Static and Kinetic Geometric Spanners with Applications

Menelaos I. Karavelas Leonidas J. Guibas

Graphics Lab., Computer Science Dept., Stanford University
Stanford, CA 94305, U.S.A.
{karavelas, guibas}@cs.stanford.edu

Abstract

It is well known that the Delaunay Triangulation is a spanner graph of its vertices. In this paper we show that any bounded aspect ratio triangulation in two and three dimensions is a spanner graph of its vertices as well. We extend the notion of spanner graphs to environments with obstacles and show that both the Constrained Delaunay Triangulation and bounded aspect ratio conforming triangulations are spanners with respect to the corresponding visibility graph. We also show how to kinetize the Constrained Delaunay Triangulation. Using such time-varying triangulations we describe how to maintain sets of near neighbors for a set of moving points in both unconstrained and constrained environments. Such nearest neighbor maintenance is needed in many virtual environments where nearby agents interact. Finally, we show how to use the Constrained Delaunay Triangulation in order to maintain the relative convex hull of a set of points moving inside a simple polygon.

1 Introduction

Let G be a connected n-vertex graph with arbitrary positive edge weights. A subgraph G' is a *t-spanner* if for any pair of vertices, their distance in G' is at most t times longer than their distance in G. The value t is the *stretch factor* associated with G'. Spanner graphs have several applications. They appear as the underlying graph structure in distributed systems and communication networks [2, 18], as well as in biology [3]. There are also works that deal with the problem of computing sparse spanner graphs in the context of points in Euclidean spaces [1, 7, 9, 8, 13, 14, 15].

A use of spanners of particular interest to us is for nearest neighbor queries. Given a reference point in a graph, we can perform a breadth first search on the associated spanner and prune the search using the current distance along the spanner and the known stretch factor. In the physical world where motion is invariably present, we may be interested in maintaining nearest neighbors of certain or all the nodes as the underlying graph evolves over time. Indeed, the behavior of many physical or social systems can be modeled in terms of short-range interactions between the nodes, containment of some nodes by other groups of nodes, etc.

In this paper we deal with the relationship between bounded aspect ratio triangulations and spanner graphs for a set of geometric points. First, we show that bounded aspect ratio triangulations in two and three dimensions are spanners with respect to the complete graph induced by the Euclidean distance between the points.

Second, we extend the notion of spanners for environments with obstacles. More specifically, if G is a planar straight-line graph (PSLG), then the visibility graph $\mathcal{V}(G)$ of G is the graph that consists of all the edges of G, as well as all the edges between points in G that do not properly intersect edges of G. Using $\mathcal{V}(G)$ we can define what we call the *geodesic distance* between two points in G, which is the length of the shortest path in $\mathcal{V}(G)$ between the two points. We show that any bounded aspect ratio triangulation that conforms with G is a spanner, and moreover that the Constrained Delaunay Triangulation (CDT) is also a spanner, with respect to this geodesic distance.

Next, we deal with the case of moving points and obstacles. We discuss how to maintain the CDT using the notion of Kinetic Data Structures (KDS) [4, 10]. Using the Delaunay Triangulation (DT) as the underlying structure we show how to maintain near neighbors of points in moving point sets in two and three dimensions. The same can also be done for constrained environments in two dimensions, using the CDT. Finally, we discuss how to use the CDT for maintaining the relative convex hull of a set of points moving inside a simple polygon.

Section 2 of the paper contains the proof that bounded aspect ratio triangulations are spanners. Section 3 discusses the generalization to environments with obstacles. In Section 4 we show that the CDT is a spanner graph as well. In Section 5 we discuss how

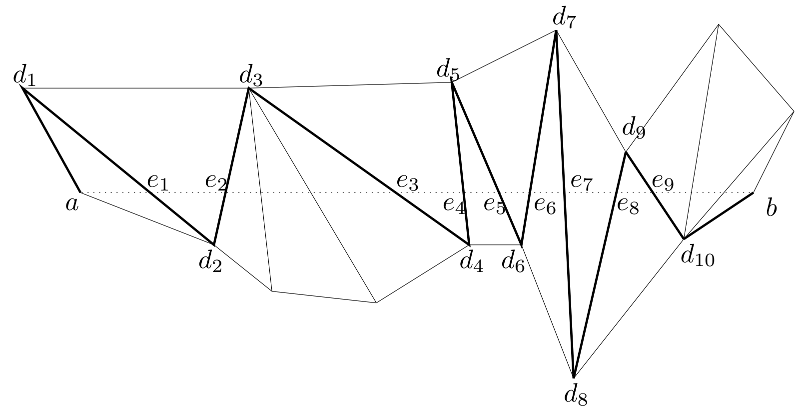

Figure 1: The zone of the segment ab, and the chosen path from a to b in the triangulation.

to kinetize the CDT. In Section 6 we describe the nearest neighbor maintenance algorithm and in Section 7 we deal with the maintenance of the relative convex hull. The final section of the paper is devoted to concluding remarks and open problems.

2 Fat triangulations are spanners

2.1 Triangulations in two dimensions. Let abc be a triangle and let h be its longest side (hypotenuse) and v the corresponding height. The *aspect ratio* of abc is typically defined to be $A(abc) = h/v$ [6], a quantity that is always at least $2\sqrt{3}/3 \geq 1$. There exist other definitions for the aspect ratio of a triangle, which are roughly equivalent to the one we are using in this work. It can easily be shown that if θ is the smallest angle of abc, then

$$\frac{1}{\sin\theta} \leq A(abc) \leq \frac{2}{\sin\theta}\,. \tag{2.1}$$

Let $\mathcal{T}$ be a triangulation. We define the aspect ratio $A(\mathcal{T})$ to be the maximum of the aspect ratios of the triangles in $\mathcal{T}$. If θ_{min} is the minimum angle in $\mathcal{T}$ then the bounds (2.1) hold for $A(\mathcal{T})$ and θ_{min}.

It is plausible to expect that the edges of convex partitions of the plane all of whose faces are 'fat' (by some measure) form a spanner graph of the partition vertices. This is so because for every straight shortcut through a fat face there is a path along the face boundary whose length is larger than the length of the shortcut by at most a constant factor. The main result of this section is to validate a special case of this intuition, by showing that bounded aspect ratio triangulations are spanner graphs of their vertices.

Theorem 2.1. *Let $\mathcal{T}$ be a triangulation of a point set S, such that $A(\mathcal{T}) \leq \alpha$. If a and b are two points in S, then $d_{\mathcal{T}}(a,b) \leq 2\alpha\, d(a,b)$, where $d_{\mathcal{T}}(a,b)$ denotes the length of the shortest path in $\mathcal{T}$ between a and b, and $d(a,b)$ is the Euclidean distance between a and b.*

Proof. Let a and b be two points in S. Without loss of generality we can assume that no point of S lies on the segment ab. If ab is an edge of $\mathcal{T}$ then $d_{\mathcal{T}}(a,b) = d(a,b) \leq 2\alpha d(a,b)$.

If not, then consider the triangles t_0, t_1, …, t_s, t_{s+1} crossed by ab. The line ab separates the points of these triangles (except a and b) into two sets that lie in different half-planes w.r.t. to ab. Moreover, there exists an ordering of the edges of the t_i's crossing ab, induced by the distance of their intersection with ab from a.

We construct a path from a to b zig-zagging above and below the line ab, as follows. From a go to either one of the points of t_0 incident to a. If we are at a point that is incident to b, then go to b. If we are at a point d_i not incident to b, consider all the edges incident to d_i that cross ab. Then d_{i+1} is the endpoint incident to d_i that corresponds to the edge of maximal order with respect to the ordering induced by ab.

Let $a = d_0, d_1, \ldots, d_s, d_{s+1} = b$ be the path defined above (see Fig. 1). This path has the property that, except at the endpoints, two consecutive vertices of the path lie on different sides of the ab. Let e_i be the intersection of $d_{i-1}d_i$ with the line ab, and let us focus on the triangle $e_i d_i e_{i+1}$. Let $\phi_i = \angle d_i e_i e_{i+1}$, $\omega_i = \angle e_i e_{i+1} d_i$ and $\theta_i = \angle e_i d_i e_{i+1}$. Clearly $\theta_{min} \leq \theta_i \leq \pi$.

If $\theta_i > \pi/2$, then

$$d(e_i,d_i) + d(d_i,e_{i+1}) \leq \frac{\pi}{2} d(e_i,e_{i+1}) \leq 2\alpha\, d(e_i,e_{i+1}).$$

If $\theta_i \leq \pi/2$, then using the sine law in the triangle $e_i d_i e_{i+1}$ and the bounds for θ_i we get

$$\begin{aligned} d(e_i,d_i) + d(d_i,e_{i+1}) &= \frac{d(e_i,e_{i+1})}{\sin\theta_{min}}(\sin\omega_i + \sin\phi_i) \\ &\leq 2\alpha\, d(e_i,e_{i+1}). \end{aligned}$$

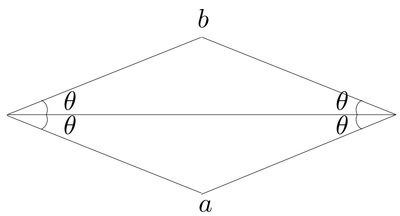

Figure 2: A construction that gives the lower bound for the optimal stretch factor c_{opt}.

Therefore,

$$\begin{aligned} d_{\mathcal{T}}(a,b) &\le \sum_{i=0}^{s} d(d_i, d_{i+1}) \\ &\le 2\alpha \sum_{i=0}^{s} d(e_i, e_{i+1}) = 2\alpha d(a,b)\,. \end{aligned}$$

□

Let c_{opt} be the optimal constant that bounds the ratio between the distances $d_{\mathcal{T}}(a,b)$ and $d(a,b)$. What we have just proved is that $c_{opt} \le 2\alpha$. It is also easy to verify that $c_{opt} \ge \alpha/2$. Consider the triangulation in Fig. 2; the distance between the points a and b on the triangulation is $d(a,b)/\sin\theta$, which is greater than $\alpha d(a,b)/2$.

2.2 Triangulations in three dimensions. In three dimensions the aspect ratio of a tetrahedron is usually defined as the ratio of the radius R of the smallest containing sphere to the radius r of the largest sphere inscribed in the tetrahedron [17]. The aspect ratio $A(\mathcal{T})$ of a three dimensional triangulation $\mathcal{T}$ is defined as the maximum aspect ratio of any tetrahedron in the triangulation. An *interior angle* of the triangulation is an angle between two faces F and G where F and G are a facet and an edge, two facets, or two edges, that have a common intersection and that one face is not a subset of the other (see [17]). If θ_{min} is the minimum interior angle of the triangulation and α a bound on the aspect ratio of the triangulation, then there exist constants c_1 and c_2 such that

$$\frac{c_1}{\theta_{min}} \le \alpha \le \frac{c_2}{\theta_{min}}\,.$$

Proving the spanner property for fat triangulations in three dimensions is more demanding. It requires two steps: first we approximate the straight line path by a path on the faces of the crossed tetrahedra, and then that latter path by another path following only the edges of the tetrahedra. The corresponding theorem is as follows:

THEOREM 2.2. *Let $\mathcal{T}$ be a triangulation of a three dimensional point set S, such that $A(\mathcal{T}) \le \alpha$. Then*

$$\frac{d_{\mathcal{T}}(a,b)}{d(a,b)} \le \beta^2, \quad \beta = \max\{\frac{2\alpha}{c_1}, \frac{\pi}{2}\}\,,$$

where a, b are points in S, $d_{\mathcal{T}}(a,b)$ is the distance of the shortest path in $\mathcal{T}$ between a and b and $d(a,b)$ is the Euclidean distance between a and b.

Proof. We are going to describe a path on the tetrahedrization for which the suggested bound holds.

Consider two points $a, b \in S$ and consider all the triangles that intersect the interior of ab. The intersections of these triangles with ab induce an ordering for the set of triangles. Also, any two consecutive triangles, w.r.t. this ordering, share an edge. If more than two consecutive triangles share a common edge, we discard of all but the first and last triangle. In the remainder of the proof we shall deal with this reduced set of triangles $t_0, t_1, \ldots, t_s, t_{s+1}$.

Let $a = e_0, e_1, \ldots, e_s, e_{s+1} = b$ be the intersections of the triangles with the line ab. We can construct a two-leg polygonal path q_i from e_i to e_{i+1} that lies on the triangles t_i and t_{i+1}, that has the property

$$d_{q_i}(e_i, e_{i+1}) \le \beta\, d(e_i, e_{i+1}). \tag{2.2}$$

Consider an endpoint w of the common edge of t_i and t_{i+1}. Project e_i and e_{i+1} on the common edge with lines parallel to the edges incident to w. Then connect e_i and e_{i+1} to the midpoint f_i of the two projections. The path q_i is the polygonal line $e_i f_i e_{i+1}$. It can be easily seen that the angle $\angle e_i f_i e_{i+1}$ is bounded from below by θ_{min}, which establishes (2.2).

Using the construction above, we have created a polygonal path Q with vertices $a = e_0, f_1, e_1, \ldots, e_s, f_s, e_{s+1} = b$, that separates the endpoints of the edges of the triangles t_i in two disjoint sets (except for a and b), depending on which side of the polygonal path they reside (see Fig. 3). It also induces an ordering for the edges of the t_i's that intersect it. Construct a three-dimensional path from a to b using the edges of the t_i's as follows. From a go to either one of its two incident vertices in t_0. If we are at a point d_i that is incident to b, go to b. If we are at a point d_i not incident to b, consider all edges incident to d_i, that intersect Q. Among those edges choose the one of maximal order w.r.t. the ordering induced by Q; d_{i+1} is the endpoint of this edge incident to d_i. This construction yields a 3D path P with vertices $a = d_0, d_1, \ldots, d_k, d_{k+1} = b$, that goes back and forth across the polygonal line Q. Let $f'_1, f'_2, \ldots, f'_k$ be the subset of the f_i's corresponding to the edges $d_i d_{i+1}$, and let $f'_0 = a, f'_{k+1} = b$. Since the

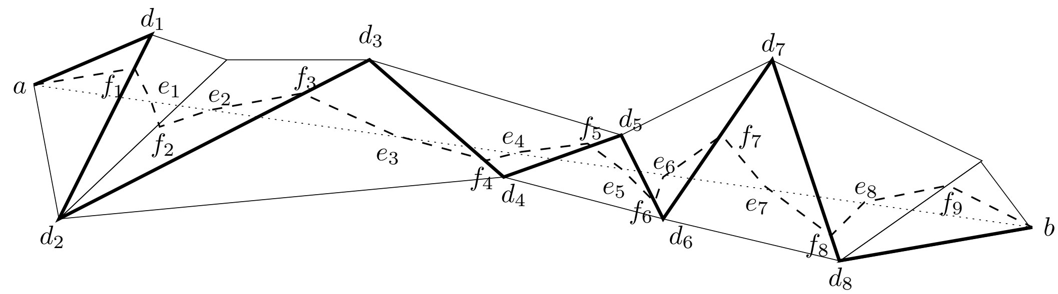

Figure 3: The reduced set of triangles intersecting ab and the paths Q (dashed line) and P (thick solid line).

angles $\angle f_i' d_{i+1} f_{i+1}'$ are bounded from below by θ_{min}, we easily get

$$d(f_i', d_{i+1}) + d(d_{i+1}, f_{i+1}') \leq \beta\, d(f_i', f_{i+1}')$$

which in turn yields :

$$d_P(a,b) = \sum_{i=0}^{k} d(d_i, d_{i+1}) \leq \beta \sum_{i=0}^{k} d(f_i', f_{i+1}').$$

But

$$\sum_{i=0}^{k} d(f_i', f_{i+1}') \leq d_Q(a,b) = \sum_{i=0}^{s} d_{q_i}(e_i, e_{i+1}).$$

Combining the above inequalities with (2.2) we get :

$$\begin{aligned} d_P(a,b) &\leq \beta \sum_{i=0}^{s} d_{q_i}(e_i, e_{i+1}) \\ &\leq \beta^2 \sum_{i=0}^{s} d(e_i, e_{i+1}) = \beta^2 d(a,b)\,, \end{aligned}$$

the result to be shown. □

We believe that similar ideas can be used to prove an analogous result for fat triangulations in any dimension.

3 Environments with obstacles

Let G be a PSLG. The graph G induces a subdivision $\mathcal{S}(G)$ of the plane into regions. Let also $\mathcal{V}(G)$ be the visibility graph associated with G. If v is a vertex of G, then we denote with F_v the set of faces of $\mathcal{S}(G)$ adjacent to v.

We focus on paths that lie entirely within one face of the subdivision $\mathcal{S}(G)$ and do not cross any constraining edges. The following definition captures these requirements.

Definition 3.1. *A path P on the plane between two vertices u and w of G, such that $F_u \cap F_w \neq \emptyset$, is called* legal *if*

1. *the entire path P lies inside the closure of exactly one face of $\mathcal{S}(G)$.*
2. *we can find a path as close as we want to P that shares the same endpoints with P, and the interior of which lies in the interior of the same face as P.*

Definition 3.2. *The* geodesic distance *$d_G(u,w)$, with respect to the graph G, is the length of the shortest legal path between u and w on $\mathcal{V}(G)$, measured in the Euclidean metric.*

We call a triangulation $\mathcal{T}(G)$ *constrained* (with respect to G) if the vertices of $\mathcal{T}(G)$ are those of G and every edge in G is an edge in $\mathcal{T}(G)$. We call a triangulation *conforming* if every vertex in G is in $\mathcal{T}(G)$ and every edge in G is the union of some edges in $\mathcal{T}(G)$. Clearly a constrained triangulation is also conforming.

Theorem 3.1. *Let G be a PSLG and let $\mathcal{T}(G)$ be a conforming triangulation of G such that $A(\mathcal{T}) \leq \alpha$. If u and w are two vertices in G sharing a face of $\mathcal{S}(G)$, then $d_{\mathcal{T}(G)}(u,w) \leq 2\alpha\, d_G(u,w)$.*

Proof. Let $u = v_0, v_1, \ldots v_n = w$ be the sequence of vertices of G that consist of the shortest legal path in $\mathcal{V}(G)$. If $v_{k-1}v_k$ is a portion of a constrained edge, then

$$d_{\mathcal{T}(G)}(v_{k-1}, v_k) = d(v_{k-1}, v_k) \leq 2\alpha\, d(v_{k-1}, v_k)\,.$$

If $v_{k-1}v_k$ is not portion of a constrained edge then consider the path from v_{k-1} to v_k described in the proof of Theorem 2.1. For this path we know that $d_{\mathcal{T}(G)}(v_{k-1}, v_k) \leq 2\alpha d(v_{k-1}, v_k)$. Moreover, it is easy to find a homeomorphism from this path to the segment $v_{k-1}v_k$, which implies that the path lies in the same face as $v_{k-1}v_k$. Therefore,

$$\begin{aligned} d_{\mathcal{T}(G)}(u,w) &\leq \sum_{k=1}^{n} d_{\mathcal{T}(G)}(v_{k-1}, v_k) \\ &\leq 2\alpha \sum_{k=1}^{n} d(v_{k-1}, v_k) = 2\alpha\, d_G(u,w)\,. \end{aligned}$$

□

4 The CDT is a spanner

Dobkin, Friedman and Supowit [9] have shown that the DT is a spanner graph of its vertices. The stretch factor M they could prove was approximately 5.08. Later, Kiel and Gutwin [14] improved the stretch factor to approximately 2.42. It turns out that we can generalize the proof in [9] for the constrained case, and therefore show that the CDT is also a spanner, with respect to the geodesic distance — with the same stretch factor as in [9].

We will prove our result as follows. Let $\mathcal{D}(G)$ be the CDT of G. If u and w are two vertices of G, we find a path P from u to w on $\mathcal{D}(G)$ that is in the same face as the shortest path from to u to w in the plane. We shall then prove that the length of this path is at most M times $d_G(u, w)$, where $M = (1 + \sqrt{5})\,\pi/2$ is the stretch factor in [9]. This gives the desired result, since

$$CDT(u, w) \leq d_P(u, w) \leq M\, d_G(u, w)\,,$$

where $CDT(u, w)$ is the length of the shortest legal path between u and w on $\mathcal{D}(G)$.

Let $u = v_0, v_1, \ldots, v_n = w$ be the sequence of vertices of G that consist of the shortest legal path on $\mathcal{V}(G)$. If an edge $v_{k-1}v_k$ is a constrained edge then obviously it is an edge in the CDT; thus

$$CDT(v_{k-1}, v_k) = d(v_{k-1}, v_k) \leq M\, d(v_{k-1}, v_k).$$

If $v_{k-1}v_k$ is not a constrained edge then v_{k-1} is visible from v_k. It now suffices to find a path on $\mathcal{D}(G)$ from v_{k-1} to v_k for which the inequality holds.

The path P from v_{k-1} to v_k is constructed in the same manner as in [9], but instead of using the Voronoi diagram we use the *bounded Voronoi diagram* [16].

Our proof that $d_P(v_{k-1}, v_k)$ is at most $M d(v_{k-1}, v_k)$ generalizes that in [9] by making sure that P lies in the same face with the shortest legal path between v_{k-1} and v_k on $\mathcal{D}(G)$:

THEOREM 4.1. *Let a, b be two points of G that are mutually visible. Then there exists a CDT path P from a to b of length $d_P(a, b)$, such that $d_P(a, b) \leq M\, d(a, b)$, where $M = (1 + \sqrt{5})\,\pi/2$.*

Proof. We can assume without loss of generality that no point of G lies in the interior of ab. Using $Vorb$(G), the bounded Voronoi diagram, construct the path from a to b as in [9]. Let $a = b_0, b_1, \ldots, b_{m-1}, b_m = b$ be the vertices corresponding to the sequence of bounded Voronoi regions traversed by walking along the line ab. The path $a = b_0, b_1, \ldots, b_{m-1}, b_m = b$ is called the *direct CDT path* from a to b. This path lies on the CDT of G due to Fact 2.3 in [16]. Lemmas 1, 2 and 3 in [9] still hold and moreover so does:

LEMMA 4.1. *If a and b are mutually visible, then the direct CDT path between a and b lies in the same face as the segment ab.*

Proof. Let p_{i+1} be the intersection with the line ab of the common edge of the bounded Voronoi regions of b_i and b_{i+1}. Let t_i be the triangle $b_i p_{i+1} b_{i+1}$. The interior of this triangle is empty of vertices and edges of G. Moreover, the segments $b_i p_i$ and $p_i b_{i+1}$ are also empty of points or edges of G. Now consider the triangle $p_i b_i p_{i+1}$, which we call s_i. The interior of s_i is empty because it is a subset of the bounded Voronoi region of b_i. The union of the interiors of the t_i's and the s_i's, as well as the segments $p_i b_i$ and $b_i p_{i+1}$, cover the interior of the region between the polygonal line $b_0 b_1 \ldots b_m$ and the line ab and in fact that region is empty. Therefore, we can define a homeomorphism from $b_0 b_m$ to the polygonal line $b_0 b_1 \ldots b_m$, which implies that the direct CDT path lies in the same face as $ab \equiv b_0 b_m$. □

The only thing that remains to be established is that the paths $z_k z_{k+1}$ referred to in Lemma 4 in [9] lie in the same face of the line ab. Because of Lemma 4.1 the direct CDT path between z_k and z_{k+1} lies in the same face as the segment $z_k z_{k+1}$. On the other hand the segment $z_k z_{k+1}$ lies in the same face as the segment ab. This is because the area $\{q : \mathbf{y}(q) \geq 0$ and q below $z_k z_{k+1}\}$ is empty, since otherwise $z_k z_{k+1}$ would not be an edge of the convex hull. □

5 Kinetizing the CDT

We start off with a definition.

DEFINITION 5.1. *Let $\mathcal{T}$ be a triangulation and let e be an edge in $\mathcal{T}$. Let T_1, T_2 be the triangles adjacent to e and let u, v be the endpoints of e. Finally let a, b be the vertices of T_1, T_2 that are not u or v. We say that e* passes the InCircle test *if and only if InCircle(a, u, v, b) is false.*

It is shown in [5, Lemma 3] that local InCircle tests establish the global CDT property, i.e. :

LEMMA 5.1. *A triangulation $\mathcal{T}(G)$ of a PSLG G is the CDT if and only if all the non-constrained edges of $\mathcal{T}$ pass the InCircle test.*

Therefore, in order to maintain the CDT we only need to check when an edge fails its InCircle test; when this happens, a single edge flip restores the correctness of the CDT. If we assume that the moving vertices of the CDT do not hit constrained edges, then the only events are such edge flips. When such an event happens we need $\mathcal{O}(1)$ time to update our KDS, i.e., the KDS for the CDT is responsive. However, as in the DT case, the KDS is not local since a moving point may be associated with $\Omega(n)$ certificates. Finally, if the motions of the vertices are algebraic, the total number of combinatorial changes in the CDT, which is also the number of events that we have to process, is $\mathcal{O}(n^2\lambda_s(n))$, where $\lambda_s(n)$ is the maximum length of a Davenport-Schinzel sequence of length n and order s; the order s depends on the complexity of the algebraic motion.

6 Nearest neighbor maintenance

Suppose that we have a set V of moving points in two (three) dimensions and a point $p \in V$, for which we want to know the points in V that are within a certain distance r_p from p. The naive approach is to maintain the distance from p to every other point in V and keep those that are within the prescribed distance. We show how to do better using the Delaunay triangulation of V. Let C_p be the circle (sphere) centered at p with radius r_p. Our crucial observation is that, if we are maintaining the DT of V, the only points that enter or exit C_p are endpoints of edges of the DT crossing C_p exactly once (called crossing edges from now on). Hence, maintaining the near neighbors of p reduces to maintaining the DT and updating the set of crossing edges, whenever a point enters or exits C_p.

This observation can be generalized for constrained two-dimensional environments represented as a PSLG G. A constrained edge e that intersects C_p twice is called a *blocking edge*. The points q that we keep track of are those that are inside C_p and not blocked from p by a blocking edge. It turns out that all such points can be approached from p using a path in the CDT of G that lies entirely inside C_p. Again, as in the unconstrained case, points of interest that enter or exit C_p are endpoints of edges of the CDT crossing C_p. Hence maintaining this point set of interest means maintaining the CDT, as well as maintaining the set of crossing edges.

In this section we will treat the 2D constrained and unconstrained case together, since the DT is a special case of the CDT, but we will treat the 3D case separately. We will precisely define the set of points that we want to maintain and prove that the CDT or DT are good triangulations to use to encapsulate proximity information between the points in our point set. We shall then provide the nearest neighbor maintenance algorithm, which essentially describes how to maintain the set of crossing edges described above.

If we want to maintain the set of near neighbors for a set $S \subseteq V$ of points, we can apply the ideas described above for each one of the points in S separately. A noteworthy feature of our method is that, except for the overhead of maintaining the Delaunay triangulation, it is *motion-sensitive*: all other events processed by the structure reflect actual changes to the neighborhoods of the points of interest. Though the overhead of maintaining the Delaunay triangulation can be significant in the worst case, in practice it has nearly linear efficiency and it can be a useful piece of infrastructure for other applications as well, including clustering, communications, etc.

6.1 Kinetic nearest neighbors in 2D. Let $G(V, E)$ be a PSLG and p be a point in V. The points in V are assumed to be moving. With p we associate a circle C_p, containing p, of radius r_p, which may be time varying. The circle C_p will contain the point p in its interior throughout time.

DEFINITION 6.1. *Let T be a constrained triangulation of G. We call a point q in V* approachable *from p, if q is inside C_p and there exists a path from p to q in T that lies entirely in C_p.*

DEFINITION 6.2. *We say that an edge e of T* properly intersects *C_p, if one endpoint of e lies outside of C_p and the other endpoint of e is approachable from p.*

The fact that the point set that we want to maintain is the set of approachable points w.r.t. the CDT is established by the following theorem.

THEOREM 6.1. (the maximality property) *Let A be the set of points in $V \cap C_p$ that are not blocked from p by a blocking edge. Then A is the set of approachable points of p with respect to the CDT of G.*

Proof. Let $q \in A$ be a point not approachable from p. This implies that there exists an edge e with endpoints u and v, such that u, v are outside of C_p. The edge e splits C_p in two regions and p, q are in different regions. Consider the triangles that contain e, and let q' be the third vertex of the triangle that lies on the same half-space as q. Clearly e cannot be a constrained edge, since then q would not be approachable from p. If q' is not inside C_p, then the circle passing through $q'uv$ must contain either p or q, or some other point in C_p that is visible from either u or v. This contradicts the CDT property for e. If q' is inside C_p then the circle passing through $q'uv$ must contain p or some other vertex in C_p that is visible from u or v. Again we have contradicted the CDT property for e. □

Finally we have the following key theorem, which is the basis of the kinetization process.

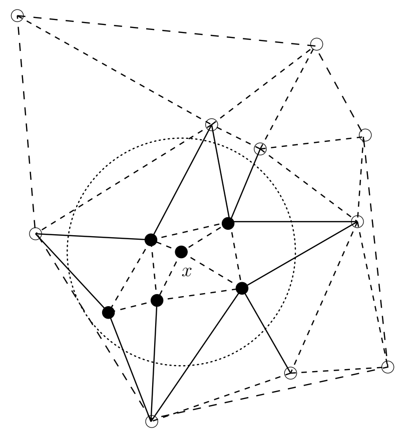

Figure 4: Keeping track of the points that may enter or exit C_x.

THEOREM 6.2. *Let T be the CDT and let $p \in V$ be a point associated with a circle C_p. If a point $q \in V$ enters/exits the circle C_p at some time t_0 and is visible from at least one point inside C_p, then there exists an edge of T between q and a point inside C_p.*

Proof. At time t_0, q is on the boundary of C_p. Let $\{C_r\}$ be the family of circles with center r that pass through q, where r is a point on the segment pq. Consider the circle $C_{r'}$ such that r' is at maximal distance from q, and $C_{r'}$ contains no points of V in its interior that are visible from q. Note that because the set of points of V that are visible from q at t_0 is non-empty by assumption, such a circle $C_{r'}$ always exists. Due to the maximality of r', $C_{r'}$ touches a point $q' \in C_p$ that is visible from q. Clearly the edge qq' is a CDT edge. □

6.2 The kinetic maintenance algorithm. Let A_p be the set of approachable points from p w.r.t. the CDT. Let also E_p be the set of edges of the CDT that properly intersect C_p. As we have already mentioned our goal is to maintain these two sets. In order to do that we have to handle two types of events: edge flips that are required to maintain the CDT, and events that correspond to points entering or exiting C_p.

Whenever an edge flip happens we only have to update the set E_p. If the old edge was in E_p we need to delete it; if the new edge properly intersects C_p we need to add it to E_p.

When a point q enters C_p we have to look at q's neighbors. For those neighbors that are outside C_p we only need to add the corresponding edges to E_p. For those that are inside and in A_p we need to remove the corresponding edges from the edge set E_p. Finally for the neighbors that are inside but not in A_p (this can only occur in the constrained case) we need to add them to the point set A_p and perform the same tests for their neighbors recursively.

When a point q exits C_p the situation is entirely symmetric: for all the neighbors that are outside delete the corresponding edges from E_p. For the neighbors that are inside and remain approachable after the point exits, we need to add the corresponding edges to the set E_p. Finally as far as the remaining neighbors are concerned, we have to delete them from the set A_p of approachable neighbors, delete any edges in E_p that adjacent to them and recursively do the same for their neighbors.

The construction and algorithm described above can be directly generalized, for the unconstrained case, to any L_p metric with $1 < p < \infty$. In particular, we can maintain in exactly the same way near neighbors that are within distance r from a given point q in the L_p metric by maintaining the L_p-metric version of the DT.

A variant of the problem above is where we want to maintain the k nearest neighbors of a point p. Suppose that we have initially computed which are these neighbors. Then the radius r_p of C_p is the distance between p and its k-th nearest neighbor p_k – clearly in this case r_p is time varying. When a point exits C_p, then this point becomes the new p_k, and we have to update how r_p changes with time. If a point enters C_p, then this point becomes the new p_k, and the old one is no longer in the set A_p, and again we need to update r_p. As far as the set E_p is concerned, the only difference now is that we also maintain all the edges adjacent to p_k, no matter whether the neighbors of p_k are inside or outside of C_p.

6.3 Kinetic nearest neighbors in 3D. Our goal in three dimensions is to maintain the set of points that are inside C_p. As in two dimensions, if T is the Delaunay triangulation, this set is the same as the set of A_p of approachable points w.r.t. the DT, and Theorem 6.2 remains true. The proof is slightly more challenging in this case — but we omit the details from this version of the paper.

The three-dimensional Delaunay triangulation is maintained by simply doing some face-edge or edge-face flips [12]. As a result, what we need to do in the 3D case in order to update our nearest neighbors structure is the same as in the two-dimensional unconstrained case and the kinetic maintenance algorithm works as is, the only difference being that flips replace edges with facets (or vice versa), as opposed to edges with edges.

7 The relative convex hull

Relative convex hulls have been of interest in both the computer vision [19] and computational geometry community [11]. In this section we describe how to maintain the relative convex hull for a set of points S moving inside a simple polygon P.

Let R be the relative convex hull of S with respect to P. We will refer to the edges of P as p-edges and to the edges of R as r-edges. Note that a p-edge can be an r-edge, and also note that the graph G with vertices the set $P \cup S$ and edges the union of the set of p-edges and r-edges is a PSLG.

We want to construct and maintain the CDT of G and properly update both G and the triangulation whenever points need to be added or removed from R. There are two kinds of events that we need to handle other than the edge-flip events that we need to process in order to maintain the CDT.

The first kind is the situation when a point in $(S \cup P) \setminus R$ becomes a point of R. Let p be the point in $(S \cup P) \setminus R$ and let q and r be the endpoints of the r-edge that p hits. It can easily be verified that when p becomes collinear with q and r, then the triangle pqr is a triangle of the CDT. What we have to do in this case is to remove qr, add qp and pr to the set of edges in G, and retriangulate the area around p. This can be done by triangulating the quadrangle created by the deletion of qr, and then by simply invoking the standard edge-flip algorithm for producing the CDT given any triangulation, with the appropriate initial edge list [5].

The second kind of event is the symmetric one, when a point in R becomes a point of $(S \cup P) \setminus R$. Let p be the point in R and let q and r be the endpoints of the r-edges incident to p. Such an event can be detected by scheduling CCW tests for all consecutive triplets in R. What we have to do in this case is to delete all the edges connecting p with points inside or outside R, depending on whether $p \in P$ or $p \in S$, respectively, add the edge qr in G, remove the r-edges qp and pr from G (but not from the CDT), triangulate the hole next to p and reconstruct the CDT using the edge-flip algorithm.

8 Conclusions

In this paper we have shown that bounded aspect ratio triangulations in two and three dimensions are spanner graphs. The same result holds true for conforming bounded aspect ratio triangulations in two dimensions, in which case the reference graph is the visibility graph of the input PSLG. We have also proved that the CDT is a spanner graph with respect to the underlying visibility graph. We suspect that there is a common generalization of the spanner property of these triangulations. We conjecture that the spanner property holds whenever a triangulation has the property that every triangle can be circumscribed by a 'fat' shape not containing other triangulation vertices (the witness circles do that for Delaunay).

Based on locality properties of the CDT, we have shown how to kinetize the CDT when the nodes of the input PSLG are moving points. Knowing how to kinetically maintain the DT and the CDT enables us to maintain near neighbors for moving point sets in two and three dimensions, as well as maintain the relative convex hull of a set of points moving inside a polygon. These are useful capabilities in the context of virtual reality systems where object or agent behavior depends on their immediate surroundings or environment.

Although the CDT has certain optimality properties, it does not have a bounded aspect ratio. This raises the question of how to construct bounded aspect ratio conforming triangulations that are easily kinetizable. Finally, since the stretch factors that are presented in this paper are not necessarily optimal, we would like to compute optimal stretch factors for both the CDT and bounded aspect ratio triangulations.

The algorithm that we propose for the relative convex hull seems far from optimal in the sense that it has to process lots of events that have to do with with the CDT maintenance and not the combinatorial structure of the RCH. We would like to investigate alternative ways to approach this problem so that the number of events associated with points inside the RCH depends on their proximity to the hull.

Acknowledgments

The authors wish to thank Olaf Hall-Holt and Li Zhang for useful discussions. Leonidas Guibas and Menelaos Karavelas were supported by NSF grant CCR-9910633 and U.S. Army Research Office MURI grant DAAH04-96-1-007, as well as by a grant from the Intel Corporation.

References

[1] I. Althöfer, G. Das, D. Dobkin, D. Joseph, and J. Soares. On sparse spanners of weighted graphs. *Discrete Comput. Geom.*, 9:81–100, 1993.

[2] B. Awerbuch. Complexity of network synchronization. *J. ACM*, pages 804–823, 1985.

[3] H. J. Bandelt and A. W. M. Dress. Reconstructing the shapoe of a tree from observed dissimilarity data. *Adv. in Appl. Math.*, 7:309–343, 1986.

[4] J. Basch, L. J. Guibas, and J. Hershberger. Data structures for mobile data. In *Proc. 8th ACM-SIAM Symp. on Discrete Algorithms*, pages 747–756, 1997.

[5] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. In D.-Z. Du and F. Hwang,

editors, *Computing in Euclidean Geometry*, pages 47–123. World Scientific, 2nd edition, 1995.

[6] M. Bern, D. Eppstein, and J. Gilbert. Provably good mesh generation. *J. Comput. System Sci.*, 48:384–409, 1994.

[7] L. P. Chew. There is a planar graph almost as good as the complete graph. In *Proc. 2nd ACM Symp. on Comput. Geom.*, pages 169–177, 1986.

[8] G. Das and D. A. Joseph. Which triangulations approximate the complete graph? In *Proc. Internat. Symp. on Optimal Algorithms*, volume 401 of *LNCS*, pages 168–192. Springer-Verlag, Berlin, 1989.

[9] D. P. Dobkin, S. J. Friedman, and K. J. Supowit. Delaunay graphs are almost as good as complete graphs. In *28th Annual Symp. Found. of Comput. Sci.*, pages 20–26, 1987.

[10] L. J. Guibas. Kinetic data structures — a state of the art report. In *Proc. 3rd Work. Algorithmic Found. of Robotics*, 1998. To appear.

[11] J. Hershberger. Optimal parallel algorithms for triangulated simple polygons. In *Proc. 8th Annual Symp. on Comput. Geom.*, pages 33–42, 1992.

[12] B. Joe. Construction of three-dimensional delaunay triangulations using local transformations. *Comput. Aided Geom. Design*, 8:123–142, 1991.

[13] J. M. Keil. Approximating the complete euclidean graph. In *Proc. SWAT*, volume 318 of *LNCS*, pages 208–213. Springer-Verlag, Berlin, 1988.

[14] J. M. Keil and C. A. Gutwin. The delaunay triangulation closely approximates the complete euclidean graph. In *Proc. WADS*, volume 382 of *LNCS*, pages 47–56. Springer-Verlag, Berlin, 1989.

[15] C. Levcopoulos and A. Lingas. There are planar graphs almost as good as the complete graphs and as short as minimum spanning trees. In *Proc. Internat. Symp. on Optimal Algorithms*, volume 401 of *LNCS*, pages 9–13. Springer-Verlag, Berlin, 1989.

[16] A. Lingas. Voronoi diagrams with barriers and the shortest diagonal problem. *Inform. Process. Lett.*, 32:191–198, 1989.

[17] S. A. Mitchell and S. A. Vavasis. Quality mesh generation in three dimensions. In *Proc. 8th ACM Symp. on Comput. Geom.*, pages 212–221, 1992.

[18] P. Peleg and J. Ullman. An optimal synchronizer for the hypercube. *SIAM J. Comput.*, 18:740–747, 1989.

[19] F. Sloboda, B. Zat'ko, and P. Ferianc. Minimum perimeter polygon and its application. In R. Klette and W. G. Kropatsch, editors, *Theoretical Foundations of Computer Vision*, pages 59–70. Akademie-Verlag, Berlin, 1992.