



ELSEVIER

Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Computational Geometry 27 (2004) 211–235

Computational
Geometry

Theory and Applications

www.elsevier.com/locate/comgeo

Kinetic collision detection between two simple polygons

Julien Basch^a, Jeff Erickson^{b,1}, Leonidas J. Guibas^{c,2}, John Hershberger^{d,*}, Li Zhang^e

^a SiRF Technology Inc., 395 Oyster Point Blvd., South San Francisco, CA 94080, USA

^b Department of Computer Science, University of Illinois, Urbana, IL 61801-2987, USA

^c Computer Science Department, Stanford University, Stanford, CA 94305, USA

^d Mentor Graphics Corp., 8005 SW Boeckman Road, Wilsonville, OR 97070-7777, USA

^e HP Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304, USA

Received 19 December 2002; received in revised form 23 October 2003; accepted 5 November 2003

Communicated by M. de Berg

Abstract

We design a kinetic data structure for detecting collisions between two simple polygons in motion. In order to do so, we create a planar subdivision of the free space between the two polygons, called the *external relative geodesic triangulation*, which certifies their disjointness. We show how this subdivision can be maintained as a kinetic data structure when the polygons are moving, and analyze its performance in the kinetic setting.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Kinetic data structures; Collision detection; Geodesic triangulation

1. Introduction

The problem of *collision detection* between moving objects is fundamental to simulations of the physical world. It has been studied in a number of different communities, including robotics, computer

* Corresponding author.

E-mail addresses: julien@basch.org (J. Basch), jeffe@cs.uiuc.edu (J. Erickson), guibas@cs.stanford.edu (L.J. Guibas), john_hershberger@mentor.com (J. Hershberger), l.zhang@hp.com (L. Zhang).

URLs: <http://www.uiuc.edu/ph/www/jeffe/> (J. Erickson), <http://graphics.stanford.edu/~guibas/> (L.J. Guibas), <http://graphics.stanford.edu/~lizhang/> (L. Zhang).

¹ Portions of this research were done at the International Computer Science Institute, Berkeley, CA. Research supported by National Science Foundation grant DMS-9627683 and by US Army Research Office MURI grant DAAH04-96-1-0013.

² Research partially supported by National Science Foundation grant CCR-9623851 and by US Army Research Office MURI grant DAAH04-96-1-0007.

graphics, computer-aided design, and computational geometry. Methods have been developed for the case of rigid bodies moving freely in two and three dimensions. Many extant techniques for collision checking on objects of complex geometry rely on hierarchies of simple bounding volumes surrounding each of the objects [9,13,14,17]. For a given placement of two non-intersecting objects, their respective hierarchies are refined only to the coarsest level at which the primitive shapes in the two hierarchies can be shown to be pairwise disjoint. This and several other optimizations have improved considerably the cost of collision detection. Though a physical simulation involves several other computational tasks, such as motion dynamics integration, graphics rendering, and collision response, collision detection still remains one of the most time consuming tasks in such a system.

Motion in the physical world is in general continuous over time, and many systems attempt to speed up collision checking by exploiting this temporal coherence, instead of repeating a full collision check *ab initio* at each time step [22]. Swept volumes in space or space-time have also been used towards this goal [4,13]. Though time-stepping at equal increments is customary for motion integration, collisions tend to be very irregularly spaced over time. If we know the motion laws of the objects, then it makes sense to try to predict exactly when collisions will happen. There have been a few theoretical papers in computational geometry along these lines [7,11,24], but their results are not so useful in practice because they use complex data structures and are only applicable for limited types of motion.

In this paper we focus on a problem that, while simple, still adequately addresses a number of the fundamental issues that arise as we try to move away from the limitations of these earlier methods. Our problem is that of detecting collisions between two simple polygons moving rigidly in the plane. What makes this problem challenging is that the two polygons can be quite intertwined and thus in close proximity in many places at once. We adopt the point of view of *kinetic data structures* [3,10], as it is very natural for the collision detection problem.

A kinetic data structure, or KDS for short, is built on the idea of maintaining a discrete attribute of objects in motion by updating a proof of its correctness as the objects move. The proof consists of a set of elementary conditions, or *certificates*, based on the kinds of tests performed by ordinary geometric algorithms (counterclockwise tests in our case). Those of the certificates that can fail as a result of the rigid motion of the polygons are placed in an event queue, ordered according to their earliest failure time. When a certificate fails, the proof needs to be updated. Unless a collision has occurred, we perform this update and continue the simulation. In contrast to fixed time step methods, for which the fastest moving object determines the time step for the entire system, a kinetic method is based on *events* (the certificate failures) that have a natural significance in terms of the problem being addressed (collision detection in this case). The kinetic model allows us to perform a rigorous combinatorial time-cost analysis and obtain practical solutions at the same time.

Unlike earlier collision detection methods that have focused on bounding volume hierarchies for complex objects, we focus on the free space between the moving objects. We tile this free space into cells of a certain type. Some cells of this tiling deform continuously as the objects move. As long as all the cells in the tiling remain non-self-intersecting, the tiling itself functions as the KDS proof of separation, or non-collision, between the objects. At certain times, of course, cells will become invalid and a combinatorial change to the tiling will become necessary. In designing a good tiling we seek to satisfy three somewhat opposing desiderata:

- select a deformable cell shape whose self-collisions are easy to detect (i.e., require few certificates),

- select a tiling that can conform or adjust to the motion of the polygons, so that its combinatorial structure remains valid for as long as possible, and
- make it easy to update the tiling when cell self-collisions do occur.

These desiderata are directly related to the compactness, efficiency, and responsiveness of our KDS.

We obtain such a tiling by maintaining a moving polygonal line separating the two polygons. More specifically, we maintain a structure containing the *relative convex hull* [29] of the two polygons. This structure is what we call the *external relative geodesic triangulation* (ERGT), a planar subdivision combining the idea of the relative convex hull and of the *geodesic triangulation* of a simple polygon [5]. (This latter structure was also used by Mount [18] for the static problem of intersection detection.) The ERGT effectively defines a set of flexible shells surrounding each of the polygons. The space between these shells consists of *pseudo-triangles*, which are the basic shapes used in our tiling. The ERGT can be quickly updated upon certificate failures (it is *responsive*, as described in [10]) and has many other nice properties. For example, as we will see, the number of certificates of our separation proof is related to the size of the *minimum link separator* for the two polygons [28]. Thus our separation proof automatically adapts to the complexity of the relative placement of the two polygons—from a single separating line when the polygons are far apart to as complex as necessary when they have many points of near contact. This feature is important in the kinetic model, in which objects are allowed to change their *motion plan* unpredictably.

The quality of a KDS is measured in part by the number of events it has to process in the worst case (its *efficiency*). Obviously, this number depends on the type of motions allowed. We derive the surprising result that when the moving polygons are translating along algebraic trajectories of bounded degree, the relative convex hull of the two polygons changes only $O(n)$ times, where n is the complexity of the polygons. A variation on this argument shows that under such motion our KDS will process $O(n \log n)$ events. If the polygons are also allowed to rotate by a constant number of full turns, we show that the number of events is near quadratic in the worst case (the obvious bound is cubic). These bounds are nearly optimal for structures incorporating the relative convex hull.

Since the publication in 1999 of a preliminary version of this paper [2], Kirkpatrick, Snoeyink, and Speckmann have proposed an alternative KDS for maintaining the separation of polygons moving in the plane [15,16]. Their approach uses a logarithmic factor fewer certificates than ours, but does not allow a provable bound on the number of events, as ours does. This difference may be because our structure is canonical—it depends on the current position of the polygons, but not on previous positions—whereas theirs is not.

Pseudo-triangles and pseudo-triangulations have proven useful in many additional contexts in the recent past. They have been used for ray shooting [5], computing visibility graphs and visibility complexes [19–21], art gallery guard placement [26], rigidity analysis and polygon unfolding/straightening [12,27], to mention just a few applications.

Section 2 presents the exterior relative geodesic triangulation for two non-intersecting simple polygons and the associated separation proof derived from it. It also shows how this proof can be maintained under continuous motion. Sections 3, 4, and 5 present the event bounds for the two models of motion considered. Section 6 concludes with plans for further work.

2. Certification and maintenance

We denote the boundary of a simple polygon P by ∂P , and adopt the convention that a simple polygon is an open set. For two vertices $a, b \in \partial P$, we denote by $C(a, b)$ the relatively open polygonal chain along ∂P from a to b going counter-clockwise. The complement of P is called the *free space* of P and denoted \mathcal{F}_P . The shortest path from a to b homotopic to $C(a, b)$ in \mathcal{F}_P is denoted $\pi_0(a, b)$.

In this paper, we consider *two* non-intersecting polygons P and Q . Define \mathcal{F} , the *free space*, to be the complement of $P \cup Q$. We denote by $\pi(a, b)$ the shortest path from a to b that is homotopic to $C(a, b)$ in \mathcal{F} . It is an oriented polygonal chain called the *geodesic* from a to b . If an edge connects two vertices of ∂P (respectively ∂Q), it is called a PP edge (respectively a QQ edge). If it connects a vertex from P and a vertex from Q , it is called a PQ edge. An oriented edge that connects two vertices u, v of P in \mathcal{F}_P is denoted \overrightarrow{uv} . Finally, we denote by $R(a, b)$ the open region delimited by $C(a, b)$ and $\pi(a, b)$ (Fig. 1).

Proposition 2.1. *Let (a, b) and (c, d) be two pairs of vertices, with each pair either on ∂P or ∂Q . If (a, b) and (c, d) are on different polygons, then $R(a, b)$ and $R(c, d)$ do not intersect. If (a, b) and (c, d) are on the same polygon and $C(a, b)$ and $C(c, d)$ do not overlap, then $R(a, b)$ and $R(c, d)$ do not intersect. If $C(a, b) \subseteq C(c, d)$, then $R(a, b) \subseteq R(c, d)$.*

Proof. The proof is based on the observation that two geodesics cannot intersect twice so as to create an open region S of free space bounded on all sides by the union of the geodesics. In such a case at least one of the geodesics could be shortened by following the opposite boundary of S .

If (a, b) and (c, d) are on different polygons, let a, b be the two vertices of P and c, d be two vertices of Q . Note that both $R(a, b)$ and $R(c, d)$ are all in free space. Hence, if they overlap, then any component S of $R(a, b) \cap R(c, d)$ can serve as the forbidden region that shows that at least one of $\pi(a, b)$ and $\pi(c, d)$ is not locally optimal.

If (a, b) and (c, d) are on the same polygon and $C(a, b)$ and $C(c, d)$ do not overlap, the argument is the same as above. If $C(a, b) \subseteq C(c, d)$ and $R(a, b) \not\subseteq R(c, d)$, then $R(a, b) \setminus R(c, d)$ contains one or more open regions bounded entirely by edges of $\pi(a, b)$ and $\pi(c, d)$. Once again at least one of the two geodesics is not locally optimal. \square

Given a subsequence S of vertices of P , the *pinned geodesic cycle* of P based on S is the sequence of geodesics in \mathcal{F} joining consecutive vertices of S . The subsequence is cyclic, so the last vertex is joined to the first. We say that the vertices of S are *pinned*.

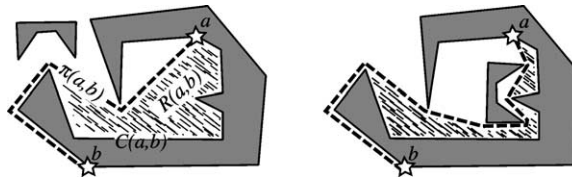


Fig. 1. The geodesic from a to b for two placements of the small polygon. A shortest path endpoint is indicated by a star.

2.1. External relative geodesic triangulation

Proposition 2.1 allows us to define a planar map. Let T_P (respectively T_Q) be a binary tree whose leaves are the edges of P (respectively Q) in counter-clockwise order, and let T be the binary tree whose root has T_P and T_Q as its two children. Each subtree of T (except T itself) is associated with a polygonal chain on one of the boundaries, so T defines a hierarchy of polygonal chains. With each node v of T that is not the root, we associate the geodesic between the two extreme vertices of the subtree rooted at v .

We associate with the root node a geodesic homotopic to ∂Q and pinned to a vertex p of P . We choose p depending on the configuration so that, when the convex hulls of P and Q are disjoint, this geodesic contains an inner common tangent and an outer common tangent between the hulls.

To maintain this condition, p must vary as the polygons move. We accomplish this as follows: Without loss of generality assume the diameter of P is no smaller than that of Q . Let (p_1, p_2) be a diametral pair of vertices of P . Consider enclosing P between two maximally separated parallel lines tangent to P . The lines pass through p_1 and p_2 . Because Q 's diameter is no greater than that of P , Q can intersect at most two of the three slabs bounded by these lines. We choose $p \in \{p_1, p_2\}$ such that Q does not intersect the half-plane bounded by the line through p , and maintain this condition over time. Whenever Q is about to enter the half-plane incident to p , we move p from p_1 to p_2 or vice versa, extending the geodesic loop by wrapping it along the portion of the convex hull of P from p_1 to p_2 that is not visible from Q . See Fig. 2.

Likewise, when Q leaves the half-plane incident to whichever one of $\{p_1, p_2\}$ is *not* p , we shorten the geodesic loop by moving p from p_2 to p_1 (or vice versa) and dropping the doubled path along the back side of P 's convex hull. Note that moving p from p_1 to p_2 or vice versa takes only $O(1)$ time. Although the loop may gain or lose $\Theta(n)$ edges, these paths can be precomputed and spliced in or out of the loop in constant time.

By Proposition 2.1, this system of geodesics defines a planar map (a *tiling*) that we call the *external relative geodesic triangulation* (or ERGT) of the pair (P, Q) based on T (Fig. 3). In the rest of this paper, we take a tree of depth $O(\log n)$ to define the ERGT, where n is the total number of vertices of the two polygons. In particular, we choose T_P and T_Q to be *complete binary trees*: each internal node has two children, and leaves appear only on two adjacent levels of the tree. Thus the number of nodes of T_P or T_Q with depth k is 2^k for all except the two maximal-depth levels of the tree. Fig. 4 shows an example of part of T_P and the corresponding portion of the ERGT.

Observation 2.2. *The edges of the ERGT are obtained by superimposing $O(\log n)$ pinned geodesic cycles for each polygon, and possibly some inner and outer common tangents of the polygons.*

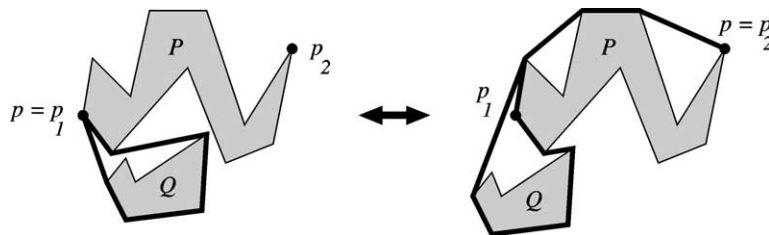


Fig. 2. The loop's attachment vertex p moves from p_1 to p_2 as Q moves past P .

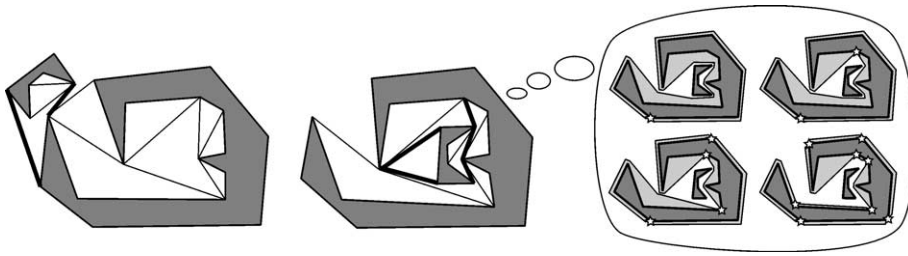


Fig. 3. Planar maps induced by the ERGT of two polygons, with the PQ edges in bold. The right polygon is dreaming about its first four pinned geodesic cycles.

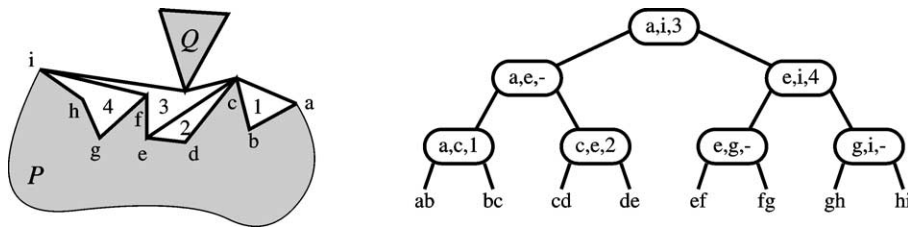


Fig. 4. The subchain a, \dots, i of ∂P , the corresponding part of T_P , and the associated ERGT. Each tree node lists the endpoints of the associated chain and the numerical label of the nontrivial pseudo-triangular face associated with the node, if any. A “—” means that there is no nontrivial face between the node’s chain and the chains of its children.

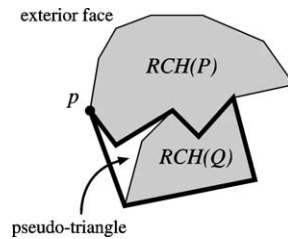


Fig. 5. The root node defines an infinite exterior face and a pseudo-triangle.

The ERGT has a number of properties that are straightforward generalizations of those of the geodesic triangulation of [5]. Consider a node v in T_P or T_Q . It has an associated geodesic, and its two children define two geodesics obtained by pinning an additional vertex. The open region between these three geodesics, if non-empty, is a face of the planar map called a *pseudo-triangle* (three convex vertices joined by three concave chains). We associate this face with the node v . The root node of T defines two faces: the infinite face (made of only one concave chain linked to itself by a convex vertex) and a pseudo-triangle. See Fig. 5. For any line segment in \mathcal{F} , the sequence of nodes of T corresponding to the faces whose interiors it crosses lie along a path in T .

A planar map is made of vertices, edges, and faces. Additionally, we will say that two adjacent edges along a face define a *corner*, which can be either *reflex*, *convex*, or *degenerate* (i.e., 0° or 180°) on that face. A condition that states that a given corner is reflex or convex is called a *corner certificate*. Now, suppose that we move the vertices in compliance with the certificates of all corners: are we sure that the map will remain planar, i.e., that no two edges will intersect? For a general planar map, this is not the

case, but the special structure of the pseudo-triangles allows the ERGT to be certified by its corners, in the following sense:

Lemma 2.3. *Let P, Q be two simple polygons moving continuously between time 0 and time t_f , and let Σ_t be their ERGT at t . If no corner of Σ_0 becomes degenerate before t_f , then $\Sigma_t = \Sigma_0$ for all $t < t_f$.*

Proof. We argue first that so long as the corners of a geodesic triangle stay nondegenerate while the polygons move, the geodesic triangle cannot self-intersect. Consider the slopes of lines tangent to the concave chains of a pseudo-triangle. (Imagine rolling a line along the inner boundary of the pseudo-triangle, maintaining contact with the chains as it moves.) The convexity of the corners and the concavity of the chains means that the slope of the rolling line changes monotonically as the point of tangency moves along the boundary. In particular, the slopes of the inner tangent lines supporting the different chains lie in three disjoint ranges. By the mean value theorem applied to slopes, if a line intersects a chain more than once, then the slope of the line lies in the chain's tangent range. In particular, a line that bisects the angle at a convex corner lies in the slope range of none of the chains, and therefore it separates the two chains incident to the corner. Applying this argument at all three convex corners shows that the chains of the pseudo-triangle intersect only at the convex corners. The boundary of the pseudo-triangle remains simple as the polygons move.

Now consider the exterior face F of the root node. It is bounded by a single concave chain whose ends meet at a convex vertex. The complement of the exterior face, \bar{F} , is tiled by P, Q , and a collection of pseudo-triangles. Since none of them collapses, as argued above, no vertex of ∂F can penetrate ∂F coming from the side of \bar{F} (that is, the collapse shown on the right side of Fig. 6 is not possible). Let p be the convex vertex of F . For every point q on ∂F , the segment \overline{pq} lies in the closure of \bar{F} . Choose q to be the point where ∂F intersects the bisector of the angle at p . The line supporting \overline{pq} cuts \bar{F} into two convex pieces (the angle at every vertex in each piece is convex). Therefore F , as the complement of the union of two touching convex polygons, cannot collapse—no vertex of ∂F can penetrate ∂F from the side of F (from the exterior face).

Because the geodesic triangles and the exterior face tile the free space, abutting along shared edges, local validity of each face implies global validity of the ERGT tiling. \square

In particular, a collision can occur between P and Q only when a certificate fails. Thus, we will be able to detect collisions if we can maintain the ERGT.

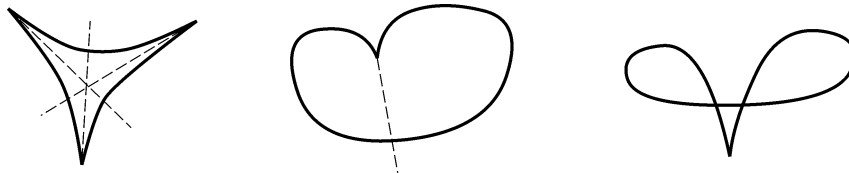


Fig. 6. A pseudo-triangle certified by convex/reflex certificates cannot self-intersect (left figure). The outer face could self-intersect (right figure), except that its complement (the interior) is prevented from collapsing by the polygons and pseudo-triangles that tile it.

2.2. Compactness and locality

Some certificates in the geodesic triangulation involve only vertices of P , some involve only vertices of Q , and some involve both. In a context in which P and Q both move rigidly, only the certificates involving both polygons can ever fail. Those are the corner certificates that are adjacent to PQ edges in the ERGT. This set of certificates is called the *active set*.

In the kinetic setting, a change of the motion plan of one of the polygons makes it necessary to recompute the failure times of the active set, and it is therefore desirable to have as small an active set as possible.

Lemma 2.4. *In an ERGT of P and Q with n vertices in total, the active set has $O(\kappa \log \frac{n}{\kappa})$ certificates, where κ is the size of a minimum link separator of P and Q .*

Proof. A line segment disjoint from the interior of P can cross a geodesic between two vertices of P at most once. Therefore, it can cross a pinned geodesic cycle at most twice. (Once a segment enters a region $R(a, b)$, it cannot come back out.) Consider a separator that has κ edges. Let Π be a pinned geodesic cycle of P . Any polygonal chain that separates P and Q has to cross all its bichromatic edges (edges between a vertex of P and a vertex of Q), hence the number of bichromatic edges created by Π is at most 2κ .

Each certificate from the active set is adjacent to a bichromatic edge, and there are at most four corners adjacent to one edge. Hence the number of certificates is at most four times the number of bichromatic edges of the ERGT.

By Observation 2.2, the total number of bichromatic edges of the ERGT is $O(\kappa \log n)$. To improve this bound, we count the number of unique bichromatic edges associated with each level of T .

Let a, b , and c be three vertices in counterclockwise order along ∂P . The portion of $\pi(a, b)$ disjoint from $\pi(a, c)$ is contained in $R(a, c) \cup C(a, c)$. Therefore, of the edges of $\pi(a, b) \setminus \pi(a, c)$, at most one is a PQ edge—only the point where $\pi(a, b)$ separates from $\pi(a, c)$ can belong to Q . A symmetric argument applies to $a, b, c \in \partial Q$. It follows that if a given level of T below the top two levels has k nodes, the maximum number of PQ edges that belong to the pinned geodesic cycle for that level and not to the cycles of any higher levels is at most k . For level j of the tree, $j > 1$, the number of unique PQ edges belonging to its geodesic cycle is at most $\min(2^j, 2\kappa)$. The levels for which 2^j is not the upper bound are $\{0, 1, \lceil \log_2 \kappa \rceil, \lceil \log_2 \kappa \rceil + 1, \dots\}$, and for each of these levels the number of unique PQ edges crossed by a separator is at most 2κ .

Because T_P and T_Q have been chosen to be complete binary trees, the height of T is at most $\lceil \log_2 n \rceil + 1$. Thus the total number of distinct PQ edges crossed by a separator is at most

$$2\kappa(2 + \log_2 n - \log_2 \kappa) + \sum_{j=2}^{\log_2 \kappa} 2^j = O\left(\kappa \left(1 + \log \frac{n}{\kappa}\right)\right).$$

This completes the proof of the lemma. \square

2.3. Maintenance of the ERGT

We now assume that we have two moving simple polygons P and Q . We assume transversality in space/time, i.e., there is at most one group of three vertices collinear at any given time, and if three

vertices are collinear at t , then the triangle they define really flips orientation between just before and just after that instant. This assumption also implies non-degeneracy of P and Q separately: no three vertices of P are collinear, and the same holds for Q . (The general position assumption is for purposes of exposition; it can be removed, if necessary, by applying a symbolic perturbation [6].) We further assume that, given the knowledge of the motions of P and Q , we can compute the failure time of any certificate with $O(1)$ computation.

We maintain the ERGT of P and Q by taking all corner certificates of the associated planar map, and putting them in an event queue ordered by time of failure. As the ERGT remains the same when no certificate fails, we just need to describe how to update it when there is an event. An event, in general, involves updating the geodesic triangulation, and descheduling and rescheduling in the event queue the corner certificates that are affected by this update.

As we have seen, there are two types of certificates (reflex and convex), and therefore two types of events, which are pictured in Fig. 7. The failure of a convex certificate (right to left in Fig. 7) is easy to handle, as there is only one possible resulting map.³ However, when a reflex certificate fails, we need to choose between three possible resulting maps. This can be done with the help of the tree on which the ERGT is based.

Consider the situation on the left of Fig. 7. Let f be the face of the reflex certificate, and let v be its adjacent vertex. The face f has two adjacent faces around v , which we denote f_ℓ and f_r . Recall that our ERGT is based on a binary tree T , and that each face has an associated node in this tree (each node is associated primarily with a geodesic and secondarily with the face, if any, between its geodesic and those of its child nodes). Let v, v_ℓ, v_r be the three nodes associated with our faces. If either adjacent face is absent, that is, f is bounded on either side by a polygon edge, not a face, let the appropriate v_ℓ or v_r be the node associated with the polygon edge adjacent to f .

Proposition 2.5. *The relative positions of nodes v, v_ℓ, v_r are different in each case of Fig. 7. More precisely, if none of them is the root node of T , the cases are:*

- (a) *the three nodes are not on a common path,*
- (b) *the three nodes are on a common path, and v_r is between v and v_ℓ ,*
- (c) *the three nodes are on a common path, and v_ℓ is between v and v_r .*

Proof. As we mentioned in the paragraph after Observation 2.2, if a segment in \mathcal{F}_P crosses a sequence of faces, the associated nodes lie along a path in T . In each case (a)–(c), we choose appropriate segments

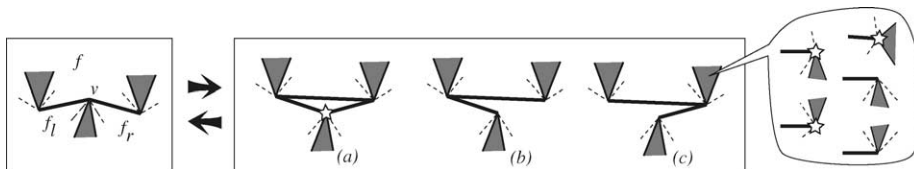


Fig. 7. The failure of a convex certificate (right to left), and of a reflex certificate (left to right). The polygon edges incident to the extreme vertices may lie on either side of the edge involved in the event (right inset).

³ In case (a), three convex certificates fail at the same time, and extra care must be taken to handle this properly.

around the vertex v to prove the claim. In case (b), we can draw a segment that crosses f, f_r, f_ℓ in this order, and in case (c) we can draw a segment that crosses f, f_ℓ, f_r in this order.

In case (a) we need to show that the three nodes do not lie on a common path. In this case faces f, f_ℓ , and f_r lie on the three sides of a central triangle τ . Let v_τ be the node associated with τ . We prove that none of the three nodes v, v_ℓ , and v_r lies between the other two on a common path. Without loss of generality, suppose that v lies between v_ℓ and v_r . We can draw a segment that cuts f_ℓ, τ, f_r in order. Thus both v and v_τ lie between v_ℓ and v_r . The two possible orders are v_ℓ, v, v_τ, v_r and v_ℓ, v_τ, v, v_r . In the first case, v lies between v_ℓ and v_τ ; however, we can draw a segment that proves that v_τ lies between v_ℓ and v . In the second case, v lies between v_τ and v_r ; however, we can draw a segment that proves that v_τ lies between v and v_r . Hence neither order is possible, and the three nodes v, v_ℓ , and v_r cannot lie on a common path. \square

If one of the nodes is the root node, we can distinguish how the event should be handled with a little extra work. Details are left to the reader.

Here is high-level pseudocode for the kinetic maintenance of the ERGT:

```

Let  $Q$  be a time-ordered priority queue of convex and reflex certificates.
while ( $Q$  is not empty) {
   $C \leftarrow Q.\text{popHead}()$ ;
  Dequeue all certificates involving the edges of  $C$ ;
  if ( $C$  is a convex certificate)
    Restructure the ERGT as in Fig. 7 (right to left);
  else
    Restructure the ERGT as in Fig. 7 (left to right), choosing
      which local configuration to create by using Proposition 2.5;
  Enqueue certificates for all the corners involving modified edges of
    the ERGT, with failure times based on the motions of  $P$  and  $Q$ ;
}
```

Our kinetic data structure maintains the ERGT, and, for each face, a pointer to its node in T (note that T is fixed over time: it is not the dual tree of the planar map). When a reflex certificate fails, we use these pointers as indicated by Proposition 2.5 to decide how to handle the event. This can be done in $O(1)$ time with a constant number of least-common-ancestor queries [23]. (In fact, we can find least common ancestors using a trivial $O(\log n)$ traversal of T without increasing the asymptotic running time of event handling.) In all cases, the update of the ERGT involves the destruction and creation of a constant number of edges. Each corner certificate that is disturbed during this process needs to be descheduled or rescheduled in the event queue, which takes an additional time logarithmic in the size of the active set. In other words, our KDS is responsive.

The reader is invited to examine Fig. 3, and to imagine how the structure of the ERGT would change if the small polygon were to move around.

Theorem 2.6. *Our kinetic data structure for maintaining an ERGT of two moving polygons P and Q is compact and responsive: it uses $O(\kappa \log \frac{n}{\kappa})$ certificates, where κ is the size of a minimum link separator for P and Q , and each certificate failure can be corrected in $O(\log n)$ time.*

As noted in passing in Section 2.2, a change to the flight plan of P or Q changes the failure time of every certificate. Thus our KDS is not local. However, this is to be expected—a polygon does not have constant description complexity, and the many possible near-collisions between two polygons must be represented somehow.

3. Efficiency preliminaries

In the next two sections, we analyze the number of combinatorial changes to the ERGT as the polygons move. The ERGT is composed of $O(\log n)$ pinned geodesic cycles. A combinatorial change to it occurs when a vertex is inserted into or deleted from the vertex sequence of one of those pinned geodesic cycles.

We analyze the number of combinatorial changes under two models of motion: pure translation and translation with bounded rotation. In both cases, we will assume without loss of generality that P is stationary and Q is moving. The position and orientation of Q are given by a moving orthogonal reference frame—a point $q(t)$ and a pair of orthogonal unit vectors $x(t)$, $y(t)$ —whose coordinates are continuous functions of time. The vertices of Q have fixed coordinates relative to the moving reference frame. In order to compute the failure times of corner certificates in constant time, we assume that the coordinates of the reference frame, and thus the coordinates of every vertex, are polynomials of bounded degree. Any rigid motion can be approximated by such a moving reference frame to any desired accuracy, for a limited time. However, bounded-degree algebraic rigid motions necessarily have non-uniform angular velocity and can cover only a constant number of full turns.

Remark. A polynomial representation of rotational motion is inherently inexact: If $x(t) = (a(t), b(t))$ is required to be a unit vector, then $(a(t))^2 + (b(t))^2 = 1$, and this is not possible with non-constant polynomials. Square roots must enter into the representation to keep $|x(t)| = 1$. Possibilities include $x(t) = (a(t), \sqrt{1 - (a(t))^2})$ and $x(t) = (a(t), b(t)) / \sqrt{(a(t))^2 + (b(t))^2}$, where $a(t)$ and $b(t)$ are polynomials. One can also write $x(t) = (\cos \theta(t), \sin \theta(t))$, where $\theta(t)$ is polynomial and bounded to some constant-size interval. However, introducing sines and cosines increases the complexity of the computation. In all cases, if $x(t) = (a(t), b(t))$, then $y(t) = (-b(t), a(t))$.

For both pure translation and rigid motion, we show that the worst-case number of changes to the ERGT is about the same as the worst-case number of changes to the relative convex hull. Our results are as follows:

Theorem 3.1. *If two simple polygons P and Q with n vertices translate along algebraic trajectories of degree k , then the number of changes to their ERGT is $O(kn \log n)$. The worst-case number of changes to the relative convex hull is $\Omega(kn)$.*

Theorem 3.2. *If two simple polygons P and Q with n vertices undergo bounded-degree algebraic rigid motion, then the number of changes to their ERGT is $O(n^{2+\varepsilon})$ for any $\varepsilon > 0$, where the hidden constant depends on the exact parameters of the motion. The worst-case number of changes to the relative convex hull is $\Omega(n^2)$.*

Like the bounds for other kinetic data structures [3,8,10], these bounds do not actually require the motion to be algebraic, but only that it satisfy certain combinatorial conditions. However, it is *not*

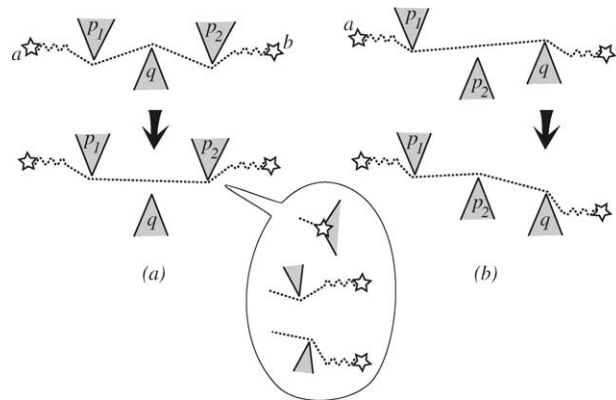


Fig. 8. (a) An inner appearance and (b) an outer appearance of the oriented edge $\overrightarrow{p_1 p_2}$ along $\pi(a, b)$ as Q moves downwards. In each case, the polygon edges incident to the extreme vertices may lie on either side of the connecting edge (inset), except that, for the outer appearance, the vertex from Q cannot be pinned.

sufficient to assume that any individual certificate can fail only a constant number of times, as it is for most previous KDSs. Whether our bounds can be extended to more general classes of pseudo-algebraic motion remains an open problem.

Lemma 3.3. *Any combinatorial change to the ERGT involves the appearance or disappearance of some PP or QQ edge in a pinned geodesic cycle.*

Proof. A combinatorial change consists of the insertion or deletion of a single vertex v between two other vertices in the vertex sequence of some pinned geodesic cycle. Without loss of generality suppose $v \in \partial P$ is inserted between two vertices u and w . If either u or w is a P vertex, then it forms a new PP edge with v ; otherwise \overline{uw} is a QQ edge that disappears when v is inserted. Symmetric arguments hold if v is deleted and/or belongs to ∂Q . \square

It suffices to bound the number of appearances, because the total number of disappearances is at most n more than the number of appearances. A PP edge may appear for two different reasons: if Q suddenly stops intersecting it (we call this an *inner appearance*), or if one of the P vertices starts intersecting a PQ edge (we call this an *outer appearance*). Fig. 8 illustrates these two events for specific positions of the vertices of P .

4. Bounds for translational motion

In this section, we consider the case in which the motion of Q is pure translation. In this case, the motion of Q can be described by the motion of a single point. Let us say that the position of this point at time t is $q(t)$. We say that the motion is *convex* between t_1 and t_2 if the projection of $q(t)$ on any line ℓ has at most one local extremum in the open time interval $t_1 < t < t_2$. Thanks to the following lemma, we can assume without loss of generality that the motion of Q is convex.

Lemma 4.1. *If the coordinates of $q(t)$ are polynomials of degree k , then $q(t)$ can be decomposed into $O(k)$ convex motion fragments.*

Proof. Let $q(t) = (x, y) = (X(t), Y(t))$. We break the trajectory at all its vertical and horizontal tangents, and at its points of inflection. An inflection point of the trajectory is a point where $\frac{d^2y}{dx^2} = 0$. Using the standard prime notation for derivatives, $Y'(t) = \frac{dy}{dt}$ and $X'(t) = \frac{dx}{dt}$. Then

$$\frac{dy}{dx} = \frac{dy/dt}{dx/dt} = \frac{Y'(t)}{X'(t)},$$

and

$$\frac{d^2y}{dx^2} = \frac{d}{dx} \left(\frac{Y'(t)}{X'(t)} \right) = \frac{d}{dt} \left(\frac{Y'(t)}{X'(t)} \right) \frac{1}{X'(t)} = \frac{X'(t)Y''(t) - Y'(t)X''(t)}{(X'(t))^3}.$$

Therefore, we break the time domain at all nontrivial solutions to $X'(t) = 0$, $Y'(t) = 0$, and $X'(t)Y''(t) = X''(t)Y'(t)$. This divides the motion into $O(k)$ motion fragments. We need to argue that each fragment is convex.

We focus on a single fragment given by a time interval $[t_0, t_1]$. By the mean value theorem applied to slopes, no horizontal or vertical line cuts the fragment trajectory more than once. If the fragment is nonconvex, let ℓ be a line such that $q(t)$ has multiple local extrema when projected on ℓ during the interval $t_0 < t < t_1$. Let a and b be consecutive local extrema; a and b are tangent to two parallel lines perpendicular to ℓ , and the trajectory between a and b lies between these lines. The second derivatives of the trajectory with respect to x have opposite sign at a and b . The mean value theorem implies that the second derivative has a zero between a and b , which contradicts our choice of fragment intervals. \square

To prove Theorem 3.1, we consider the number of appearances of PP and QQ edges on a single pinned geodesic cycle of P , which we call $PGC(P)$ in the sequel. We denote the relative convex hull of P (a geodesic cycle with no pinned vertices) by $RCH(P)$.

Lemma 4.2. *Any QQ edge of $PGC(P)$ is also an edge of $RCH(P)$ and of $RCH(Q)$.*

Proof. Viewed as a polygon, $RCH(P)$ contains $PGC(P)$. Neither contains any point of Q 's interior. Hence the vertices of a QQ edge of $PGC(P)$ also belong to $RCH(P)$. Because the edge belongs to $PGC(P)$, no part of P crosses the edge. Hence the edge is also an edge of $RCH(P)$.

Any vertex of Q that appears on $RCH(P)$ lies on a path that is the intersection of the boundaries of $RCH(P)$ and $RCH(Q)$. Therefore, if a QQ edge appears on $RCH(P)$, it must lie along this path, and hence be part of $RCH(Q)$. \square

According to this lemma, if a QQ edge appears or disappears on $PGC(P)$, that edge is also an edge of $RCH(Q)$. However, its appearance on $PGC(P)$ may not correlate with a change to $RCH(Q)$.

We now bound the number of appearances of PP edges on $PGC(P)$. Subsequently, we will extend this to bound the total number of combinatorial changes to $PGC(P)$. The idea, explained in more detail later, is to apply the argument symmetrically to $PGC(Q)$ and to $RCH(Q)$, and then separately bound the number of times each QQ edge that ever appears on $RCH(Q)$ can enter $PGC(P)$.

Recall that $\pi_0(a, b)$ is the shortest path that avoids only the polygon P for $a, b \in P$. It is identical to $\pi(a, b)$ when the convex hulls of P and Q are disjoint. Let $R_0(a, b)$ be the region bounded by $\pi_0(a, b)$

and $C(a, b)$, just as $R(a, b)$ is the region bounded by $\pi(a, b)$ and $C(a, b)$. Note that for any position of Q , $R(a, b) \subseteq R_0(a, b)$ — Q pushes $\pi(a, b)$ inward, toward $C(a, b)$, and never outward.

Let \overrightarrow{uv} be an oriented PP edge. We extend it beyond v until it hits P (or to infinity, if it does not hit P), and denote the part of this segment (or ray, or empty set) by \overrightarrow{uv} . We call \overrightarrow{uv} the *extension of \overrightarrow{uv} at v* . The extension \overrightarrow{vu} is defined symmetrically. The union of \overrightarrow{uv} and its two extensions is denoted $\widehat{\overrightarrow{uv}}$. It cuts \mathcal{F}_P in up to four components (at most four, by non-degeneracy).

Definition 4.3. The *pocket* of the oriented edge \overrightarrow{uv} , denoted $\text{pocket}(uv)$, is the component of $\mathcal{F}_P \setminus \widehat{\overrightarrow{uv}}$ that is locally to the left of \overrightarrow{uv} . If \overrightarrow{uv} is a finite segment, the *neighbor pocket of v* is the component with \overrightarrow{uv} on its boundary, and \overrightarrow{uv} not on its boundary. The neighbor pocket of u is defined symmetrically (Fig. 9).

If a neighbor pocket is on the same side of $\widehat{\overrightarrow{uv}}$ as $\text{pocket}(uv)$, it is called *forward-facing*, otherwise it is called *backward-facing*.

Note that a pocket may be finite or infinite. If \overrightarrow{uv} is clockwise-oriented, its pocket may be unbounded, and we call this an *infinite pocket*. Because a neighbor pocket may be either forward-facing or backward-facing, the lids of a pocket and its neighbors may have opposite orientations. At most one of a pocket and its neighbor pocket(s) can be infinite. See Fig. 10. A neighbor pocket is a pocket in its own right. The

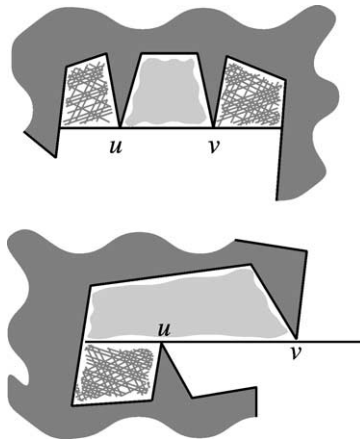


Fig. 9. Two instances of an oriented edge \overrightarrow{uv} . The pocket it defines is in gray, and the neighbor pockets are hatched.

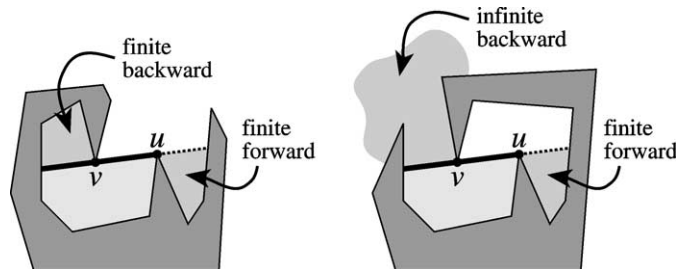


Fig. 10. Neighbor pockets. $\text{pocket}(uv)$ is lightly shaded, neighbor pockets are shaded darker, and the lid of $\text{pocket}(uv)$ is shown as a thick line.

endpoints of the extension segment that defines it may not be vertices, but the definition of a pocket does not require them to be.

A finite pocket of P is *full* if it contains at least one point of Q , and *empty* otherwise. The *lid* of $\text{pocket}(uv)$ is the portion of \overleftrightarrow{uv} on its boundary. If a finite pocket contains a pinned vertex of $PGC(P)$ in its closure, we say that the pocket itself is *pinned*.

Lemma 4.4. *Let P_1 and P_2 be two finite, disjoint pockets of P whose lids are parallel, with P_1 and P_2 locally on the same sides of their lids. Suppose that at some time t_1 , pocket P_1 is full and P_2 is empty, and that at some later time t_2 , pocket P_1 is empty and P_2 is full. If P and Q remain disjoint and translate continuously between t_1 and t_2 , then there exists a time t' , with $t_1 < t' < t_2$, at which both P_1 and P_2 are empty.*

Proof. We argue by quantifying the notion of fullness. Let $d(P_i, t)$ denote the “depth of penetration” of pocket P_i at time t . If P_i is empty, $d(P_i, t) = 0$. If P_i is full, $d(P_i, t)$ is the maximum perpendicular distance from the line supporting the lid of P_i to a vertex of Q inside P_i . Note that for any time interval in which P_i is full, the depth of penetration is always determined by the same vertex of Q .

Let $t' \in [t_1, t_2]$ be the first time when P_1 becomes empty. We want to argue that P_2 is empty at t' .

If P_2 is full at any time before t' , let $t_f \in [t_1, t']$ be the last time before t' when P_2 becomes full. Then $d(P_1, t_f) > 0$ and $d(P_2, t_f) = 0$. Since depth of penetration is determined by the same vertex during any interval of fullness, it follows that $d(P_1, t) \geq d(P_2, t)$ for all $t_f \leq t \leq t'$. That is, P_1 is full until after P_2 is empty again. When P_1 becomes empty, P_2 is empty as well. \square

The definitions of pockets and neighbor pockets allow us to characterize the placements of Q for which a PP edge can be present on $PGC(P)$.

Lemma 4.5. *If an oriented PP edge e is present on $PGC(P)$ for some position of Q , then $\text{pocket}(e) \subset R(a, b)$, for two consecutive pinned vertices a and b .*

Proof. Edge e belongs to $\pi(a, b)$ for two consecutive pinned vertices a and b . For each endpoint v of e , either the lid of $\text{pocket}(e)$ ends at v , or $\pi(a, b)$ bends away from $\text{pocket}(e)$ when it leaves the lid of $\text{pocket}(e)$ (if $\pi(a, b)$ crossed the lid to enter $\text{pocket}(e)$, then the path could be shortcut by a path not containing e). See Fig. 11. Hence the only intersection of $\text{pocket}(e)$ with $\pi(a, b)$ is a segment of the pocket lid, and because $\text{pocket}(e)$ is locally to the left of $\pi(a, b)$ there, $\text{pocket}(e)$ is contained in $R(a, b)$. \square

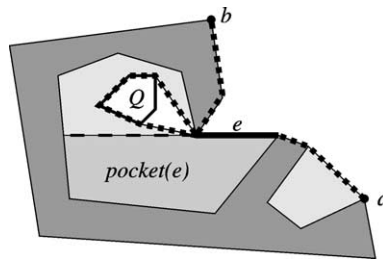


Fig. 11. $\text{pocket}(e) \subset R(a, b)$. $R(a, b) \setminus \text{pocket}(e)$ is lightly shaded, and $\text{pocket}(e)$ is shown slightly darker.

Corollary 4.6. *If an oriented PP edge e is present on $PGC(P)$ for some position of Q , then $pocket(e)$ is finite, empty, and contains no pinned vertices not collinear with e .*

Proof. This follows immediately from the preceding lemma, since $R(a, b)$ is finite, empty, and has pinned vertices only at a and b . \square

Lemma 4.7. *If a PP edge e is present on $PGC(P)$ for some position of Q not tangent to e or its extensions, then for each unpinned endpoint v of e , the extension of e at v is a non-empty segment or ray. If the extension is a finite segment, then the neighbor pocket of v is either infinite, pinned, or full.*

Proof. Edge e belongs to $\pi(a, b)$ for some pair of consecutive pinned vertices a and b . If a vertex v of e is unpinned, $\pi(a, b)$ must bend at v , and P must lie locally in the interior of the angle formed by the bend. Hence the extension at v is non-empty.

Now suppose the extension is finite and the neighbor pocket N is finite and unpinned. The definition of the neighbor pocket implies that N lies on the same side of the line supporting e as P does in the vicinity of v —see Fig. 10. Because $\pi(a, b)$ bends around P at v , it enters N . Because N is unpinned, $\pi(a, b)$ comes back out of N , crossing its lid a second time. If Q does not enter N , $\pi(a, b)$ can be shortcut using a segment of the lid of N . Hence N must be full. \square

This characterization lets us bound the number of appearances of any PP edge, as a pocket can become empty at most once during a single convex motion.

Lemma 4.8. *An oriented PP edge has at most one inner appearance and one outer appearance on $PGC(P)$ during a single convex motion.*

Proof. At the instant a PP edge e has an inner appearance, some point of Q exits $pocket(e)$. Hence the projection of Q 's motion on the normal to e is directed away from $pocket(e)$. Before e can have a second inner appearance, Q must re-enter $pocket(e)$, then reverse direction to exit it again, creating a non-convex motion.

An outer appearance of e occurs when a vertex of Q enters a neighbor pocket N of e . If N is forward-facing, it is empty prior to e 's appearance; if N is backward-facing, it completely contains Q after the appearance. We break possible consecutive outer appearances of e into three cases.

- (a) The same neighbor pocket of e is entered by a vertex of Q twice. The argument for inner appearances of e applies: Q must cross the lid of the neighbor pocket three times, with alternating directions. See Fig. 12(a).
- (b) Different neighbor pockets are entered, but both neighbor pockets lie on the same side of the line supporting e . See Fig. 12(b). Q crosses the line in the same direction for each appearance, and must cross it in the opposite direction between appearances. (If the neighbor pockets lie on the same side of the line as $pocket(e)$, both must be full for e to appear, and Q must exit one before the second appearance; if the neighbor pockets lie on the opposite side of the line from $pocket(e)$, only one of them can be full at a time, and Q must exit the first one before entering the second.)
- (c) Different neighbor pockets are entered, and they lie on opposite sides of the line supporting e . See Fig. 12(c). Edge e separates its two extensions in \mathcal{F}_P . During an outer appearance of e , Q crosses

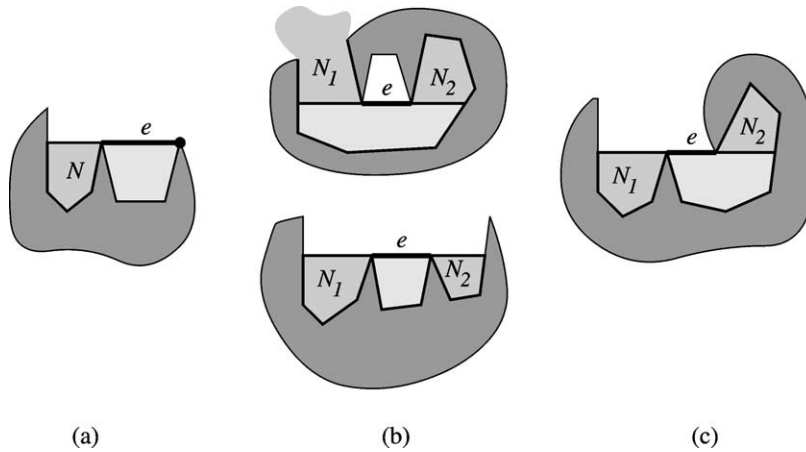


Fig. 12. Each oriented PP edge has at most one inner and one outer appearance on $PGC(P)$.

one of the extensions but does not cross e . Therefore, between successive outer appearances, Q must cross one extension, then e , then the other extension, which gives a nonconvex motion. \square

To bound the number of appearances of PP edges on $PGC(P)$, we identify three types of edges that appear on $PGC(P)$:

1. edges that belong to $\pi_0(a, b)$ for some pair of consecutive pinned vertices a and b ;
2. edges not of type (1) that have at least one finite, unpinned, backward-facing neighbor pocket at an endpoint not on $\pi_0(a, b)$; and
3. edges not of type (1) or type (2).

Note that by Lemma 4.7, an edge not of type (1) has a non-empty extension at each of its endpoints that is not on $\pi_0(a, b)$ —the endpoint cannot be pinned. We prove that there are $O(n)$ edges of each type. Lemma 4.8 then shows that there are only $O(n)$ appearances of these edges.

The first bound is easy: there are $O(n)$ edges on $\pi_0(a, b)$, summed over all consecutive pinned vertices a and b .

We now bound the contribution of edges of type (2), those with at least one finite, backward-facing neighbor pocket at a vertex not on $\pi_0(a, b)$.

Lemma 4.9. *There are at most $O(n)$ edges e such that e is an edge of $\pi(a, b)$ for some position of Q , $e \notin \pi_0(a, b)$, and some endpoint of e not on $\pi_0(a, b)$ has a finite, unpinned, backward-facing neighbor pocket.*

Proof. Let $e = \overline{uv}$, and let u be an endpoint inside $R_0(a, b)$ that has a finite, unpinned, backward-facing neighbor pocket N . By Lemma 4.7, whenever e belongs to $\pi(a, b)$, Q must be contained in N . See Fig. 13. The path $\pi(a, b)$ can be decomposed into three parts: $\pi(a, u)$, a loop inside N from u around Q and back to u , and $\pi(u, b)$. The portion of $\pi(a, b)$ outside N depends only on u . Therefore, there is only one edge e associated with each vertex $u \in (C(a, b) \setminus \pi_0(a, b))$ that can belong to any $\pi(a, b)$. It follows that there are only $O(n)$ edges that meet the conditions of the lemma. \square

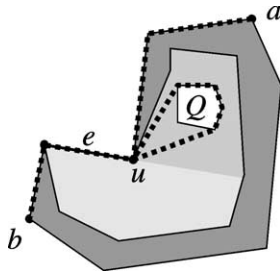


Fig. 13. Q is completely contained in the backward-facing neighbor pocket of e .

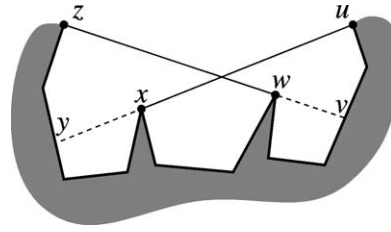


Fig. 14. Two restricted PP edges that cross.

We now focus our attention on edges of type (3), PP edges of $PGC(P)$ that do not belong to any $\pi_0(a, b)$ and have no finite, unpinned, backward-facing neighbor pockets at their endpoints not on $\pi_0(a, b)$. We call these edges *restricted PP edges*.

Lemma 4.10. *If two restricted PP edges of $PGC(P)$ that appear during the motion of Q cross (i.e., their interiors intersect), then each has a finite, forward-facing neighbor pocket that is contained in the other edge's pocket.*

Proof. By non-degeneracy, the two edges must cross properly—they cannot be collinear. Let the two edges be \overline{ux} and \overline{wz} , and without loss of generality assume that x is contained in $pocket(wz)$ and w is contained in $pocket(ux)$. See Fig. 14. Because all the regions $R_0(a, b)$ are disjoint for different pairs (a, b) , \overline{ux} and \overline{wz} both belong to $\pi(a, b)$ for the same consecutive pair of pinned vertices a and b , albeit at different times.

By Lemma 4.5, if e is a PP edge of $PGC(P)$, $pocket(e)$ is contained in $R(a, b)$. By transitivity, $pocket(e)$ is also contained in $R_0(a, b)$.

Since $pocket(wz)$ and $pocket(ux)$ are both contained in $R_0(a, b)$, neither x nor w is pinned. Hence the extensions at x and w must exist, by Lemma 4.7. Furthermore, each extension is contained in the other edge's pocket: for example, \overline{vw} cannot intersect \overline{ux} , because \overline{vz} intersects \overline{ux} in only one place, namely on \overline{wz} . Therefore, the extensions are finite, and bound neighbor pockets. We now argue that these neighbor pockets are finite. Edge \overline{wz} and its extension at w cut $pocket(ux)$ into three regions. Two of these regions are incident to \overline{ux} , and the third is the neighbor pocket at w . The neighbor pocket is fully contained in $pocket(ux)$ and hence finite (and unpinned). The same argument applies to the neighbor pocket of \overline{ux} at x . Because \overline{ux} and \overline{wz} are restricted PP edges, the neighbor pockets are forward-facing. \square

Lemma 4.11. *No two restricted PP edges that have outer appearances on $PGC(P)$ during a single convex motion of Q can cross.*

Proof. If a restricted PP edge e has an outer appearance on $PGC(P)$, it follows from Corollary 4.6 and Lemma 4.7 that $pocket(e)$ is empty, and some vertex of Q crosses the lid into a neighbor pocket at the moment of appearance. The pocket Q enters is finite and unpinned, and empty just prior to the appearance of e (otherwise $\pi(a, b)$ would already enter the pocket). If e has two neighbor pockets that are finite and unpinned (so both must be full for e to appear), then both neighbor pockets are forward-facing—otherwise Q would have to intersect $pocket(e)$ to intersect both neighbor pockets.

We prove the lemma by contradiction. Suppose the lemma is false, and the two crossing edges are \overline{ux} and \overline{wz} , as shown in Fig. 14, and without loss of generality assume that \overline{wz} appears before \overline{ux} , at some time t_1 . Define the positive normal to \overline{wz} to be the one that points away from $pocket(wz)$. When \overline{wz} appears at time t_1 , some vertex of Q crosses \overline{wz} in the negative direction (because the neighbor pocket Q enters is forward-facing). Just after t_1 , $pocket(vw)$ is full and $pocket(wz)$ is empty. When \overline{ux} appears at some time t_3 , $pocket(xy)$ is full and $pocket(ux)$ is empty, implying that $pocket(vw)$ is empty and $pocket(wz)$ is full.

By Lemma 4.4, $pocket(vw)$ becomes empty before $pocket(wz)$ becomes full, at time t_2 , with $t_1 < t_2 < t_3$. Hence some vertex of Q crosses \overline{wz} in the positive direction at t_2 , and another vertex crosses \overline{wz} in the negative direction at t_3 . The projection of Q 's motion on the normal to \overline{wz} therefore has two successive local extrema, and Q 's motion is not convex. \square

Lemma 4.12. *No two restricted PP edges that have inner appearances on $PGC(P)$ during a single convex motion of Q can cross.*

Proof. Once again, the proof is by contradiction. Suppose that the two crossing edges are \overline{ux} and \overline{wz} , as shown in Fig. 14. By Corollary 4.6, each pocket changes from full to empty at the moment of its edge's appearance. By Lemma 4.7, the forward-facing neighbor pocket inside $R_0(a, b)$ must be full. Without loss of generality, assume that \overline{wz} appears before \overline{ux} . Just after \overline{wz} appears, $pocket(vw)$ is full and $pocket(wz)$ is empty. Hence $pocket(ux)$ is full and $pocket(xy)$ is empty. Let this time be t_1 .

Just after \overline{ux} appears (at some time t_4), $pocket(ux)$ is empty and $pocket(xy)$ is full. By Lemma 4.4, there exists a time t_2 , with $t_1 < t_2 < t_4$, at which $pocket(ux)$ and $pocket(xy)$ are both empty. But at $t_3 = t_4 - \varepsilon$ for some infinitesimal ε , $pocket(ux)$ is full (because the appearance of \overline{ux} is an inner appearance). Thus at times t_1 , t_2 , t_3 , and t_4 , $pocket(ux)$ is successively full, empty, full, and empty. At the instants of transition between full and empty states, the projection of Q 's motion on the normal to \overline{ux} is successively positive, negative, and positive. Hence Q 's motion is not convex. \square

Lemma 4.13. *The number of PP edge appearances on $PGC(P)$ during a single convex motion is $O(n)$.*

Proof. There are at most n edges on all the $\pi_0(a, b)$ paths (edges of type (1)). Lemma 4.9 shows that there are $O(n)$ edges of type (2). Lemma 4.11 shows that the graph of type (3) edges that make an outer appearance is planar, and hence there are $O(n)$ such edges. Similarly, Lemma 4.12 shows that there are $O(n)$ type (3) edges that make an inner appearance. Taken together, these lemmas show that there are $O(n)$ PP edges that appear on $PGC(P)$ during a single convex motion. Finally, Lemma 4.8 shows that each of these edges has at most two appearances during one convex motion. \square

We now consider QQ edges that appear on $PGC(P)$ during a single convex motion. Such edges are edges of the relative convex hull of Q , and hence by applying Lemma 4.13 to QQ edges on $PGC(Q)$ we can show that only $O(n)$ QQ edges appear on $PGC(P)$. However, such an edge might have multiple appearances on $PGC(P)$, even though it remains continuously on $RCH(Q)$ —Lemma 4.13 bounds the number of appearances on $RCH(Q)$, not on $PGC(P)$.

Lemma 4.14. *During a single convex motion, any QQ edge has at most one appearance on $PGC(P)$ that is not also an appearance on the relative convex hull of Q .*

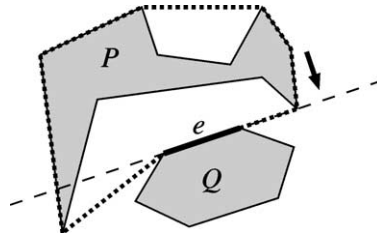


Fig. 15. Edge e is a QQ edge of $RCH(Q)$; it joins $PGC(P)$ because a vertex of P crosses the line supporting e .

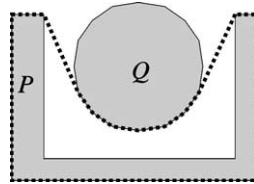


Fig. 16. As Q moves up and down $k/2$ times, $RCH(P)$ changes $\Theta(kn)$ times.

Proof. In this proof we view the relative motion of P and Q from Q 's perspective: P moves relative to Q . Let e be the QQ edge in question. If at time t edge e is already on $RCH(Q)$ and becomes an edge of $PGC(P)$, it means that one endpoint of e belongs to $PGC(P)$ before t , and the other becomes a vertex of $PGC(P)$ at t (by the nondegeneracy assumption). For this to happen, an edge of $PGC(P)$ (a PQ edge) not previously on $PGC(Q)$ must become collinear with e at t . A vertex of P crosses the line supporting e at t , locally moving toward Q . See Fig. 15. For e to disappear from $PGC(P)$ without leaving $RCH(Q)$, the reverse motion must occur: a vertex of P must move across the line supporting e , moving locally away from Q . Thus for e to appear twice on $PGC(P)$ without appearing on or leaving $RCH(Q)$, P must move toward Q , away, and toward again (projected on the normal to e)—a nonconvex motion. \square

Putting Lemma 4.14 together with Lemma 4.13 applied to Q , we have the following corollary.

Corollary 4.15. *The number of QQ edge appearances on $PGC(P)$ during a single convex motion is $O(n)$.*

We can now finish the proof of Theorem 3.1.

Proof of Theorem 3.1. It follows from Lemma 3.3, Lemma 4.13, and Corollary 4.15 that the number of changes to a single $PGC(P)$ during one convex motion is $O(n)$. There are $O(\log n)$ pinned geodesic cycles in the ERGT, and so the total number of changes to the ERGT during one convex motion is $O(n \log n)$. By Lemma 4.1, the relative motion of P and Q can be decomposed into $O(k)$ convex motion fragments. This establishes upper bound part of the theorem.

To see that the worst-case number of changes to the relative convex hull is $\Omega(kn)$ for translational motion of algebraic degree k , consider a U-shaped polygon P with $O(1)$ vertices, and a regular n -gon Q . See Fig. 16. The n -gon Q moves into and out of the cavity of P $k/2$ times, with a motion given by

$y = c \prod_{i=1}^{k/2} (t - i)^2$. The relative convex hull of P gains and loses edges one at a time, going from size $\Theta(1)$ to size $\Theta(n)$ $k/2$ times. Thus the number of combinatorial changes to it is $\Theta(kn)$. \square

5. Bounds for rigid motion

When Q is allowed to rotate as well as translate, the key Lemma 4.4 does not apply. Indeed, we can construct an example of algebraic rigid motion in which the relative convex hull of P and Q (and hence their ERGT) changes quadratically many times, as shown in Fig. 17. In the figure, P is fixed, and Q rotates around the point o . When a tooth of Q crosses a dashed line, which is the extension of an edge on the left convex chain of P , the relative convex hull of P changes combinatorially. Suppose Q has n regularly spaced teeth, and the convex chain of P has n edges and is flat enough that the teeth of Q cross all the dashed lines one after another. Then each tooth of Q causes the relative convex hull to change n times. In total, the relative convex hull changes quadratically many times. Since the relative convex hull of P is the outermost pinned geodesic cycle if we choose the first pinned vertex on the convex hull of P , the ERGT changes quadratically many times as well.

To prove a nearly matching upper bound, we once again consider inner and outer appearances separately. In the remainder of this section, Q undergoes algebraic rigid motion, as described in Section 3, and as usual, P is stationary.

We first relate the outer appearances on the ERGT to some visibility changes in a continuously changing scene.

Lemma 5.1. *Let P be a fixed simple polygon and $p \in \partial P$. If a point q moves along a bounded-degree algebraic path in \mathcal{F}_P , then the visibility between p and q in \mathcal{F}_P changes $O(1)$ times.*

Proof. The region where q is not visible from p is made of disjoint pockets whose lids are collinear with p . Each time q disappears and reappears from view, it has to enter and leave a pocket, and the slope of pq reaches a maximum or a minimum. Since q is moving along an algebraic path of bounded degree, the slope of pq has only $O(1)$ local extrema. There is one special case in which q passes behind P , but this can also happen only $O(1)$ times. \square

Lemma 5.2. *The visibility between any point of ∂P and any point of ∂Q changes $O(1)$ times.*

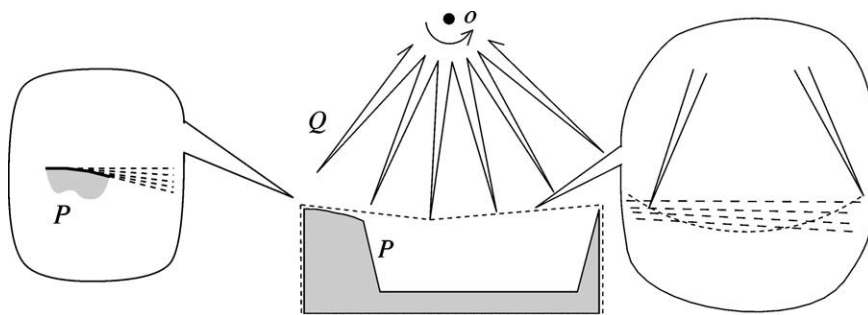


Fig. 17. When Q rotates around the point o , each tooth causes the relative convex hull to change n times.

Proof. Suppose that at time t , q starts (or ceases) to be visible from p . This happens only when there is a vertex r that ceases (or starts) to block the visibility from q to p . If $r \in P$, then the visibility from q to p changes even if we consider only the visibility with respect to P . If $r \in Q$, similarly, the visibility from q to p changes, even considering only the visibility with respect to Q . By Lemma 5.1, there can be only $O(1)$ such events. \square

An interesting observation is that the previous lemma doesn't hold for two points on the same polygon. In Fig. 17, the visibility between a vertex on the left convex chain of P and the tip vertex on the right side of P changes n times as Q makes a full rotation.

Lemma 5.3. *The number of outer appearances of PP and QQ edges on the ERGT is $O(n^2)$.*

Proof. If a PP edge $\overline{p_1 p_2}$ has an outer appearance due to a vertex q as in Fig. 8(b), we charge this outer appearance to the pair (p_1, q) . As q stops being visible from p_1 at that time, such a pair can be charged at most a constant number of times, by Lemma 5.2. The same applies to QQ edges. \square

Surprisingly, the bound on the number of inner appearances is much more involved and requires a lower envelope argument using Davenport–Schinzel sequences [25].

For each convex vertex $q \in \partial Q$, we choose a ray $\vec{q_0}$ that divides the exterior angle at q into two subangles, each less than π . Let P' be a subset of the vertices of P . A vertex $p \in P'$ is an *upper extreme visible vertex* for q in P' if it is visible from q and, among all visible vertices of P' on the same side of $\vec{q_0}$ as p , \vec{qp} makes the smallest angle with $\vec{q_0}$. A vertex p is a *lower extreme visible vertex* if the angle between $\vec{q_0}$ and \vec{qp} is maximized. Informally, an extreme vertex is the “lowest” or “highest” vertex among a set of vertices visible from q (Fig. 18). We have the following characterization of a PQ edge.

Lemma 5.4. *Suppose that \overline{pq} is a PQ edge on $\pi(a, b)$ where $a, b, p \in P$, $q \in Q$. Then p is either a lower extreme visible vertex for q in $C(a, b)$, or an upper extreme visible vertex in $C(b, a) \cap \pi_0(a, b)$.*

Proof. The region $R(a, b)$ has a reflex angle at q —otherwise $\pi(a, b)$ could be shortened. Therefore the two neighbors of q on $\pi(a, b)$ lie on opposite sides of $\vec{q_0}$. Because $\vec{q_0}$ enters $R(a, b)$, the first polygon point intersected by $\vec{q_0}$ must lie on $C(a, b)$. Note that for a vertex $p \in P$ to appear on $\pi(a, b)$, it must belong to either $C(a, b)$ or $C(b, a) \cap \pi_0(a, b)$. (This follows because $R(a, b) \subseteq R_0(a, b)$; the named vertices are exactly those on the boundary of $R_0(a, b)$.)

Consider the situation in which $p \in C(a, b)$, and p is visible from q . (If p is not visible from q , \overline{pq} cannot be an edge on $\pi(a, b)$.) If p is not a lower extreme visible vertex, then there is some vertex $p' \in \pi(a, b)$ such that $\vec{qp'}$ makes a larger angle with $\vec{q_0}$ than \vec{qp} . Edges $\vec{qp'}$ and $\vec{q_0}$ cut $R(a, b)$ into three pieces, with a and b at opposite ends and p in the middle piece—this follows because the vertices of $C(a, b)$ visible from q appear in polygon order. Any path from a to b containing the edge \overline{qp} cannot be the shortest path, as it can be shortcut by $\vec{qp'}$ (Fig. 18(a), with $p' = p_2$).

The shortest path $\pi(a, b)$ is contained inside $R_0(a, b)$, and so we can consider path-finding inside that polygon. The vertices of $R_0(a, b) \cap C(b, a)$ that appear on $\pi(a, b)$ appear in reverse polygon order; the same is true for $\pi_0(a, b)$. The path formed by $\vec{q_0}$ and \vec{qp} splits $R_0(a, b)$ in two, with a and b in opposite sides. If p is not an upper extreme visible vertex for q , then there is some visible vertex $p' \in R_0(a, b) \cap C(b, a)$ closer in angle to $\vec{q_0}$ than p . A path from a to b that uses \overline{qp} must cross $\vec{qp'}$

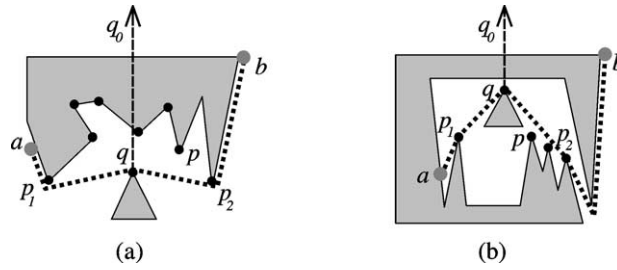


Fig. 18. (a) p_1 and p_2 are lower extreme visible vertices of q in $C(a, b)$. (b) p_1 and p_2 are upper extreme visible vertices of q in $C(b, a) \cap \pi_0(a, b)$.

on the way from q to the path end, and hence is not a shortest path—it can be shortcut by $\overrightarrow{qp'}$ (Fig. 18(b), with $p' = p_2$). \square

By Lemma 5.4, we can bound the number of inner appearances.

Lemma 5.5. *The number of inner appearances on a single pinned geodesic cycle is $O(n\lambda_s(n))$, where $\lambda_s(n)$ is the nearly linear maximum length of a Davenport–Schinzel sequence. The parameter s is a constant depending on the parameters of the motion.*

Proof. We focus on PP edges that appear on a specific geodesic $\pi(a, b)$ in the course of the motion. Note that an inner appearance of a PP edge is due to the disappearance of a PQ edge.

Consider a specific vertex q of ∂Q . For each vertex $p \in C(b, a) \cap \pi_0(a, b)$, plot the angle between \overrightarrow{qp} and $\overrightarrow{q_0}$ as a function of time, but only for those times when p is visible from q . Lemma 5.2 guarantees that each vertex p defines only a constant number of arcs in this plot. By Lemma 5.4, an edge \overrightarrow{qp} is a PQ edge on the pinned geodesic cycle only if p is either on the lower or the upper envelope of these arcs. Likewise, for vertices $p \in C(a, b)$, we plot the angle between \overrightarrow{qp} and the reversal of $\overrightarrow{q_0}$ as a function of time. An edge \overrightarrow{qp} is a PQ edge only if p is on the upper or lower envelope of this plot. For each of the two plots, the upper envelope corresponds to vertices on one side of $\overrightarrow{q_0}$ and the lower envelope to vertices on the other side.

Vertex q causes the appearance of a PP edge on $\pi(a, b)$ when q has two incident PQ edges $\overrightarrow{qp_1}$ and $\overrightarrow{qp_2}$ and those edges become collinear: $\overrightarrow{p_1p_2}$ replaces $\overrightarrow{qp_1}$ and $\overrightarrow{qp_2}$ on $\pi(a, b)$. This occurs only if the angle between $\overrightarrow{q_0}$ and $\overrightarrow{qp_1}$ differs by $\pm\pi$ from the angle between $\overrightarrow{q_0}$ and $\overrightarrow{qp_2}$. That is, q makes a PP edge appear on $\pi(a, b)$ only if one of the four upper/lower envelopes intersects one of the others when shifted vertically by π . Since any two arcs intersect $O(1)$ times (each intersection corresponds to a collinearity between q and two vertices of P), the number of PP appearances is proportional to the number of vertices on the envelopes. Let m be the number of vertices on $C(a, b)$ and $C(b, a) \cap \pi_0(a, b)$, i.e., the number of vertices of $R_0(a, b)$. The number of vertices on the envelopes is $O(\lambda_s(m))$ [25]. Each vertex $p \in P$ appears on at most one path $\pi_0(a, b)$ for which $p \notin C(a, b)$. Therefore, the sum of $m = |R_0(a, b)|$ over all consecutive pinned vertex pairs $a, b \in \partial P$ is $O(n)$. Thus the number of PP edges that q creates is $O(\lambda_s(n))$. Summing over all vertices of Q proves the lemma. \square

Applying Lemma 5.5 to the $O(\log n)$ pinned geodesic cycles of the ERGT gives an $O(n \log n \lambda_s(n))$ bound on the number of inner appearances. Since $\lambda_s(n) = O(n^{1+\varepsilon})$ for any positive ε [25], this bound,

combined with Lemma 5.3's bound on the number of outer appearances, completes the proof of the upper bound part of Theorem 3.2. The lower bound follows from the example at the beginning of this section.

6. Conclusion

We have presented an efficient and responsive KDS for the problem of collision detection between two moving simple polygons in the plane. We believe that the kinetic setting is the framework of choice to approach problems of collision detection, even when the motion plans are not fully known. We recently generalized this structure to the case of multiple moving polygons [1]. We would like to integrate the approach presented in this paper with the distance sensitive approach for convex polygons in [8]. We still do not know whether these ideas can be successfully adapted to three-dimensional non-convex bodies. The most straightforward extension of our ideas to three dimensions leads, unfortunately, to non-polyhedral tilings of the free space.

Acknowledgements

We wish to thank Jorge Stolfi for fruitful discussions and John Bauer for helping to clarify certain cases of the ERGT maintenance while he was implementing it.

References

- [1] P.K. Agarwal, J. Basch, L.J. Guibas, J. Hershberger, L. Zhang, Deformable free-space tilings for kinetic collision detection, *Internat J. Robotics Res.* 21 (3) (2002) 179–197.
- [2] J. Basch, J. Erickson, L. Guibas, J. Hershberger, L. Zhang, Kinetic collision detection between two simple polygons, in: *Proc. 10th ACM-SIAM Sympos. Discrete Algorithms*, 1999, pp. 102–111.
- [3] J. Basch, L. Guibas, J. Hershberger, Data structures for mobile data, *J. Algorithms* 31 (1) (1999) 1–28.
- [4] S. Cameron, Collision detection by four-dimensional intersection testing, in: *Proc. IEEE Internat. Conf. Robot. Autom.*, 1990, pp. 291–302.
- [5] B. Chazelle, H. Edelsbrunner, M. Grigni, L. Guibas, J. Hershberger, M. Sharir, J. Snoeyink, Ray shooting in polygons using geodesic triangulations, *Algorithmica* 12 (1994) 54–68.
- [6] H. Edelsbrunner, E.P. Mücke, Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms, *ACM Trans. Graph.* 9 (1) (1990) 66–104.
- [7] D. Eppstein, J. Erickson, Raising roofs, crashing cycles, and playing pool: Applications of a data structure for finding pairwise interactions, in: *Proc. 14th Annu. ACM Sympos. Comput. Geom.*, 1998, pp. 58–67.
- [8] J. Erickson, L.J. Guibas, J. Stolfi, L. Zhang, Separation-sensitive kinetic collision detection for convex objects, in: *Proc. 10th ACM-SIAM Sympos. Discrete Algorithms*, 1999, pp. 327–336.
- [9] S. Gottschalk, M.C. Lin, D. Manocha, OBB-tree: A hierarchical structure for rapid interference detection, in: *Proc. SIGGRAPH '96, Comput. Graph.*, 1996, pp. 171–180.
- [10] L.J. Guibas, Kinetic data structures—A state of the art report, in: *Proc. 3rd Workshop on Algorithmic Foundations of Robotics*, 1998, pp. 191–209.
- [11] P. Gupta, R. Janardan, M. Smid, Fast algorithms for collision and proximity problems involving moving geometric objects, *Computational Geometry* 6 (1996) 371–391.
- [12] R. Haas, D. Orden, G. Rote, F. Santos, B. Servatius, H. Servatius, D. Souvaine, I. Streinu, W. Whiteley, Planar minimally rigid graphs and pseudo-triangulations, in: *Proc. 19th Annu. ACM Sympos. Comput. Geom.*, 2003, pp. 154–163.

- [13] P.M. Hubbard, Collision detection for interactive graphics applications, *IEEE Trans. Visualization and Computer Graphics* 1 (3) (1995) 218–230.
- [14] P.M. Hubbard, Approximating polyhedra with spheres for time-critical collision detection, *ACM Trans. Graph.* 15 (3) (1996) 179–210.
- [15] D. Kirkpatrick, J. Snoeyink, B. Speckmann, Kinetic collision detection for simple polygons, *Internat. J. Comp. Geom. Appl.* 12 (2002) 3–27.
- [16] D. Kirkpatrick, B. Speckmann, Kinetic maintenance of context-sensitive hierarchical representations for disjoint simple polygons, in: *Proc. 18th Annu. ACM Sympos. Comput. Geom.*, 2002, pp. 179–188.
- [17] J. Klosowski, M. Held, Joseph S.B. Mitchell, K. Zikan, H. Sowizral, Efficient collision detection using bounding volume hierarchies of k -DOPs, *IEEE Trans. Visualizat. Comput. Graph.* 4 (1) (1998) 21–36.
- [18] D.M. Mount, Intersection detection and separators for simple polygons, in: *Proc. 8th Annu. ACM Sympos. Comput. Geom.*, 1992, pp. 303–311.
- [19] M. Pocchiola, G. Vegter, Computing the visibility graph via pseudo-triangulations, in: *Proc. 11th Annu. ACM Sympos. Comput. Geom.*, 1995, pp. 248–257.
- [20] M. Pocchiola, G. Vegter, Pseudo-triangulations: Theory and applications, in: *Proc. 12th Annu. ACM Sympos. Comput. Geom.*, 1996, pp. 291–300.
- [21] M. Pocchiola, G. Vegter, Topologically sweeping visibility complexes via pseudo-triangulations, *Discrete Comput. Geom.* 16 (1996) 419–453.
- [22] M.K. Ponamgi, D. Manocha, M.C. Lin, Incremental algorithms for collision detection between general solid models, in: *Proc. ACM Siggraph Sympos. Solid Modeling*, 1995, pp. 293–304.
- [23] B. Schieber, U. Vishkin, On finding lowest common ancestors: Simplification and parallelization, *SIAM J. Comput.* 17 (6) (1988) 1253–1262.
- [24] E. Schömer, C. Thiel, Efficient collision detection for moving polyhedra, in: *Proc. 11th Annu. ACM Sympos. Comput. Geom.*, 1995, pp. 51–60.
- [25] M. Sharir, P.K. Agarwal, *Davenport–Schinzel Sequences and Their Geometric Applications*, Cambridge University Press, New York, 1995.
- [26] B. Speckmann, C. Tóth, Allocating vertex π -guards in simple polygons via pseudo-triangulations, in: *Proc. 14th ACM-SIAM Sympos. Discrete Algorithms*, 2003, pp. 109–118.
- [27] I. Streinu, A combinatorial approach to planar non-colliding robot arm motion planning, in: *Proc. 41st Annu. IEEE Sympos. Found. Comput. Sci.*, IEEE, 2000, pp. 443–453.
- [28] S. Suri, Minimum link paths in polygons and related problems, PhD Thesis, Dept. Comput. Sci., Johns Hopkins Univ., Baltimore, MD, 1987.
- [29] G.T. Toussaint, Shortest path solves translation separability of polygons, Report SOCS-85.27, School Comput. Sci., McGill Univ., Montreal, PQ, 1985.