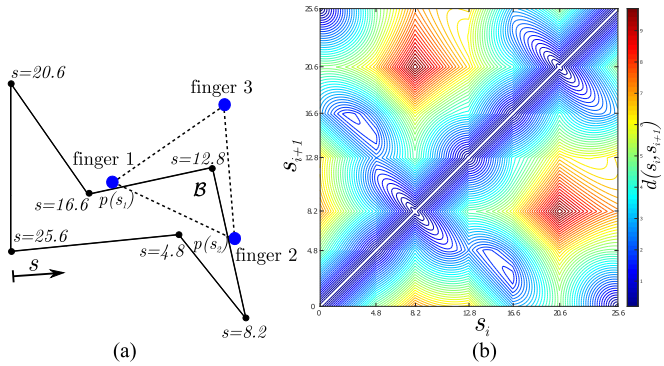

Fig. 3. (a) The s parameterization of the object's boundary. When fingers 1 and 2 contact the object at $p(s_1)$ and $p(s_2)$, the position of finger 3 is uniquely determined. (b) The two-finger contact space $\mathcal{U}_{i,i+1}$, overlaid with the contours of $d(s_i, s_{i+1})$ induced by the object shown in (a). Note the partition of $\mathcal{U}_{i,i+1}$ into rectangles according to the object's edge pairs (best viewed in color).

IV. The Three-Finger Hand Contact Space

The contact submanifold $\mathcal{S}$ consists of the two-finger contact manifolds together with their intersection along the three-finger contact curves in $\mathcal{F}$. The hand's *contact space* is a parametrization of $\mathcal{S}$ in terms of the finger contacts along the object's boundary. Contact space will be used to find an *escape path* which starts at (q_0, σ_0), passes through (q_1, σ_1), and ends at a three-finger *pinching configuration*.

A. The Two-Finger Contact Spaces

Let us assume that the object is grasped along its outer boundary. The outer boundary will be parameterized by arclength in counterclockwise direction using the scalar parameter $s \in [0, L]$, where L is the object's perimeter (Fig. 3(a)). Let the three fingers be labeled in counterclockwise order, and let fingers i and $i+1$ contact the object's boundary at the points $p(s_i)$ and $p(s_{i+1})$, where $i = 1,2,3$ mod 3 (Fig. 3(a)). The definition of the two-finger contact space follows.

Definition 3: Let a polygonal object be contacted by fingers i and $i+1$ at $p(s_i)$ and $p(s_{i+1})$. The **two-finger contact space** is the parameterization of all two-finger contacts along the object's outer boundary, given by the set $\mathcal{U}_{i,i+1} = [0, L] \times [0, L]$ in the (s_i, s_{i+1}) plane.

A three-finger hand has *three* two-finger contact spaces: $\mathcal{U}_{12}$, $\mathcal{U}_{23}$, and $\mathcal{U}_{31}$. Each of these spaces is partitioned into *contact space rectangles*, each associated with a particular pair of object edges (including same-edge pairs). A contact space rectangle is denoted $\mathcal{R}_{jk}$, meaning that fingers i and $i+1$ are contacting edges j and k of $\mathcal{B}$. The *diagonal* $\Delta_{i,i+1} = \{(s_i, s_{i+1}) \in \mathcal{U}_{i,i+1} : s_i = s_{i+1}\}$ represents all two-finger pinching configurations along the object's boundary.

Next we define the inter-finger distance function on the individual two-finger contact spaces.

Definition 4: Let $\mathcal{U}_{i,i+1}$ be the two-finger contact space associated with fingers i and $i+1$. The inter-finger distance function is the scalar valued function $d : \mathcal{U}_{i,i+1} \to \mathbb{R}$ given by $d(s_i, s_{i+1}) = \|p(s_i) - p(s_{i+1})\|$.

The contours of the inter-finger distance function in $\mathcal{U}_{i,i+1}$ are depicted in Fig. 3(b). Note that $d(s_i, s_{i+1})$ is non-negative and continuous on $\mathcal{U}_{i,i+1}$, and attains a global minimum of zero along the diagonal $\Delta_{i,i+1}$. Moreover, $d(s_i, s_{i+1})$ forms a *convex function* in the individual contact space rectangles. Also

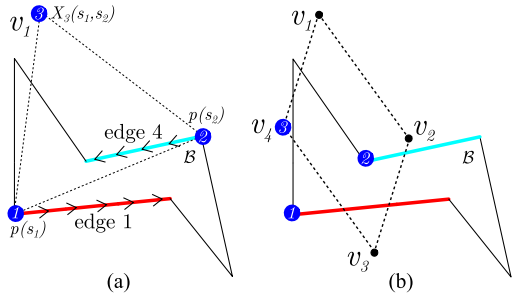

Fig. 4. The third-finger parallelogram construction: (a)–(b) The vertices of the parallelogram v_1, v_2, v_3, v_4 are determined by the position of finger $i+2$, while fingers i and $i+1$ are located at the endpoints of two edges of $\mathcal{B}$.

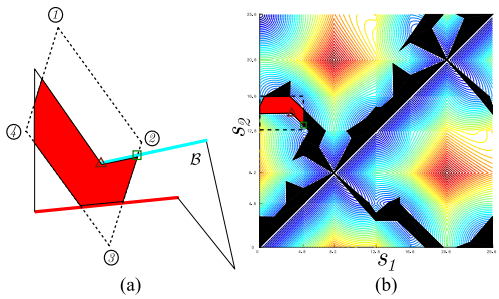

Fig. 5. (a) A parallelogram with non-feasible third finger positions marked by a red region inside $\mathcal{B}$. (b) Applying T^{-1} on the red region gives the contact space obstacle in the rectangle $\mathcal{R}_{1,4}$. The brown triangle and green box mark *corner points* of the contact space obstacle, which lie within $\mathcal{R}_{1,4}$ and on one of its bounding lines. The contact space $\mathcal{U}_{12}$ is overlaid with all contact space obstacles induced by finger 3.

note that the contours of $d(s_i, s_{i+1})$ have an identical layout in $\mathcal{U}_{12}, \mathcal{U}_{23}$ and $\mathcal{U}_{31}$.

B. The Third-Finger Contact Space Obstacles

When fingers i and $i+1$ move along the stationary object's boundary, the third finger must move in a way that maintains the fingers' equilateral formation. Since the third finger may not penetrate the stationary object $\mathcal{B}$ during this motion, *contact space obstacles* are induced in $\mathcal{U}_{i,i+1}$. A graphical technique for constructing these obstacles is next described. The position of finger i along an edge of $\mathcal{B}$ is given by:

$$p(s_i) = p_i^0 + \left(s_i - s_i^0\right) t_i \quad s_i \in \left[s_i^0, s_i^0 + L_i\right]$$

where $p_i^0 = p(s_i^0)$ is the edge's initial vertex, t_i is the edge's *unit tangent,* and L_i is the edge's length. The position of finger $i+1$ along some other edge of $\mathcal{B}$ is similarly described by $p(s_{i+1})$. Let R_ϕ denote a 60° counterclockwise rotation matrix in $\mathbb{R}^2$. Since the hand maintains an equilateral triangle formation, and the finger indices maintain a counterclockwise order, the position of the third finger, denoted X_{i+2}, is given by:

$$X_{i+2}(s_i, s_{i+1}) = p(s_i) + R_\phi(p(s_{i+1}) - p(s_i))$$

Substituting for $p(s_i)$ and $p(s_{i+1})$ gives:

$$X_{i+2}(s_i, s_{i+1}) = u + \left(s_i - s_i^0\right) v + \left(s_{i+1} - s_{i+1}^0\right) w \qquad (2)$$

where $u = p_i^0 + R_\phi\left(p_{i+1}^0 - p_i^0\right)$, $v = (I - R_\phi)\, t_i$, and $w = R_\phi t_{i+1}$ are constant vectors. It can be asserted that $X_{i+2}(s_i, s_{i+1})$ is a parametrization of a *parallelogram* in $\mathbb{R}^2$, with vertices located at u, $u + L_i v$, $u + L_{i+1} w$ and $u + L_i v + L_{i+1} w$. The parallelogram construction is illustrated in Fig. 4.

Note that each parallelogram is associated with a particular contact space rectangle $\mathcal{R}_{jk}$. The parallelograms are next used to determine the following contact space obstacles.

Definition 5: When fingers i and $i+1$ contact the stationary object $\mathcal{B}$, the **contact space obstacles,** denoted $\mathcal{CB}_{i,i+1}$, consist of all points $(s_i, s_{i+1}) \in \mathcal{U}_{i,i+1}$ for which $X_{i+2}(s_i, s_{i+1})$ lies within $\mathcal{B}$.

The parallelogram $X_{i+2}(s_i, s_{i+1})$ can be thought of as a linear mapping applied to the contact space rectangle $\mathcal{R}_{jk}$. By applying the *inverse mapping* on the intersection of the object with the area enclosed by the parallelogram in $\mathbb{R}^2$, one can obtain the contact space obstacles in $\mathcal{R}_{jk}$ (Fig. 5). Denote by T the 2×2 matrix which transforms the contact space basis vectors to v and w. The inverse transformation matrix is given by $T^{-1} = \frac{1}{w^T J v}\left[w\ -v\right]^T J$, where $J = \left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right]$. Fig. 5(b) shows the entire collection of contact space obstacles depicted as black regions in $\mathcal{U}_{12}$, with the obstacles' layout being identical in all three two-finger contact spaces.

The contact space obstacles form polygonal regions as stated in the following lemma.

Lemma 4.1: The contact space obstacles of an equilateral three-finger hand form **polygonal regions** in the two-finger contact spaces $\mathcal{U}_{i,i+1}$, where $i = 1,2,3 \bmod 3$.

Note that every *corner point* of a contact space obstacle corresponds to a hand configuration at which at least one finger is contacting a vertex of $\mathcal{B}$. This fact can be asserted by inspecting Fig. 5.

Finally consider how the two-finger contact spaces relate to each other. The boundary curves of the contact space obstacles correspond to configurations at which all three fingers contact the object. These curves can be considered to be *gluing seams* which connect the three two-finger contact spaces to each other. Following is a formal definition of the hand's contact space, using standard quotient space notation from topology.

Definition 6: The **three-finger hand contact space,** denoted $\mathcal{U}$, is given by the gluing of the individual two-finger contact spaces $\mathcal{U}_{12}$, $\mathcal{U}_{23}$, and $\mathcal{U}_{31}$ along the contact space obstacle boundaries:

$$\mathcal{U} = \mathcal{U}_{12} \cup \mathcal{U}_{23} \cup \mathcal{U}_{31} / bdy\left(\mathcal{CB}_{i,i+1}\right) \sim bdy\left(\mathcal{CB}_{i+1,i+2}\right)$$

where $bdy\left(\mathcal{CB}_{i,i+1}\right)$ is the contact space obstacle boundary, and the quotient identifies the corresponding point triplets on $bdy\left(\mathcal{CB}_{12}\right)$, $bdy\left(\mathcal{CB}_{23}\right)$ and $bdy(\mathcal{CB}_{31})$.

The following proposition states that the hand's contact space $\mathcal{U}$ is topologically equivalent to the contact submanifold $\mathcal{S}$ in the hand's free c-space $\mathcal{F}$ [4].

Proposition 4.2: The three-finger hand's contact space $\mathcal{U}$ is **topologically equivalent** (homeomorphic) to the contact submanifold $\mathcal{S}$ in the hand's free c-space $\mathcal{F}$.

A search for an escape path in $\mathcal{S}$ can therefore be performed in the hand's contact space $\mathcal{U}$.

C. Contact Space Representation of Immobilizing, Puncture and Pinching Grasps

The points in $\mathcal{U}$ that correspond to immobilizing, puncture, and pinching grasps will become nodes of the caging graph. The following proposition characterizes these grasps in contact space $\mathcal{U}$ [4].

Proposition 4.3: The immobilizing and puncture grasps of $\mathcal{B}$ appear respectively as **local minima** and **saddle points** of the inter-finger distance function $d(s_i, s_{i+1})$ in $\mathcal{U}$.

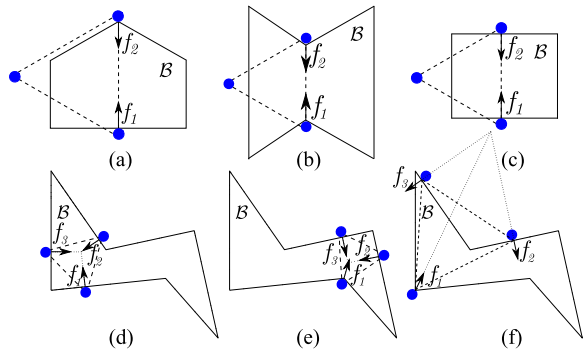

Fig. 6. Equilibrium grasps of $\mathcal{B}$: (a) A grasp at a vertex and an opposing edge. (b) A grasp at two opposing vertices. (c) A grasp at parallel edges. (d) An immobilizing grasp at interior points of edges of $\mathcal{B}$. (e) An immobilizing grasp where one finger contacts a vertex of $\mathcal{B}$. (f) A 3-finger puncture grasp.

To identify the extremum points of $d(s_i, s_{i+1})$ in $\mathcal{U}$, recall that immobilizing and puncture grasps are equilibrium grasps. The two-finger equilibrium grasps occur in three cases (Fig. 6(a)–(c)): (1) One finger contacts a vertex and the other contacts an opposing edge of $\mathcal{B}$. (2) The two fingers contact opposing vertices of $\mathcal{B}$. (3) The two fingers contact opposing parallel edges of $\mathcal{B}$. This leads to the following lemma.

Lemma 4.4: A two-finger equilibrium grasp of $\mathcal{B}$ is an extremum point of $d(s_i, s_{i+1})$ which lies at a **corner point** of $\mathcal{R}_{jk}$, or at an **interior point** of a bounding line of $\mathcal{R}_{jk}$ in $\mathcal{U}$.

The three-finger equilibrium grasps correspond to extremum points of $d(s_i, s_{i+1})$ along the *boundary* of the contact space obstacles. Since $d(s_i, s_{i+1})$ is convex in each contact space rectangle $\mathcal{R}_{jk}$, it can have at most one extremum point in the interior of each line segment along a contact space obstacle boundary, leading to the following lemma [4].

Lemma 4.5: A three-finger equilibrium grasp of $\mathcal{B}$ is an extremum point of $d(s_i, s_{i+1})$ which lies at a **corner point** of a contact space obstacle in $\mathcal{U}$, or at a single point in the interior of each line segment along the boundary of the contact space obstacle in $\mathcal{U}$.

Three-finger equilibrium grasps are illustrated in Fig. 6(d)–(f). Note that all of these extremum points will become nodes of the caging graph. Finally, let us characterize the pinching configurations in $\mathcal{U}$.

Definition 7: The **escape points** in $\mathcal{U}$ are all points on the three-finger diagonal Δ, which correspond to three-finger pinching configurations along the object's boundary.

Since the fingers maintain an equilateral formation, every two-finger pinching configuration automatically forms a three-finger pinching configuration. *Escape nodes* are located on any of the two-finger diagonals. At escape nodes the three fingers are pinched together and can escape to infinity.

V. The Three-Finger Caging Graph

This section describes the *caging graph*, which allows a search for the escape path in the hand's contact space $\mathcal{U}$. The construction of the caging graph will be described in two stages: (1) The caging graph's nodes and edges which are embedded in $\mathcal{U}$. (2) The caging graph's *tunnel edges*, which ensure equivalence between the caging graph and the hand's free c-space $\mathcal{F}$.

A. The Naive Caging Graph

Let $G(V, E)$ denote the *caging graph*, defined as follows.

Definition 8: The **nodes** of G are the corner points of each contact space rectangle $\mathcal{R}_{jk}$, the extremum points of $d(s_i, s_{i+1})$ along the rectangle's bounding lines, the corner points of the contact space obstacles in $\mathcal{R}_{jk}$, and all extremum points of $d(s_i, s_{i+1})$ along the contact space obstacles' boundaries in $\mathcal{U}$. The **edges** of G are all linear segments that connect pairs of nodes in each $\mathcal{R}_{jk}$ without intersecting any contact space obstacle.

By construction, all the extremum points of the inter-finger distance function $d(s_i, s_{i+1})$ are nodes of the caging graph. Such extremum points represent the collection of all feasible and non-feasible equilibrium grasps of the object $\mathcal{B}$. Note that the nodes of G also include escape nodes located at corners of $\mathcal{R}_{jk}$, such that $j = k$.

Computation of the caging graph nodes: First consider the computation of the two-finger nodes. The nodes at the *corners* of a contact space rectangle $\mathcal{R}_{jk}$ correspond to hand configurations at which two fingers contact vertices of $\mathcal{B}$. The nodes at the *extremum points* of $d(s_i, s_{i+1})$ along the bounding lines of $\mathcal{R}_{jk}$ can be computed by evaluating $\nabla d(s_i, s_{i+1}) = \vec{0}$ along each bounding line. A two-finger node represents a *feasible* grasp only if the position of the non-contacting third finger, $X_{i+2}(s_i, s_{i+1})$, at this node lies *outside* the object $\mathcal{B}$. Using a simple $O(n)$ test if a given point lies in a polygon, the computation of all two-finger nodes can be performed in $O(n^3)$ steps, where n in the number of $\mathcal{B}$'s edges.

Next consider the computation of the three-finger nodes, at which all three fingers contact $\mathcal{B}$ while maintaining an equilateral formation. Nodes at the *corner points* of the contact space obstacles correspond to hand configurations at which at least one finger contacts a vertex of $\mathcal{B}$. Nodes at the *extremum points* of $d(s_i, s_{i+1})$ along the contact space obstacle boundaries correspond to three-finger equilibrium grasps of $\mathcal{B}$, and can be computed using (1). The computation of three-finger nodes also takes $O(n^3)$ steps.

A pseudo-code for computing the graph nodes is summarized in [4]. Fig. 8 shows a two-finger contact space with the four types of nodes of G embedded in it. Next consider the construction of the edges of G. The following lemma will be used for the *node-connecting test*, which checks whether two nodes in $\mathcal{R}_{jk}$ can be connected by a feasible graph edge.

Lemma 5.1: Let fingers i and $i+1$ contact the stationary object $\mathcal{B}$. A **line segment** in the two-finger contact space rectangle $\mathcal{R}_{jk}$ in $\mathcal{U}_{i,i+1}$ corresponds to a **linear path** traversed by finger $i+2$ in $\mathbb{R}^2$ while fingers i and $i+1$ move along edges of $\mathcal{B}$.

Lemma 5.1 can be verified by restricting (2) to a line segment in $\mathcal{R}_{jk}$. The node-connecting test is demonstrated in Fig. 7. The test is performed on all possible node pairs in each $\mathcal{R}_{jk}$. Fig. 8 shows a two-finger contact space with the edges of G represented by black line segments. The caging graph $G(V, E)$ is next examined in relation to the topology of the inter-finger distance function $d(s_i, s_{i+1})$ in $\mathcal{U}$. Let us introduce the notion of *sublevel subgraph* of G. The notation $d(s_i(\boldsymbol{v}), s_{i+1}(\boldsymbol{v}))$ will denote the inter-finger distance at the point in $\mathcal{U}_{i,i+1}$ represented by the node $\boldsymbol{v}$.

Definition 9: Let $G(V, E)$ be the caging graph over contact space $\mathcal{U}$. For each $c \geq 0$, a **c-sublevel subgraph** of G, denoted $G_c(V_c, E_c)$, is a subgraph of G whose nodes are given by $V_c = \{\boldsymbol{v} \in V : d(s_i(\boldsymbol{v}), s_{i+1}(\boldsymbol{v})) \leq c\}$.

The following theorem asserts that the caging graph G topologically captures the sublevel structure of $\mathcal{U}$.

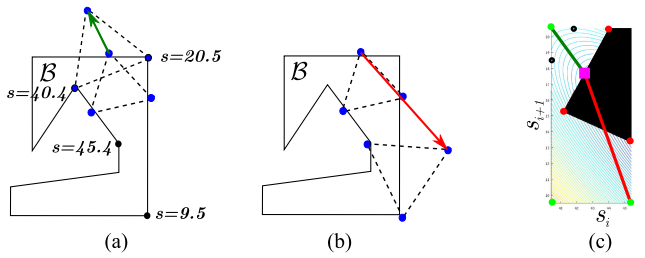

Fig. 7. Performing the node-connecting test on two pairs of graph nodes in $\mathcal{U}_{i,i+1}$: (a) The green line connecting the positions of finger $i+2$ does *not* intersect the object $\mathcal{B}$, hence the corresponding graph nodes can be connected by a graph edge. (b) The red line connecting the positions of finger $i+2$ intersects the object $\mathcal{B}$, hence the corresponding graph nodes cannot be connected. (c) The green and red lines shown in $\mathcal{U}_{i,i+1}$.

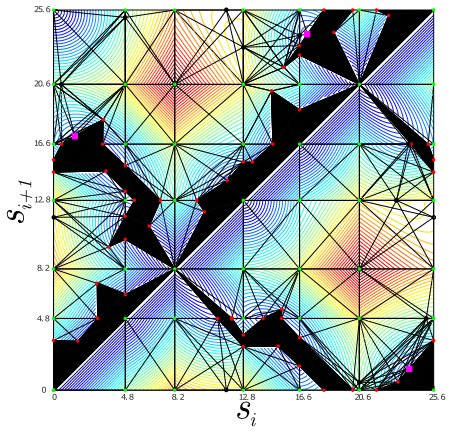

Fig. 8. The two-finger contact space $\mathcal{U}_{i,i+1}$ overlaid with the nodes and edges of the caging graph G, for $i = 1, 2, 3 \bmod 3$.

Theorem 1: The caging graph G **captures the sublevel structure** of $\mathcal{U}$: there exists a contact space path in $\mathcal{U}$ between two nodes of G, $\boldsymbol{v_1}$ and $\boldsymbol{v_2}$, lying entirely in a c-sublevel set of $d(s_i, s_{i+1})$ in $\mathcal{U}$ if and only if there exists a caging graph path between $\boldsymbol{v_1}$ and $\boldsymbol{v_2}$ lying entirely in a c-sublevel subgraph of G.

The proof appears in [4] and is based on the *convexity* of $d(s_i, s_{i+1})$ in each $\mathcal{R}_{jk}$. Given a *contact space path* between two nodes of G, the idea is to partition the path into segments according to the contact space rectangles $\mathcal{R}_{jk}$ it passes through, and then show that each path segment has an equivalent piecewise linear path which passes between nodes of G and lies in the same sublevel set of $d(s_i, s_{i+1})$ in $\mathcal{U}$. Next, a *caging graph path* is considered between two nodes in a sublevel subgraph of G, and the contact space path corresponding to the graph edges in each $\mathcal{R}_{jk}$ is shown to lie in the same sublevel set of $d(s_i, s_{i+1})$ in $\mathcal{U}$.

B. The Augmented Caging Graph

A search for an escape path in G is valid only if it yields the same result as a search in the hand's full free c-space $\mathcal{F}$. Since the caging graph G is sublevel equivalent to $\mathcal{U}$, and $\mathcal{U}$ is homeomorphic to $\mathcal{S}$, to ensure the same result, *sublevel equivalence* between $\mathcal{S}$ and the ambient $\mathcal{F}$ in terms of connectivity must be established. Let $\mathcal{F}_c = \{(q, \sigma) \in \mathcal{F} : \sigma \leq c\}$ denote the sublevel set of free configurations in the hand's free c-space $\mathcal{F}$, and let $\mathcal{S}_c$ denote the subset $\mathcal{F}_c \cap \mathcal{S}$. Sublevel equivalence between $\mathcal{S}$ and $\mathcal{F}$ is defined next.

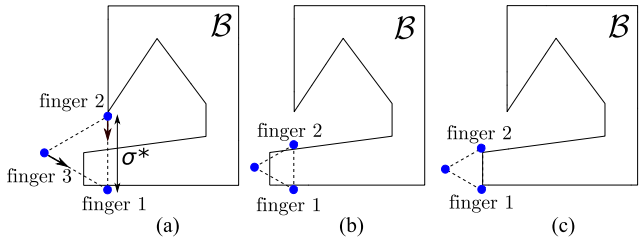

Fig. 9. Caging graph augmentation procedure: (a) A hand configuration represented by the non-feasible local minimum node $\boldsymbol{v_1}$ in $\mathcal{U}$, with $\sigma(\boldsymbol{v_1}) = \sigma^*$. (b) While finger 1 maintains contact and the hand orientation remains fixed, σ is decreased until finger 2 contacts the object. This hand configuration corresponds to a new point u in a contact space rectangle $\mathcal{R}_{jk}$. (c) A node $\boldsymbol{v_2}$ in $\mathcal{R}_{jk}$ which can be connected to u by a feasible line segment and satisfies $\sigma(\boldsymbol{v_2}) \leq \sigma^*$ is found, and the tunnel edge $(\boldsymbol{v_1}, \boldsymbol{v_2})$ is added to $\overline{E}$.

Definition 10: The contact submanifold $\mathcal{S}$ and the ambient free c-space $\mathcal{F}$ are **sublevel equivalent** in terms of connectivity, if in each connected component of $\mathcal{F}_c$ the subset $\mathcal{S}_c$ is also connected, for any $c > 0$.

Consider the function $\pi(q, \sigma) = \sigma$, which maps the hand's configuration (q, σ) to the inter-finger distance σ. Sublevel equivalence between $\mathcal{S}$ and $\mathcal{F}$ breaks when *local minima* of the restriction of $\pi(q, \sigma)$ to $\mathcal{S}$ are *not* local minima of $\pi(q, \sigma)$ in $\mathcal{F}$. These local minima correspond to grasps at which the hand can close without penetrating the object, termed *non-feasible* grasps. Since $\mathcal{S}$ and $\mathcal{U}$ are topologically equivalent, such configurations correspond to *non-feasible* minima of $d(s_i, s_{i+1})$ in $\mathcal{U}$. To ensure sublevel equivalence of $\mathcal{S}$ with $\mathcal{F}$, a graph node at a non-feasible local minimum of $d(s_i, s_{i+1})$ in $\mathcal{U}$ must be connected to a node in a lower sublevel subgraph of G by a *tunnel edge*, which represents a σ-decreasing path termed a *tunnel curve* in $\mathcal{F}$. The caging graph G is *augmented* with tunnel edges, and the complete graph is denoted $G(V, \overline{E})$. The procedure for augmenting G with tunnel edges is demonstrated in Fig. 9. Computing all edges of the complete caging graph requires $O(n^4 \log n)$ computational step, where n is the number of $\mathcal{B}$'s edges.

VI. The Caging Graph Search Algorithm

The augmented caging graph G is sublevel equivalent to the free c-space $\mathcal{F}$. Hence, a search for an escape path along the edges of G will give the same result as a search in $\mathcal{F}$. The caging problem, i.e. the search for an escape path, is thus reduced to a simple graph search. The search is based on the following intuitive notion. The inter-finger distance function $d(s_i, s_{i+1})$ can be considered as a potential energy function, where opening the hand while at least two fingers maintain contact with the object requires an increase of energy. The caging algorithm can be considered as a search for a path which connects one of the potential energy function's local minima (the initial immobilizing grasp) to its global minimum set Δ (comprised of all pinching configurations), while passing through the saddle point which requires the least increase in energy (the escape puncture grasp).

Rather than search the full caging graph, the *incremental* algorithm which is summarized in Algorithm 1 requires as input only the *node set*, and the node corresponding to the initial immobilizing grasp. The incremental search algorithm constructs the edge set using two lists: the *open list* $\mathcal{O}$ which holds the explored nodes, and the *closed list* $\mathcal{C}$ which holds the escape path nodes. The node in $\mathcal{C}$ with the maximal σ value, denoted

Algorithm 1: Incremental Caging Graph Search Algorithm.

The graph node of the initial immobilizing grasp is denoted $\boldsymbol{v}$.
Data Structures: open list $\mathcal{O}$, closed list $\mathcal{C}$, edge set $\overline{E}$, boolean matrix of explored contact space rectangles $\mathcal{E}_{n\times n\times 3}$.
Input: node set V, $\boldsymbol{v}$.
Initialize: $\mathcal{O}=\emptyset$, $\mathcal{C}=\emptyset$, $\overline{E}=\emptyset$, $\mathcal{E}=\mathbf{0}$, current node $\boldsymbol{n}=\boldsymbol{v}$.
Mark node $\boldsymbol{n}$ as explored.
while $\sigma(\boldsymbol{n})\neq 0$ **do**
  Find the unexplored rectangles $\mathcal{R}_{jk}$ containing $\boldsymbol{n}$.
  for each $\mathcal{R}_{jk}$ **do**
    Set $\mathcal{E}(j,k,\mathcal{U}_{i,i+1})=1$
    Compute all feasible graph edges between the nodes in $\mathcal{R}_{jk}$ using the line-connecting test and add them to $\overline{E}$.
  end for
  If $\boldsymbol{n}$ is a non-feasible local minimum of $d(s_i,s_{i+1})$ in $\mathcal{U}$, compute the relevant tunnel edge and add it to $\overline{E}$.
  Add the unexplored neighbors of $\boldsymbol{n}$ to $\mathcal{O}$ and mark them as explored.
  Transfer $\boldsymbol{n}$ to the end of $\mathcal{C}$.
  Sort the nodes in $\mathcal{O}$ by ascending values of σ.
  Set the node with minimal σ value in $\mathcal{O}$ as $\boldsymbol{n}$.
end while ▷ $\boldsymbol{n}$ is now an escape node
Transfer $\boldsymbol{n}$ to the end of $\mathcal{C}$.
Return: closed list $\mathcal{C}$

$\sigma_{\max}$, corresponds to the *escape puncture grasp*. Beyond $\sigma_{\max}$ the cage surrounding the object is broken and the object can escape. The caging algorithm requires at most $O\left(n^4\log n\right)$ computational steps.

VII. Three-Finger Caging Synthesis Example

The caging algorithm will be demonstrated on the rather complex object depicted in Fig. 10(a), which shows the object in an initial two-finger immobilizing grasp (q_0,σ_0). The objective is to compute the associated escape puncture grasp (q_1,σ_1). Fig. 11 shows the relevant regions of the two-finger contact spaces $\mathcal{U}_{12}$, $\mathcal{U}_{23}$ and $\mathcal{U}_{31}$, and Fig. 11(b) shows that most of contact space $\mathcal{U}$ is *irrelevant* for the search. While the complete caging graph has 5350 edges, only 216 edges were computed using the incremental search algorithm. After the search has ended, the nodes that remained in the open list $\mathcal{O}$ were colored in cyan, while the nodes in the closed list $\mathcal{C}$ were colored in magenta and other colors.

Fig. 12(a) shows the inter-finger distance at the nodes along the escape path computed by the search algorithm. Six of these nodes are of interest, and their colors are marked on Fig. 10(a)–(f) to identify the corresponding physical grasps. The green dashed line in Fig. 11(a) represents a tunnel edge between the blue node, which is at a non-feasible local minimum of $d\left(s_i,s_{i+1}\right)$ in $\mathcal{U}$, and the orange node. The light green and orange nodes are minima of $d\left(s_i,s_{i+1}\right)$ in $\mathcal{U}$. These nodes correspond to the initial and intermediate immobilizing grasps. The gray and dark green nodes are saddle points of $d\left(s_i,s_{i+1}\right)$ in $\mathcal{U}$. These nodes correspond to the intermediate puncture grasp and the *escape puncture grasp* (q_1,σ_1).

Sublevel equivalence between the caging graph G and $\mathcal{F}$ is crucial. The correct escape puncture node is the dark green node. If the caging graph is not augmented with the tunnel edge, the escape path would have to pass through the red circled node, which corresponds to the grasp in Fig. 10(g). This grasp would

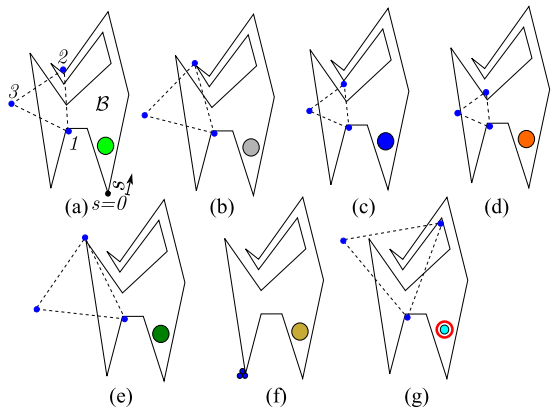

Fig. 10. Physical grasps of the example (see text): (a) An initial two-finger immobilizing grasp. (b) an intermediate puncture grasp. (c) A non-feasible equilibrium grasp. (d) A three-finger immobilizing grasp. (e) The escape puncture grasp. (f) A pinching grasp. (g) A false escape puncture grasp.

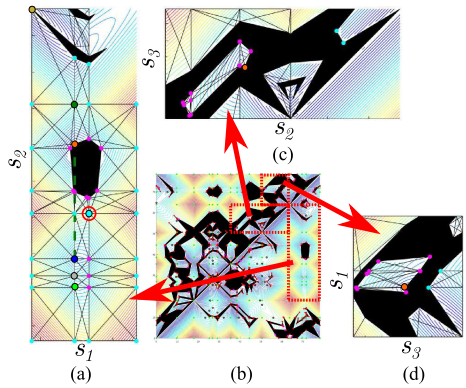

Fig. 11. Relevant part of $\mathcal{U}$ for the example (see text): (a) Part of $\mathcal{U}_{12}$ - the open list nodes were colored in cyan and the closed list nodes were colored in magenta and other colors. The green dashed line represents a tunnel edge. (b) The two-finger contact space $\mathcal{U}_{i,i+1}$ overlaid with the nodes of G. (c)-(d) Relevant parts of the two-finger contact spaces $\mathcal{U}_{23}$ and $\mathcal{U}_{31}$.

then be incorrectly identified as the escape puncture grasp, since the inter-finger distance at this grasp is greater than the distance at the true escape puncture grasp.

Caging regions: Once the escape puncture grasp has been computed, the caging regions can be rendered in $\mathbb{R}^2$. For an equilateral finger formation with a specific finger ordering, two caging regions are sufficient to define all caging configurations. In the example, (q_1,σ_1) corresponds to the escape puncture grasp in Fig. 10(e), at which two fingers contact $\mathcal{B}$, so the two caging regions were rendered for these two fingers.

When the palm frame $\mathcal{F}_B$ coincides with a finger, the boundaries of the caging regions are the *silhouette curves* of the orthogonal projection of the cavities in the hand's free c-space $\mathcal{F}$ of the σ_1 slice, onto the (d_x,d_y) plane. Note that at the hand configurations which generate the silhouette curves, the finger forces generate zero net torque about $\mathcal{F}_B$ [4]. Hence, to render the boundaries of the caging regions, set $\sigma=\sigma_1$, place $\mathcal{F}_B$ at one of the three fingers, and mark its location at configurations where at least one finger is in contact with the object and the contact forces can generate a zero net torque about $\mathcal{F}_B$. Fig. 12(b) shows the two caging regions associated with the initial two-finger immobilizing grasp (q_0,σ_0). Placing fingers 1 and 2 in their respective caging regions while restricting the inter-finger distance to $\sigma<\sigma_1$ will guarantee that the object remains caged

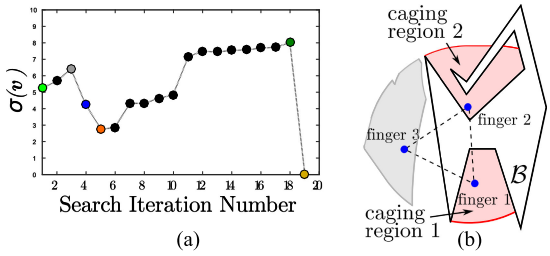

Fig. 12. (a) A graph of σ at the nodes of the closed list $\mathcal{C}$ computed for the example (see text). The colored nodes correspond to the color code in Fig. 10, and the maximal value $\sigma = \sigma_1$ is at the dark green node. (b) $\mathcal{B}$ is caged when fingers 1 and 2 are placed in their respective caging regions and $\sigma < \sigma_1$. The gray region represents all possible positions of finger 3 when $\sigma = \sigma_1$.

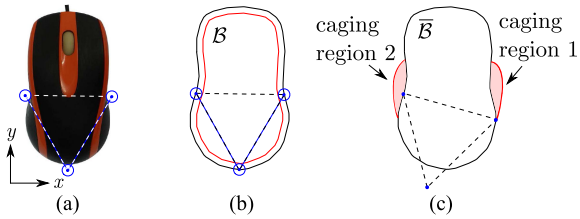

Fig. 13. (a) A computer mouse immobilized by three disc fingers. (b) The red object $\mathcal{B}$ is the projection of the mouse onto the (x, y) plane. A polygonal approximation of $\mathcal{B}$ expanded by the finger radius, denoted $\overline{\mathcal{B}}$, is depicted in black. (c) The escape puncture grasp and the two caging regions associated with the immobilizing grasp of $\overline{\mathcal{B}}$.

by the equilateral three-finger hand, thus allowing for a huge uncertainty in finger placement.

Practical considerations: Although the caging graph approach of this letter assumes polygonal objects and frictionless point fingers, with some adjustments it can be used for *any* planar object $\mathcal{B}$ and *disc* fingers of equal radii. First, the object must be expanded by the finger radius. Next, the expanded object must be approximated by a polygon, with a user specified error tolerance. The resulting polygon, denoted $\overline{\mathcal{B}}$, serves as the grasped object, while the finger center points serve as point fingers (Fig. 13). The search algorithm requires $O\left(n^4 \log n\right)$ computational steps, where n is the number of $\overline{\mathcal{B}}$'s edges. Hence, the smaller the error tolerance, the longer it will take to compute the escape puncture grasp.

VIII. Conclusion

The letter considered the caging of polygonal objects by three-finger hands which maintain equilateral triangle formations. While the hand's configuration space is four dimensional, the letter presented a caging algorithm that can be implemented in the hand's *contact space* $\mathcal{U}$, which can be thought of as $\mathbb{R}^2$. The hand's contact space $\mathcal{U}$ contains obstacles which represent forbidden third-finger positions when two fingers contact the grasped object's boundary. The letter defined a *caging graph*, G, whose nodes include all the extremum points of the inter-finger distance function $d(s_i, s_{i+1})$ in $\mathcal{U}$, as well as all corner points of the contact space obstacles. The caging graph edges consist of line segments that connect adjacent nodes in $\mathcal{U}$, as well as tunnel edges which connect local minima of $d(s_i, s_{i+1})$ that represent non-feasible equilibrium grasps to neighboring nodes of G. The caging graph is sublevel equivalent to the hand's free c-space $\mathcal{F}$. Hence, starting from an immobilizing grasp, the caging graph can be readily searched for the intermediate puncture grasps as well as the final puncture grasp through which the object can escape to infinity. Moreover, the search process can be intuitively observed as the motion of the fingers along the grasped object's boundary. Finally, the escape grasp computed by the search algorithm allows the depiction of the caging regions surrounding the desired immobilizing grasp.

We are currently working on extending the contact space approach to general triangular finger formations. Such formations will still possess the sought after mechanical simplicity, and their ability to securely grasp every polygonal object without opposing parallel edges justifies the development of efficient tools for synthesizing robust grasp sequences for such hands.

References

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

[2] T. F. Allen, E. Rimon, and J. W. Burdick, "Robust three finger three-parameter caging of convex polygons," in *Proc. Int. Conf. Robot. Autom*, 2015, pp. 4318–4325.

[3] S. B. Backus and A. M. Dollar, "An adaptive three-fingered prismatic gripper with passive rotational joints," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 668–675, Jul. 2016.

[4] H. A. Bunis and E. D. Rimon, "Caging polygonal objects using equilateral three-finger hands," Tech. Rep., Dept. of Mech. Eng., Technion, Haifa, Israel, 2016. [Online]. Available: http://robots.technion.ac.il/publications.htm

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

[6] J. Fink, M. A. Hsieh, and V. Kumar, "Multi-robot manipulation via caging in environments with obstacles," *Proc. Int. Conf. Robot. Autom.*, 2008, pp. 1471–1476.

[7] R. D. Howe, A. M. Dollar, and M. Claffee, "Robots get a grip," *IEEE Spectr.*, vol. 51, no. 12, pp. 42–47, Dec. 2014.

[8] J. Mahler, F. T. Pokorny, Z. McCarthy, A. F. van der Stappen, and K. Goldberg, "Energy-bounded caging: Formal definition and 2-D energy lower bound algorithm based on weighted alpha shapes," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 508–515, Jan. 2016.

[9] G. A. S. Pereira, M. F. M. Campos, and V. Kumar, "Decentralized algorithms for multi-robot manipulation via caging," *Int. J. Robot. Res.*, vol. 23, no. 7/8, pp. 783–795, 2004.

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

[11] P. Pipattanasomporn, P. Vongmasa, and A. Sudsang, "Caging rigid polytopes via finger dispersion control," *Proc. Int. Conf. Robot. Autom.*, 2008, pp. 1181–1186.

[12] E. Rimon and A. Blake, "Caging 2D bodies by 1-parameter two-fingered gripping systems," *Proc. Int. Conf. Robot. Autom.*, pp. 1458–1464, 1996.

[13] E. Rimon and A. Blake, "Caging planar bodies by 1-parameter two-fingered gripping systems," *Int. J. Robot. Res.*, vol. 18, no. 3, pp. 299–318, 1999.

[14] E. Rimon and J. W. Burdick, "New bounds on the number of frictionless fingers required to immobilize planar objects," *J. Robot. Syst.*, vol. 12, no. 6, pp. 433–451, 1995.

[15] A. Rodriguez, M. T. Mason, and S. Ferry, "From caging to grasping," *Int. J. Robot. Res.*, vol. 31, no. 7, pp. 886–900, 2012.

[16] A. Sudsang, F. Rothganger, and J. Ponce, "Motion planning for disc-shaped robots pushing a polygonal object in the plane," *IEEE Trans. Robot. Autom.*, vol. 18, no. 4, pp. 550–562, Aug. 2002.

[17] M. Vahedi and A. F. van der Stappen, "Caging polygons with two and three fingers," *Int. J. Robot. Res.*, vol. 27, pp. 1308–1324, 2008.

[18] W. Wan and R. Fukui, "Efficient planar caging test using space mapping," *IEEE Trans. Autom. Sci. Eng.*, vol. 99, pp. 1–12, 2016.

[19] W. Wan, R. Fukui, M. Shimosaka, T. Sato, and Y. Kuniyoshi, "A new "grasping by caging" solution using eigen-shapes and space mapping," *Proc. Int. Conf. Robot. Autom.*, 2013, pp. 1566–1573.

[20] W. Wan, R. Fukui, M. Shimosaka, T. Sato, and Y. Kuniyoshi, "Grasping by caging: A promising tool to deal with uncertainty," *Proc. Int. Conf. Robot. Autom.*, 2012, pp. 5142–5149.

[21] Z. Wang and V. Kumar, "Object closure and manipulation by multiple cooperating mobile robots," *Proc. Int. Conf. Robot. Autom.*, 2002, pp. 394–399.