Kinetic Collision Detection Between Two Simple Polygons

Julien Basch* Jeff Erickson† Leonidas J. Guibas‡ John Hershberger§ Li Zhang¶

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.

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 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. 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 remains still one of the most time consuming 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 [10]. Swept volumes in space or space-time have also been used towards this goal [2, 8]. 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 [4, 7, 11], 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* [1, 6], 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 animating a proof of its correctness through time. The proof consists of a set of elementary conditions, or *certificates*, based on the kinds of tests performed by ordinary geometric algorithms (CCW 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)

*Computer Science Department, Stanford University, Stanford, CA 94305, and Center for Geometric Computing, Department of Computer Science, Duke University, Durham, NC 27708-0129; jbasch@cs.stanford.edu; http://graphics.stanford.edu/~jbasch/.

†Center for Geometric Computing, Department of Computer Science, Duke University, Durham, NC 27708-0129, and Department of Computer Science, University of Illinois, Urbana, IL 61801-2987; jeffe@cs.uiuc.edu; http://www.uiuc.edu/ph/www/jeffe/. 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 U.S. Army Research Office MURI grant DAAH04-96-1-0013.

‡Computer Science Department, Stanford University, Stanford, CA 94305; guibas@cs.stanford.edu; http://graphics.stanford.edu/~guibas/. Research partially supported by National Science Foundation grant CCR-9623851 and by U.S. Army Research Office MURI grant DAAH04-96-1-0007.

§Mentor Graphics Corp., 8005 SW Boeckman Road, Wilsonville, OR 97070, USA; john_hershberger@mentorg.com.

¶Computer Science Department, Stanford University, Stanford, CA 94305; lizhang@cs.stanford.edu; http://graphics.stanford.edu/~lizhang/.

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,
- 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* [14] 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 [3]. (This latter structure was also used by Mount [9] 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 [6]) 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 [13]. 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.

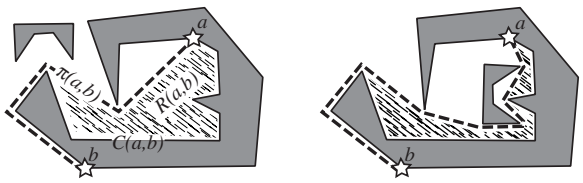

Figure 1. The geodesic from a to b for two placements of the small polygon. A shortest path endpoint is indicated by a star.

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.

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. Section 3 presents the event bounds for the two models of motion considered. Section 4 concludes with plans for further work.

2 Certification and Maintenance

We denote the boundary of a simple polygon P by ∂P, and take 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, 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 (resp. ∂Q), it is called a PP edge (resp. 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 $\overline{uv}$. Finally, we denote by $R(a, b)$ the open region delimited by $C(a, b)$ and $\pi(a, b)$ (Figure 1).

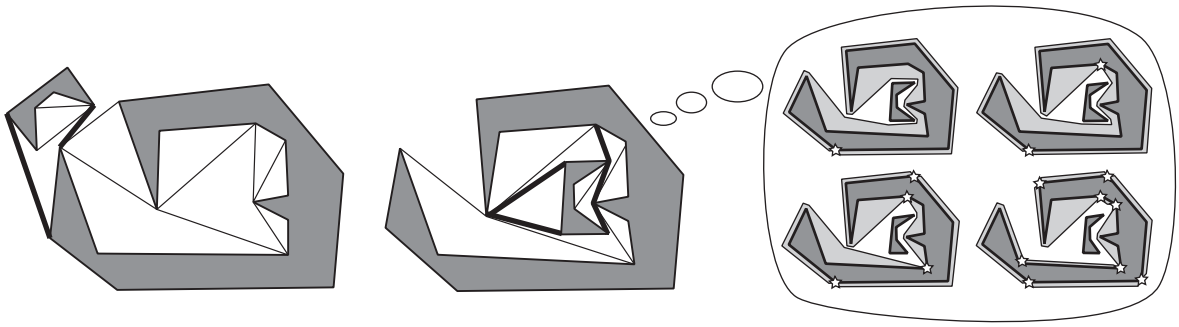

Figure 2. Planar maps induced by the ERGT of two polygons, with the PQ edges in bold. The right polygon is dreaming about its first four pinned geodesic cycles.

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 $R(a, b) \subseteq R(c, d)$.*

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. We say that the vertices of S are *pinned*.

2.1 External Relative Geodesic Triangulation

Proposition 2.1 allows us to define a planar map. Let T_P (resp. T_Q) be a binary tree whose leaves are the edges of P (resp. 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 ν of T, we associate the geodesic between the two extreme vertices of the subtree rooted at ν.

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 adds an inner common tangent and an outer common tangent to the edges already defined by the geodesics associated with the other nodes.

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 (Figure 2). 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.

Observation 2.1. *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.*

The ERGT has a number of properties, which are straightforward generalizations of those of the geodesic triangulation of [3]. Consider a node ν 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). 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. For any line segment in $\mathcal{F}$, the sequence of nodes of T corresponding to the faces whose interior 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 2.2. *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 $t < t_f$.*

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.

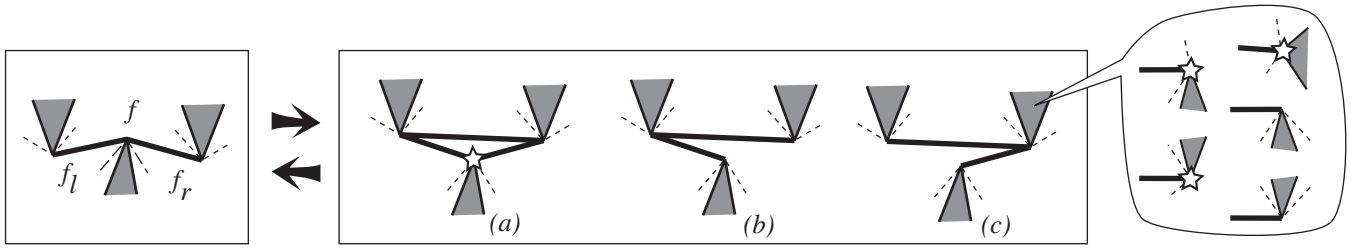

Figure 3. The failure of a convex certificate (right to left), and of a reflex certificate (left to right). The polygon edges incident to the extreme vertices may lie on either side of the edge involved in the event (right inset).

2.2 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 2.3. *In an ERGT of P and Q with n vertices in total, the active set has $O(\kappa \log n)$ certificates, where κ is the size of a minimum link separator of P and Q.*

Proof: A line segment not contained in P can cross a geodesic between two vertices of P at most once. Therefore, it can cross a pinned geodesic cycle at most twice. A separator has to cross all PQ edges, so there are at most twice as many such edges as there are segments in the separator. Finally, there are at most four certificates per PQ edge in the active set. □

2.3 Maintenance of the ERGT

We now assume that we have two moving simple polygons P, Q. We assume transversality in space/time, i.e., there are at most three vertices collinear at any given time. This assumption also implies non-degeneracy of P and Q separately: no three vertices of P are collinear, and the same holds for Q. We further assume that, given the knowledge of the motions of P and Q, we can compute the failure time of any certificate in $O(1)$.

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 Figure 3. The failure of a convex certificate (right to left in Figure 3) is easy to handle, as there is only one possible resulting map. However, when a reflex certificate fails, we need to choose between three possible resulting maps. We describe below how this can be done with the help of the tree on which the ERGT is based.

Consider the situation on the left of Figure 3. 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. Let ν, ν_ℓ, ν_r be the three nodes associated with our faces.

Proposition 2.4. *The relative positions of nodes ν, ν_ℓ, ν_r are different in each case of Figure 3. 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 ν_r is between ν and ν_ℓ,
(c) the three nodes are on a common path, and ν_ℓ is between ν and ν_r,

Proof: (sketch) As we mentioned earlier, 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 around the vertex v to prove the claim. For instance, in case (b), we can draw a segment that crosses f, f_r, f_ℓ in this order. □

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 omitted from this version of the paper.

Our kinetic data structure maintains the ERGT, and, for each face, a pointer to its node in T (note that T

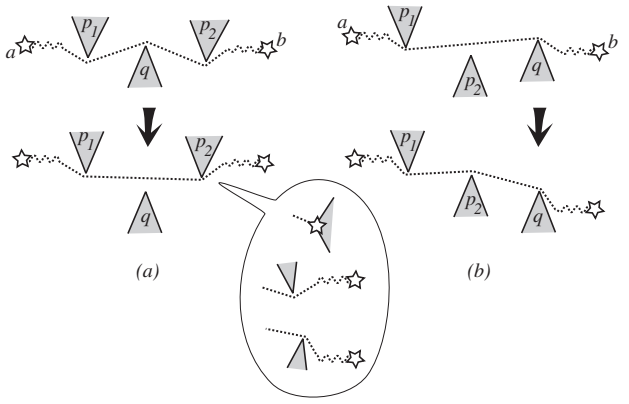

Figure 4. (a) An inner appearance and (b) an outer appearance of the oriented edge $\overline{p_1p_2}$ along $\pi(a, b)$ as Q moves downwards. In each case, the polygon edges incident to the extreme vertices may lie on either side of the connecting edge (inset).

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 2.4 to decide how to handle the event. This can be done in $O(\log \log n)$ with a constant number of least-common-ancestor queries. 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 Figure 2, and to imagine how the structure of the ERGT would change if the small polygon were to move around.

3 Efficiency

In this section, we analyze the number of combinatorial changes to the ERGT 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 is 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.

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 $O(kn \log n)$.*

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 $O(n^{2+\epsilon})$ for any $\epsilon > 0$, where the hidden constant depends on the exact parameters of the motion.*

Like the bounds for other kinetic data structures [1, 5, 6], these bounds do not actually require the motion to be algebraic, but only that it satisfy certain combinatorial conditions. However, it is *not* 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.

An event, i.e., a combinatorial change to the ERGT, always involves the appearance or disappearance of a PP or a QQ edge. It is clear that it is sufficient to bound 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*). Figure 4 illustrates these two events for specific positions of the vertices of P.

3.1 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 range $t_1 \le t \le t_2$. Thanks to the following lemma, we can assume without loss of generality that the motion of Q is convex.

Lemma 3.3. *If the coordinates of $q(t)$ are polynomials of degree k, then $q(t)$ can be decomposed into $O(k)$ convex motion fragments.*

We will not prove all the cases of Theorem 3.1. Rather, we will consider only the number of appearances of PP edges on a single pinned geodesic cycle of

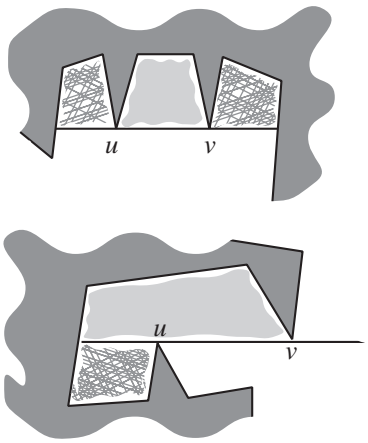

Figure 5. Two instances of an oriented edge $\overline{uv}$. The pocket it defines is in gray, and the neighbor pockets are hatched.

P, which we call PGC in the sequel. This case illustrates our main arguments, and symmetry considerations, with a careful case analysis, take care of the other possibilities.

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 $C(a, b)$.

Let $\overline{uv}$ be an oriented PP edge. We extend it beyond v until it hits P, and denote this segment (or ray, or empty set) by $\overrightarrow{uv}$. We call $\overrightarrow{uv}$ the *extension of* $\overline{uv}$ *at* v. The extension $\overrightarrow{vu}$ is defined symmetrically. The union of $\overline{uv}$ and its two extensions is denoted $\overleftrightarrow{uv}$. It cuts $\mathcal{F}_P$ in up to four components (at most four, by non-degeneracy).

Definition 3.1. *The* pocket *of the oriented edge* $\overline{uv}$ *(denoted* $pocket(uv)$*) is the component of* $\mathcal{F}_P \setminus \overleftrightarrow{uv}$ *that is locally to the left of* $\overline{uv}$*. If* $\overrightarrow{uv}$ *is a finite segment, the* neighbor pocket of v *is the component with* $\overrightarrow{uv}$ *on its boundary, and* $\overline{uv}$ *not on its boundary. The neighbor pocket of* u *is defined symmetrically (Figure 5).*

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

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

A neighbor pocket is also a pocket in its own right. The endpoints of the extension segment that defines it may not be vertices, but the definition of a pocket does not require them to be.

Lemma 3.4. *Let* P_1 *and* P_2 *be two finite, disjoint pockets in* 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: (sketch) The proof is based on the 'depth of penetration' of a full pocket. If P_1 is already full when P_2 becomes full, then P_1 will stay full until after P_2 becomes empty again, because Q sticks farther into P_1 than into P_2. Hence when P_1 first becomes empty, P_2 is empty as well. □

The definitions of the pocket and neighbor pockets allow us to characterize the placements of Q for which a PP edge can be present on PGC. In the following discussion, we will assume that a and b are consecutive pinned vertices along ∂P.

Lemma 3.5. *If a* PP *edge* e *is present on* PGC *for some position of* Q*, 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. Furthermore,* $pocket(e)$ *is empty and has no pinned vertex not collinear with* e*.*

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.

If the extension is finite and the neighbor pocket N is finite and unpinned, then $\pi(a, b)$ crosses the lid of N twice. But if Q does not enter N, $\pi(a, b)$ can be shortcut using a segment of the pocket lid. Hence N must be full.

For each endpoint v of e, either the lid of $pocket(e)$ ends at v, or $\pi(a, b)$ bends away from $pocket(e)$ at v. In either case, $\pi(a, b)$ separates from the lid at v. The only intersection of $\pi(a, b)$ with the lid of $pocket(e)$ is exactly the edge e. (If there were two intersections with the lid of a neighbor pocket, $\pi(a, b)$ could be shortcut.) It follows that $\pi(a, b)$ does not enter $pocket(e)$. However, if Q entered $pocket(e)$, or if $pocket(e)$ contained a pinned vertex not collinear with e, then $\pi(a, b)$ would be forced to enter $pocket(e)$, a contradiction. □

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.

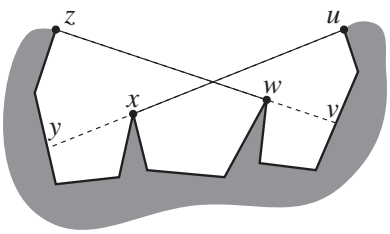

Figure 6. Two restricted PP edges that cross.

Lemma 3.6. *An oriented PP edge has at most one inner appearance and one outer appearance on PGC during a single convex motion.*

Proof: (sketch) 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.

Likewise, if e has an outer appearance, Q crosses one of the extensions of e. This motion may either be toward $pocket(e)$ or away from $pocket(e)$, depending on the configuration of e and its extensions. Nevertheless, a case analysis shows that before e can have a second outer appearance, the projection of Q's motion on the normal to e will have two local extrema. □

To bound the number of appearances of PP edges on PGC, we bound the number of appearances of each of three types of edges: (1) edges that belong to $\pi_0(a, b)$ for some pair of consecutive pinned vertices a and b (the lemma above says that there are linearly many such appearances); (2) edges not of type (1) that have at least one backward-facing neighbor pocket (we prove in the full paper that such edges belong to one of two n-vertex trees, so there are $O(n)$ of them); and (3) edges not of type (1) that have only forward-facing neighbor pockets.

We now focus our attention on PP edges of PGC that do not belong to any $\pi_0(a, b)$, and that have only forward-facing neighbor pockets. We call these edges *restricted PP edges*.

Lemma 3.7. *If two restricted PP edges of PGC 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 Figure 6. We can show that $\pi(a, b) \subseteq R_0(a, b) \cup \pi_0(a, b)$; we omit the details from this version of the paper. Hence $\overline{ux}$ and $\overline{wz}$ both belong to $\pi(a, b)$ for the same consecutive pair of pinned vertices a and b, albeit at different times.

As noted in the proof of Lemma 3.5, if e is a PP edge of PGC, then $\pi(a, b)$ does not enter $pocket(e)$. This implies that $pocket(e)$ 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 3.5. 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 $pocket(vw)$ and $pocket(xy)$. Because $\overline{ux}$ and $\overline{wz}$ are restricted PP edges, these neighbor pockets are forward-facing. Each neighbor pocket is therefore finite. It follows that $pocket(vw) \subset pocket(ux)$ and $pocket(xy) \subset pocket(wz)$. □

Lemma 3.8. *No two restricted PP edges that have outer appearances on PGC during a single convex motion of Q can cross.*

Proof: If a restricted PP edge e has an outer appearance on PGC, it follows from Lemma 3.5 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).

Suppose that the two crossing edges are $\overline{ux}$ and $\overline{wz}$, as shown in Figure 6, and $\overline{wz}$ appears before $\overline{ux}$. When $\overline{wz}$ appears, some vertex of Q crosses $\overleftrightarrow{wz}$ from above to below. Just after $\overline{wz}$ appears, $pocket(vw)$ is full and $pocket(wz)$ is empty. When $\overline{ux}$ appears, $pocket(xy)$ is full and $pocket(ux)$ is empty, implying that $pocket(vw)$ is empty and $pocket(wz)$ is full.

By Lemma 3.4, $pocket(vw)$ becomes empty before $pocket(wz)$ becomes full. Hence some vertex of Q crosses $\overleftrightarrow{wz}$ from below, and another vertex later crosses $\overleftrightarrow{wz}$ from above. 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. □

The proof of the corresponding lemma for inner appearances is similar, but easier.

Lemma 3.9. *The number of PP edge appearances on PGC is $O(n)$ during a single convex motion.*

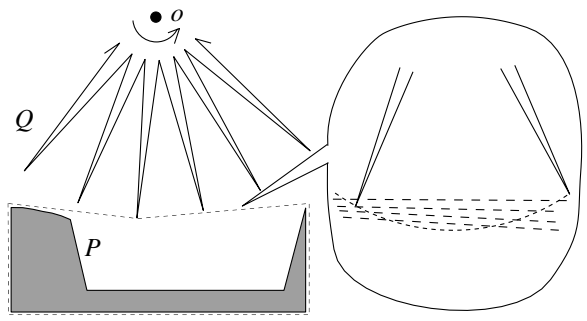

Figure 7. When Q rotates around the point o, each tooth causes the relative convex hull to change n times.

Proof: (sketch) The graph of all outer appearances of restricted PP edges is planar (Lemma 3.8), and hence of linear size; each edge can appear at most once (Lemma 3.6). The same argument applies to the inner appearances. Appearances of non-restricted edges have been bounded separately. □

This concludes the part of the proof of Theorem 3.1 that we are able to present in this proceedings.

3.2 Bounds for Rigid Motion

When Q is allowed to rotate as well as translate, the key Lemma 3.4 fails. 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 Figure 7. In the figure, P is fixed, and Q rotates around the point o. Once 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. We can make Q have n regularly spaced teeth, and the convex chain of P flat enough so 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 separately inner and outer appearances. In the remainder of this section, Q undergoes algebraic rigid motion, as described at the beginning of this section, 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 3.10. *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: (sketch) 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. □

Lemma 3.11. *The visibility between any point of ∂P and any point of ∂Q changes $O(1)$ 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 3.10, there can be $O(1)$ such events. □

An interesting observation is that the previous lemma doesn't hold for two points on the same polygon. In Figure 7, 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 3.12. *The number of outer appearances of PP and QQ edges on the ERGT is $O(n^2)$.*

Proof: If a PP edge $\overline{p_1p_2}$ has an outer appearance due to a vertex q as in Figure 4(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 3.11. The same applies to QQ edges. □

The bound on the number of inner appearances is surprisingly much more involved and requires a lower envelope argument.

Consider a convex vertex $q \in Q$ and a subset $P' \subset P$. For two vertices $p_1, p_2 \in P'$ that are visible from q, consider the cone $\mathcal{C}$ spanned by $\overline{qp_1}$ and $\overline{qp_2}$. If this cone contains Q locally at q and all the other vertices of P' visible from q belong to $\mathcal{C}$ (resp. don't belong to $\mathcal{C}$), we say that p_1 and p_2 are *upper (resp. lower) extreme visible vertices* for q in P'. Informally, an extreme vertex is the "lowest" or "highest" vertex among a set of vertices visible from q (Figure 8). We have the following characterization of a PQ edge.

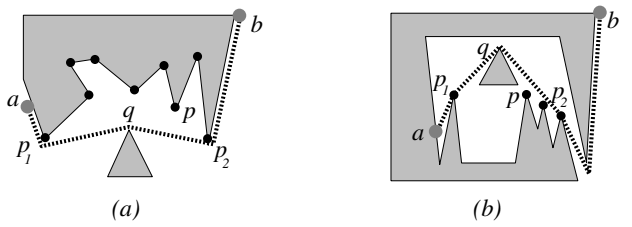

Figure 8. (a) p_1 and p_2 are lower extreme visible vertices of q in $C(a,b)$. (b) p_1 and p_2 are upper extreme visible vertices of q in $C(b,a) \cap \pi_0(a,b)$.

Lemma 3.13. *Suppose that $\overline{pq}$ is a PQ edge on $\pi(a,b)$ where $a,b,p \in P$, $q \in Q$, and $p \notin \{a,b\}$. Then p is either a lower extreme vertex for q in $C(a,b)$, or an upper extreme vertex in $C(b,a) \cap \pi_0(a,b)$.*

Proof: (sketch) First, note that for a vertex $p \in P$ to appear on $\pi(a,b)$, it must belong to either $C(a,b)$ or $C(b,a) \cap \pi_0(a,b)$. Consider the situation in which $p \in C(a,b) \setminus \{a,b\}$, and p is visible from q. (If p is not visible from q, $\overline{pq}$ cannot be an edge on $\pi(a,b)$.) If there exist two vertices p_1, p_2 on $C(a,b)$ so that p is outside of the cone bounded by $\overline{qp_1}$ and $\overline{qp_2}$, then p is inside the region bounded by $C(p_1,p_2)$, $\overline{qp_1}$, and $\overline{qp_2}$. Therefore, any path from a to b containing the edge $\overline{pq}$ cannot be the shortest path as it can be shortcut by either $\overline{qp_1}$ or $\overline{qp_2}$ (Figure 8(a)).

A similar argument applies to the case when $p \in C(b,a) \cap \pi_0(a,b)$ (Figure 8(b)). □

By Lemma 3.13, we can bound the number of inner appearances.

Lemma 3.14. *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: (sketch) 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, and let ℓ be the support line of one of the edges adjacent to q. For each vertex $p \in C(a,b)$, plot the angle between $\overline{qp}$ and ℓ as a function of time, but only for those times when p is visible from q. Lemma 3.11 guarantees that each vertex p defines only a constant number of arcs in this plot. By Lemma 3.13, an edge $\overline{qp}$ is a PQ edge on the pinned geodesic cycle only if p is either on the lower or the upper envelope of these arcs. Moreover, the disappearance of an edge pq from $\pi(a,b)$ happens exactly when p ceases to be extreme for q. Thus, the number of appearances is just the number of vertices on the upper and lower envelopes of a set of $O(n)$ monotone arcs, any pair of which intersects $O(1)$ times (since each intersection point corresponds to a collinearity between q and two vertices of P). The bound now follows from standard lower-envelope results [12].

A similar argument applies when $p \in C(b,a) \cap \pi_0(a,b)$, and, with extra care, when the appearing PP edge is between a vertex in $C(a,b)$ and one in $C(b,a) \cap \pi_0(a,b)$.

The bound is summed over all vertices of Q. □

4 Conclusion

We have presented an efficient and responsive KDS for the problem of collision detection between two moving simple polygons in the plane. This structure has been implemented and will be compared with more traditional methods.

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. In two dimensions, we would like to integrate the approach presented in this paper with the distance sensitive approach for convex polygons in the companion paper [5], and to generalize the structure to the case of multiple moving polygons. We still do not know whether these ideas can be successfully adapted to three-dimensional non-convex bodies. The most straighforward extension of our ideas to three dimensions leads, unfortunately, to non-polyhedral tilings of the free space.

Acknowledgments: 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] J. Basch, L. Guibas, and J. Hershberger. Data structures for mobile data. *Proc. 8th ACM-SIAM Sympos. Discrete Algorithms*, pp. 747–756. 1997.

[2] S. Cameron. Collision detection by four-dimensional intersection testing. *Proc. IEEE Internat. Conf. Robot. Autom.*, pp. 291–302. 1990.

[3] B. Chazelle, H. Edelsbrunner, M. Grigni, L. Guibas, J. Hershberger, M. Sharir, and J. Snoeyink. Ray shooting in polygons using geodesic triangulations. *Algorithmica* 12:54–68, 1994.

[4] D. Eppstein and J. Erickson. Raising roofs, crashing cycles, and playing pool: Applications of a data structure for finding pairwise interactions. *Proc. 14th Annu. ACM Sympos. Comput. Geom.*, pp. 58–67. 1998.

[5] J. Erickson, L. J. Guibas, J. Stolfi, and L. Zhang. Separation-sensitive kinetic collision detection for convex objects. These proceedings, 1998.

[6] L. J. Guibas. Kinetic data structures — A state of the art report. *Proc. 3rd Workshop on Algorithmic Foundations of Robotics*, p. to appear. 1998.

[7] P. Gupta, R. Janardan, and M. Smid. Fast algorithms for collision and proximity problems involving moving geometric objects. *Comput. Geom. Theory Appl.* 6:371–391, 1996.

[8] P. M. Hubbard. Collision detection for interactive graphics applications. *IEEE Trans. Visualization and Computer Graphics* 1(3):218–230, Sept. 1995.

[9] D. M. Mount. Intersection detection and separators for simple polygons. *Proc. 8th Annu. ACM Sympos. Comput. Geom.*, pp. 303–311. 1992.

[10] M. K. Ponamgi, D. Manocha, and M. C. Lin. Incremental algorithms for collision detection between general solid models. *Proc. ACM Siggraph Sympos. Solid Modeling*, pp. 293–304. 1995.

[11] E. Schömer and C. Thiel. Efficient collision detection for moving polyhedra. *Proc. 11th Annu. ACM Sympos. Comput. Geom.*, pp. 51–60. 1995.

[12] M. Sharir and P. K. Agarwal. *Davenport-Schinzel Sequences and Their Geometric Applications.* Cambridge University Press, New York, 1995.

[13] S. Suri. *Minimum link paths in polygons and related problems.* Ph.D. thesis, Dept. Comput. Sci., Johns Hopkins Univ., Baltimore, MD, 1987.

[14] G. T. Toussaint. Shortest path solves translation separability of polygons. Report SOCS-85.27, School Comput. Sci., McGill Univ., 1985.