A Geometric Predicate for Linear Time Collision Detection of Polygonal Objects

Nicole Debowski, René Weller and Gabriel Zachmann

University of Bremen

1. Introduction

From a theoretical point of view, finding collisions between a pair of polygonal objects is relatively easy: We simply have to check each triangle of one object against all triangles of the other object. The running-time of this naive algorithm is quadratic in the number of triangles. Obviously, a quadratic running-time for objects consisting of millions of polygons is not an option in practice. Hence, a lot of work has been spent developing sophisticated heuristics to reduce the number of required triangle tests, for instance by using bounding volume hierarchies to early prune parts of the objects that cannot collide. For most cases and configurations these heuristics work reasonably. Unfortunately, the quadratic bound on the number of intersections is not an overestimated upper bound, but it can be really met by certain objects like the Chazelle polyhedron: Each triangle of one Chazelle polyhedron can intersect all triangles of another Chazelle polyhedron. Actually, this worst-case does not happen very often, but it is impossible to foresee in advance *when* it happens. In physical simulations we will simply notice a little stuttering in the framerate, but in case of time-critical applications like robotics or haptics, exceeding a certain time budget can damage expensive devices or even hurt people.

In contrast to many publications about the heuristics used in practical collision detection, the work on the theoretical side is relatively sparse. There exists some observations about special objects like convex polyhedra. For instance, [DK90] used a hierarchical representation of convex polyhedra to show that the distance between two of them can be computed in $\mathcal{O}(\log(\mid P \mid) \cdot \log(\mid Q \mid))$ with $\mid P \mid$ and $\mid Q \mid$ being the number of faces of P and Q respectively. If closest features of polyhedra based on voronoi regions are considered [LC91], the worst-case running time for finding the distance is linear. For translations of convex polyhedra the running time is $\mathcal{O}(n^{\frac{8}{5}+\varepsilon})$ [ST95a] and $\mathcal{O}(n^{\frac{5}{3}+\varepsilon})$ for rotational movements of at most the second degree. Later, a generalization for more flexible movements in $o(n^2)$ have been made [ST95b]. For pairs of general polygonal objects, [WKZ06] showed an expected running time of $\mathcal{O}(n)$ or $\mathcal{O}(\log(n))$, depending on the overlap of the root bounding volumes and the diminishing factor of the AABB hierarchy that was used in the proof. However, the running-time depends on the respective bounding volume hierarchy and on the configuration of the objects, i.e. their position and orientation, not on the object itself. When using other object representations instead of polygons, e.g. sphere packings, it it possible to prove a linear complexity for the number of collision [WFZ13].

We define a simple geometric predicate that allows us to identify objects that may lead to a quadratic number of intersecting polygons like the Chazelle polyhedron. We use this predicate to prove a linear worst-case bound on the number of collisions for objects that fulfill it. Our predicate depends only on the objects themselves and not on the configuration of a pair of objects, i.e. on their motion. Moreover, our proof implies a novel algorithm for the collision detection that almost realizes a linear worst-case running time. Our algorithm can be easily parallelized: This results in an even *constant* parallel running-time by using only a linear number of processors. It does not require any complicated pre-processing or data structures, hence, it is perfectly suited also for deformable objects.

2. Our Geometric Predicate

In this section, we present the theoretical basis of our novel linear time collision detection method. Firstly, we consider only triangulated objects. However, we will extend our definitions and theorems to arbitrary polygons later. We start with the definition of our criterion that allows us to prove a linear number of intersections for objects that fulfill it. It basically relays on a simple observation: What makes the analysis of the number of potentially colliding triangles so atrocious is mainly the embedding of 2D objects, the triangles, into a 3D world, because it allows us to stack an infinite number of 2D triangles into a small 3D volume. Obviously, this is an artificial scenario. Hence, our definition aims to avoid these worst case scenarios by, in principle, assigning a certain volume to each triangle.

Definition 2.1 Let $t \in A$ be a triangle in a triangle set A and $k > 0$ some constant. Let s be a sphere with radius $\frac{r}{2}$, where r is the radius of the the smallest enclosing sphere of t. We call t *k-free* if $|\{t_j \in A | r \leq r_j \text{ and } t_j \cap (t \oplus s) \neq \emptyset\}| < k$, where r_j is radius of the smallest enclosing sphere of triangle t_j and $t \oplus s$ is the Minkowski sum of s and t.
Accordingly, we call the whole set of triangles A *k-free*, if all triangles $t_i \in A$ are *k-free*.

In other words, a triangle t is k-free if there are less than k triangles of A with a larger minimum enclosing sphere that intersect the object that results from sweeping a sphere around t. In the following, we will call those triangles *larger* triangles in order to improve the readability. More precisely, let t_i and t_j be two triangles with minimum enclosing spheres s_i and s_j. Let r_i be the radius of s_i and r_j be the radius of s_j. Then we say t_i is *larger* than t_j if $r_i \geq r_j$.

The main idea behind the definition of our criterion is that each triangle occupies a certain amount of its environment more or less exclusively. This guarantees that a single triangle cannot intersect too much larger triangles of a k-free triangle set:

Lemma 2.1 Let A be a k-free set of triangles and let $t \notin A$ be an arbitrary triangle that is not included in A. Then t intersects at most a constant number of larger triangles $t_j \in A$. More precisely, the number of intersections between t and larger triangles $t_j \in A$ is at most $3k$.

The proof of this lemma relays on a simple geometric observation:

Lemma 2.2 Let t be a triangle in 2D and c its minimum enclosing circle with radius r. Then we need at most three circles of radius $\frac{r}{2}$ to completely cover t.

Proof Note, c is not necessarily the circumcircle of the triangle. This is only true for acute triangles, i.e., all three angles are less than 90°. In this case, the center of the circle corresponds to the intersection point of the three perpendicular bisectors and it is located *inside* the triangle. In this case, we can subdivide t into six disjoint sub-triangles that are all *right* triangles: Their hypotenuses are the lines from the center of c to the vertices and the catheti are the sides of the original triangle and the perpendicular bisectors. The length of the hypotenuses of all the triangles is obviously r. Assume a circle with diameter r around each hypotenuse with the center located in the center of the respective hypotenuse. Due to Thales' theorem, these circles have to pass through the intersections between the perpendicular bisectors and the original triangles' sides, because the sub-triangles are right triangles. Consequently, each of these three circles with diameter r covers two of the sub-triangles completely. Overall, we found a complete covering of t by three circles of radius $\frac{r}{2}$. The proof for obtuse triangles is very similar and omitted here. □

With this lemma we can finish the proof of Lemma 2.1:

Proof Let c be the minimum enclosing circle of t in the plane that is spanned by t and let r be the radius of c. We construct a circle covering of t according to Lemma 2.2. We claim, that there are can be at most k triangles $t_i \in A$ intersecting a circle of radius $\frac{r}{2}$. Assume that there are $k+1$ triangles intersecting a circle of radius $\frac{r}{2}$. We arbitrarily chose t_a as on of of these triangles. By definition, the radius r_a of the minimum enclosing sphere of t_a has at most the same size as r. Hence, there have to be k triangles in $t_a \oplus s_a$, where s_a is a sphere with radius $\frac{r_a}{2}$. This is a contradiction to A is k-free. □

Theorem 2.3 Let A and B be two k-free sets of triangles. Then the total number of colliding triangles of A and B is in $O(n)$, where n is the number of triangles in A and B. More precisely, the number of intersections is at most $3nk$.

Proof We test each triangle of A against all larger triangles of B and vice versa. For each of these tests we get at most $3k$ intersections according to Lemma 2.1. Moreover, we find all pairs of colliding triangles, because either of the triangles in a pair of intersecting triangles must be larger. Overall, we get at most $3nk$ intersections. □

Unfortunately, Lemma 2.2 does not necessarily hold for arbitrary polygons. However, in order to get an upper bound we can cover the whole circumcircle of the polygon with circles that have half of its radius. This is possible with at most 7 of these spheres. This results in a slightly worse upper bound for arbitrary polygon soups:

Theorem 2.4 Let A and B be two k-free sets, each consisting of n polygons. Then the total number of colliding polygons of A and B is in $O(n)$. More precisely, the number of intersections is at most $9nk$.

3. Applications and Future Work

We can use the theoretical foundation to define an algorithm that almost realizes this linear worst-case running-time. The main idea is similar to the proof of Theorem 2.3: We test polygons only against larger polygons. A hierarchical grid allows a fast identification of possible colliding polygons. In principle, we simply double the cell size in each level of the hierarchy and assign the polygons with respect to their minimum enclosing sphere to the respective levels. This hierarchical grid adds a logarithmic factor to the running time. Fortunately, this factor depends only on the ratio between the largest polygon to the smallest polygon in each object, but not on the number of polygons. This algorithm can be easily parallelized by simply checking all polygons in parallel.

In practice, for instance in time-critical collision detection, it is essential that the upper bound on the number of collisions is as tight as possible in order to not waste computation power. In our predicate, we simply chose the radius r of the minimum enclosing sphere of each polygon. However, all the proofs hold for all $0 < \varepsilon \leq r$. Obviously, a smaller ε would result in smaller values for k. On the other hand, we would require a larger number of circles to cover the polygon. An optimal choice of ε remains an interesting question.

Another avenue for future works would be to find a geometric predicate for the opposite direction. Until now, our predicate defines only a sufficient but not a necessary criterion, i.e., it is able to exclude objects with a quadratic worst-case complexity like the Chazelle polyhedron, but there can be also objects that do not fulfill it but still have a linear number of collisions.

References

[DK90] DOBKIN D. P., KIRKPATRICK D. G.: Determining the separation of preprocessed polyhedra – a unified approach, 1990. 1

[LC91] LIN M., CANNY J.: A fast algorithm for incremental distance calculation. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on* (Apr 1991), pp. 1008–1014 vol.2. 1

[ST95a] SCHÖMER E., THIEL C.: Efficient collision detection for moving polyhedra. In *In Proc. 11th Annu. ACM Sympos. Comput. Geom* (1995), pp. 51–60. 1

[ST95b] SCHÖMER E., THIEL C.: Subquadratic algorithms for the general collision detection problem. In *Universität Münster* (1995), pp. 95–101. 1

[WFZ13] WELLER R., FRESE U., ZACHMANN G.: Parallel collision detection in constant time. In *Virtual Reality Interactions and Physical Simulations (VRIPhys)* (Lille, France, Nov. 2013), Eurographics Association. 1

[WKZ06] WELLER R., KLEIN J., ZACHMANN G.: A model for the expected running time of collision detection using AABB trees. In *Eurographics Symposium on Virtual Environments (EGVE)* (Lisbon, Portugal, 8–10 May 2006), Hubbold R., Lin M., (Eds.). 1