This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2008 proceedings.

Connectivity-based Localization of Large Scale Sensor Networks with Complex Shape

Sol Lederer* Yue Wang* Jie Gao*

* Department of Computer Science, Stony Brook University. {lederer, yuewang, jgao}@cs.sunysb.edu

***Abstract*—We study the problem of localizing a large sensor network having a complex shape, possibly with holes. A major challenge with respect to such networks is to figure out the correct network layout, i.e., avoid global flips where a part of the network folds on top of another. Our algorithm first selects landmarks on network boundaries with sufficient density, then constructs the landmark Voronoi diagram and its dual combinatorial Delaunay complex on these landmarks. The key insight is that the combinatorial Delaunay complex is provably *globally rigid* and has a *unique* realization in the plane. Thus an embedding of the landmarks by simply gluing the Delaunay triangles properly recovers the faithful network layout. With the landmarks nicely localized, the rest of the nodes can easily localize themselves by trilateration to nearby landmark nodes. This leads to a practical and accurate localization algorithm for large networks using only network connectivity. Simulations on various network topologies show surprisingly good results. In comparison, previous connectivity-based localization algorithms such as multi-dimensional scaling and rubberband representation generate globally flipped or distorted localization results.**

I. Introduction

The physical location of sensor nodes is critical for both network operation and data interpretation. In this paper we focus on anchor-free localization in which none of the nodes know their location and the goal is to recover a relative coordinate system up to global rotation and translation. This is motivated by sensor network applications in remote areas or indoor/underwater environments in which GPS or explicitly placed anchor nodes are not available or too costly. Philosophically, anchor-free localization addresses a very fundamental problem: can we recover the network geometry, simply from the network connectivity information? That is, with local knowledge (knowing which nodes are nearby), can we reconstruct the global picture?

As sensor networks scale in size, retrieving the locations of the nodes becomes even more challenging. The difficulty is not only due to the network scale, error accumulation, and the increase to both the communication and computation load, but also due to the fact that large deployments of sensor nodes are more likely to have irregular shapes as obstacles and terrain variations inevitably come in to the picture. Our emphasis in this paper is to localize a large sensor network with a complex shape, by using only the network connectivity.

Incorrect flips vs. graph rigidity. A major challenge in network localization is to figure out the correct global layout and resolve flip ambiguities. To give some intuition, Figure 1 illustrates that with only network connectivity information (or even with measurements of the edge lengths), one is unable to tell the "flip" of triangle $\triangle bcd$ relative to a neighboring triangle $\triangle abc$ locally. Both are valid embeddings. A more

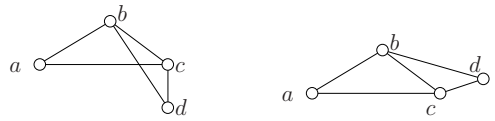


Fig. 1. A connectivity graph with two distinct embeddings having the same set of edge lengths.

severe error can happen as a global flip, shown in Figure 2, that may result from some local flips. The right figure has almost all the nodes correctly localized but has one corner folded over on itself. This is particularly devastating because a node communicating with only its neighbors cannot realize this global error. Indeed, it has been observed that localization algorithms by local optimization may get stuck at one configuration far from the ground truth (see Figure 2 in [1]).

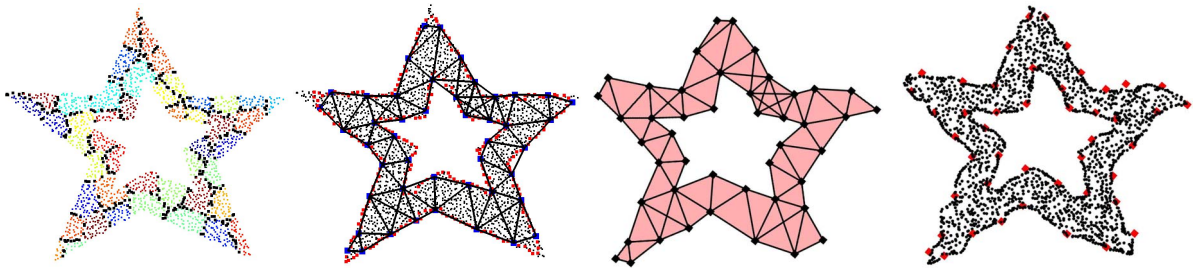
Fig. 2. Left to right: the ground truth; one possible embedding; a more devastating embedding with a global flip.

Resolving flip ambiguities represents a major difficulty in anchor-free localization. When we know the edge lengths, localization is closely related with graph rigidity [2]. A graph is (generically[1]) *rigid* if one cannot continuously deform the graph embedding in the plane without changing the edge lengths. A graph is *globally rigid* if there is a unique realization in the plane. Figure 1 is rigid but not globally rigid. Thus in anchor-free localization, global rigidity is the desirable property. Rigidity without global rigidity may yield flip ambiguities.

A number of localization algorithms deal with the problem of rigidity by exploring the graph structure [1], [3]–[5]. These algorithms either require that the network is dense enough to guarantee the network is a tri-lateration graph (such that iterative trilateration method resolves the ambiguity of flips—an even stronger notion than global rigidity) [1], [3], [4]; or, when the network is sparse, record all possible configurations and prune incompatible ones whenever possible, which, in the worst case, can result in an exponential space requirement [5]. All these algorithms require that neighbors are able to estimate their inter-distances, and they do not work

[1]We typically talk about generic rigidity, that is, the edge lengths are algebraically independent. Intuitively, this means no degeneracy.

978-1-4244-2026-1/08/$25.00



(i) (ii) (iii) (iv)

Fig. 3. Anchor-free localization from network connectivity, on a double star shape. The number of nodes is 2171. The connectivity follows a unit disk graph model with average node degree 10. (i) The Voronoi cells of the landmarks (black nodes are on the Voronoi edges); (ii) The Delaunay edges extracted from the Voronoi cells of the landmarks; (iii) Our embedding result of the extracted Delaunay complex; (iv) Our localization result of the entire network.

with network connectivity alone. Estimating the inter-distances from received signal strength can be quite noisy in a complex environment, and accurate distance estimation requires special ranging hardware.

An approach on anchor-free localization with only network connectivity is to use global optimization such as multi-dimensional scaling (MDS) [6]. MDS takes an inter-distance matrix on n nodes and extracts the node location in $\mathbb{R}^n$. For 2D embedding, the locations are taken as the largest 2D linear projection. Figure 4 (left) shows the result of (MDS) on figure 3(i). Intuitively, MDS tries to stretch the network out in every direction. For a well-connected dense network it gives an O.K. localization result. But it does not have any notion of rigidity and may produce results with global flips. See more examples in Figure 8.

Discovery of global topology. Aside from localization algorithms, recently there is a growing interest in the study of global topology of a sensor field, and its applications in point-to-point routing and information discovery. The focus is to identify high-order topological features (such as holes) from network connectivity [7]–[12] and construct virtual coordinate systems with which one can route around holes [13]–[17]. These virtual coordinates are by no means close to the real node coordinates—they are not meant to be close. Still, an interesting question remains: can the identification of the network geometric features (network boundaries, holes, etc) help in recovering the true node locations?

One such work is to use the rubberband embedding, by Rao *et al.* in [18] and by Funke and Milosavljevic in [16]. The idea is to fix the network outer boundary on a rectangle and then each internal node iteratively takes the center of gravity of its neighbors' locations as its own location. The rubberband relaxation converges to what is called the rubberband representation. With the identification of the network outer boundary, this method does give a layout without incorrect folds, but unfortunately induces large distortion as holes are typically embedded much larger than they are. An example is shown in Figure 4 (right). In the literature [16], [18] the rubberband representation is mainly used in assigning virtual coordinates to the nodes for geographical routing purposes.

Our contribution. Our approach is the convergence of graph rigidity and the high-order topological extraction of the sensor field to derive a globally rigid substructure that recovers the global network layout and provide a basis for a localization algorithm.

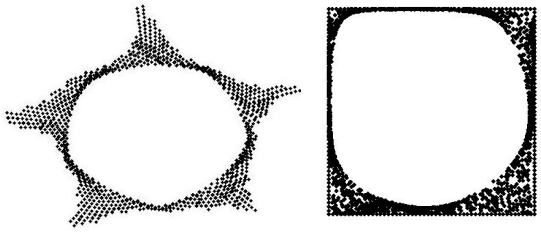


Fig. 4. Embedding of the double star. Left: multi-dimensional scaling; Right: rubberband representation.

We assume the sensor nodes are embedded in a geometric region or on a terrain, possibly with holes. The nodes nearby can directly communicate with each other but far away nodes cannot. We do not use anything beyond the network connectivity information and do not assume neighbors can measure their inter-distances, although such information can be easily incorporated to further improve the localization accuracy.

Briefly, the algorithm can be explained as follows (see Figure 3): Suppose the network boundaries (both the outer boundary and inner hole boundaries) have been discovered (say with any of the algorithms in [7]–[12]). We take samples on the network boundaries and call them *landmarks*. Each node in the network records the closest landmark in terms of network hop distance. The network is then partitioned into *Voronoi cells*, each of which consists of one landmark and all the nodes closest to it (Figure 3(i)). The Delaunay graph (Figure 3(ii)) as the dual of the Voronoi diagram, has two landmarks connected by a Delaunay edge if their corresponding Voronoi cells are adjacent (or share some common nodes).

Now, here is the key insight: given two *Delaunay triangles* sharing a common edge, there is only *one* way to embed them. Thus there is no flip ambiguity! This is because the Delaunay triangles are induced from the underlying Voronoi partitioning so intuitively we can think them as 'solid' triangles, which, when embedded, must keep their interiors disjoint (the case in Figure 1 left cannot happen). In this paper we make this intuition rigorous. We prove in the case of a continuous geometric domain that when the landmarks are sufficiently dense (with respect to the local geometric complexity), the induced Delaunay graph is rigid. Moreover, the Delaunay complex (with high-order simplices such as Delaunay triangles) is *globally rigid*, i.e., there is a unique way to embed these 'solid' Delaunay triangles in the plane.

The identification of the Delaunay triangles and, more importantly, how to embed them relative to each other overcomes a major hurdle toward anchor-free localization. We use an incremental algorithm to glue the triangles one by one. Each Delaunay edge is given a length equal to the minimum hop count between the two landmarks. Since the hop count is only a poor approximation of the Euclidean distance, we use mass-spring relaxation to improve the quality of the embedding and evenly distribute the error (Figure 3 (iii)).

Now with the landmarks localized and the network layout successfully recovered, the landmarks serve as 'anchor' nodes such that each additional node localizes itself by using trilateration with its hop count distances to 3 or more nearby landmarks (in Figure 3 (iv)).

The outline of the paper is as follows. In Section II we prove the rigidity of the Delaunay complex when landmarks are sufficiently dense in the case of a continuous domain (i.e., corresponds to the sensor density approaching infinity). The reason we explain the theory first is to introduce notations and subtle issues due to degeneracies that will show up in the discrete case as well. Readers can also choose to read Section III first, in which we explain the algorithm for the discrete network. Simulation results are presented in Section IV. Due to space constraints, many proofs are omitted from this extended abstract. The proofs are available for reference at [19].

II. Theoretical Foundations

A. Medial axis, local feature size and r-sample

We consider a geometric region $\mathcal{R}$ with obstacles inside. The boundary $\partial\mathcal{R}$ consists of the outer boundary and boundaries of inner holes. For any two points $p, q \in \mathcal{R}$, we denote by $|pq|$ their Euclidean distance and $d(p, q)$ the *geodesic* distance between them inside $\mathcal{R}$, i.e., the length of the shortest path avoiding obstacles. In a discrete network we can use the minimum hop length between two nodes as their distance, whose analog in the continuous case is the geodesic distance. In this paper all the distances are by default measured by the geodesic distances unless specified otherwise. A ball centered at a point p of radius r, denoted by $B_r(p)$, contains all the points within geodesic distance r from p.

Definition 2.1. *The* medial axis *of $\mathcal{R}$ is the closure of the collection of points, with at least two closest points on the boundary $\partial\mathcal{R}$.*

The medial axis of ∂R consists of two components, one part inside $\mathcal{R}$, called the *inner medial axis*, and the other part outside $\mathcal{R}$, called the *outer medial axis*. See Figure 5 (i). In this paper we only care about the inner medial axis.

Now we are ready to explain how to measure the local geometric complexity of $\mathcal{R}$, which will consequently decide the sampling density. An example is shown in Figure 5.

Definition 2.2. *The* inner local feature size *of a point $p \in \partial\mathcal{R}$, denoted as $ILFS(p)$, is the distance from p to the closest point on the inner medial axis.*

Next we prove an important Lemma about the inner local feature size. This Lemma and its proof are motivated by [20] and omitted from this abstract.

Lemma 2.3. *Given a disk B containing at least two points on $\partial\mathcal{R}$, for each connected component of $B \cap \mathcal{R}$, either it contains a point on the inner medial axis, or its intersection with $\partial\mathcal{R}$ is connected.*

Definition 2.4. *An r-sample of the boundary $\partial\mathcal{R}$ is a subset of points S on $\partial\mathcal{R}$ such that for any point $p \in \partial\mathcal{R}$, the ball centered at p with radius $r \cdot ILFS(p)$ has at least one sample point inside.*

Landmark density criterion Our algorithm selects the set of landmarks as an r-sample, with $r < 1$ and selects at least 3 landmarks on each boundary cycle. We will show that these landmarks capture important topological information about the network layout and can be used to reconstruct the network layout.

B. Landmark Voronoi diagram and combinatorial Delaunay graph

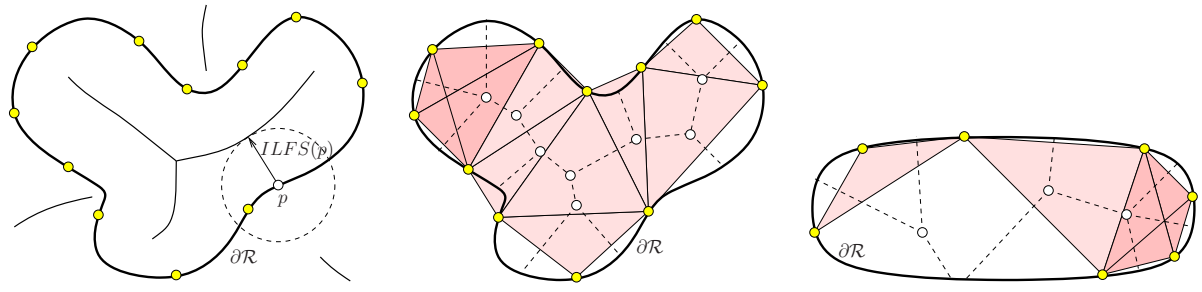
We take some points in $\mathcal{R}$ and denote them as *landmarks* S. Construct the *landmark Voronoi diagram* $V(S)$ as in [13]. Essentially each point in $\mathcal{R}$ identifies the closest landmark in terms of geodesic distance. The *Voronoi cell* of a landmark u, denoted as $V(u)$, includes all the points that have u as a closest landmark. Rigorously,

$$V(u) = \{p \in \mathcal{R} \mid d(p, u) \leq d(p, v), \forall v \in S\}.$$

Each Voronoi cell is a connected region in $\mathcal{R}$. The union of Voronoi cells covers the entire region $\mathcal{R}$. A point is said to be on the *Voronoi edge* if it has equal distance to its two closest landmarks. A point is called a *Voronoi vertex* if its distances to three (or more) closest landmarks are the same. A Voronoi edge ends at either a Voronoi vertex or a point on the region boundary $\partial\mathcal{R}$. The *Voronoi graph* is the collection of points on Voronoi edges. The *combinatorial Delaunay graph* $D(S)$ is defined as a graph on S such that two landmarks are connected by an edge if and only if the corresponding Voronoi cells of these two landmarks share some common points. We state some immediate observations about the Voronoi diagram and the corresponding combinatorial Delaunay graph below.

Observation 2.5. *A point on the Voronoi edge of two landmarks u, v* certifies *that there is a Delaunay edge between u, v in $D(S)$. A Voronoi vertex of three landmarks u, v, w* certifies *that there is a triangle between u, v, w in $D(S)$.*

In the case of a degeneracy, four landmarks or more may become cocircular and thus share one Voronoi vertex. See the left top corner in Figure 5 (ii). We will capture these high-order features by defining the Delaunay complex in the notion of abstract simplicial complex [21]. Formally, a finite system A of finite sets is an *abstract simplicial complex* if $\alpha \in A$ and $\beta \subseteq \alpha$ implies $\beta \in A$. A set α is an (abstract) *simplex* with dimension $\dim \alpha = \operatorname{card} \alpha - 1$, i.e., the number of elements in



(i) (ii) (iii)

Fig. 5. The region $\mathcal{R}$ with boundary shown in dark curves. (i) The medial axis and landmarks selected on the boundaries. Point $p \in \partial\mathcal{R}$ has a landmark within distance $ILFS(p)$. (ii) The Voronoi graph (shown in dashed lines) and the Delaunay graph/complex. (iii) When the set of landmarks is not an r-sample (with $r < 1$), the combinatorial Delaunay graph may be non-rigid.

α minus 1. The way to construct an abstract Delaunay complex is to take the *Čech complex* of the Voronoi cells, defined below.

Definition 2.6. *The* (abstract) Delaunay complex *is the collection of sets*

$$DC(S) = \{\alpha \subseteq S \mid \bigcap_{u\in\alpha} V(u) \neq \emptyset\}.$$

In other words, a set $\alpha \subseteq S$ is a Delaunay simplex if the intersection of the Voronoi cells of landmarks of α is non-empty. The dimension of the Delaunay simplex α is the cardinality of α minus 1.

Thus a landmark vertex is a Delaunay simplex of dimension 0. A Delaunay edge is a simplex of dimension 1. A Delaunay triangle is a simplex of dimension 2 (intuitively, think of the triangle as a 'solid' triangle with its interior filled up). In case of a degeneracy, k landmarks are co-circular and their Voronoi cells have non-empty intersection. This corresponds to a simplex of dimension $k-1$. The rightmost 4 landmarks in Figure 5 (iii) form a dimension-3 simplex (again, intuitively think the simplex as a solid object). We drew the Delaunay complex as shaded regions.

The definition of an abstract simplicial complex is purely combinatorial, i.e., no geometry involved, thus the name of 'abstract' complex. We can talk about an embedding or realization of an abstract simplicial complex in a geometric space as a *simplicial complex* (with geometry). We give the definitions below. The whole point of this paper is to find the geometric realization of the abstract Delaunay complex extracted from a sensor field, to recover its global layout.

A finite set of points is *affinely independent* if no affine space of dimension i contains more than $i+1$ of the points, for any i. A *k-simplex* is the convex hull of a collection of $k+1$ affinely independent points S, denoted as $\sigma = \operatorname{conv} S$. The dimension of σ is $\dim\sigma = k$. Figure 6 shows 0, 1, 2, 3-simplex in $\mathbb{R}^3$. The convex hull of any subset $T \subseteq S$ is

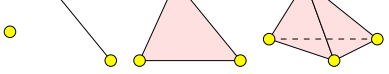


Fig. 6. 0, 1, 2, 3-simplex in $\mathbb{R}^3$.

also a simplex. It is a subset of $\operatorname{conv} S$ and called a *face* of σ. For example, take the convex hull of three points in a 3-simplex, it is a 2-simplex (a triangle). A *simplicial complex* is the collection of faces of a finite number of simplices such that any two of them are either disjoint or meet in a common face. A *geometric realization* of an abstract simplicial complex A is a simplicial complex K together with a bijection φ of the vertex set of A to the vertex set of K, such that $\alpha \in A$ if and only if $\operatorname{conv}\varphi(\alpha) \in K$. Of course the embedded ambient space has to have dimension at least equivalent to the highest dimension of the simplex in A. In our case, when there is degeneracy theoretically we will have to embed in a space with dimension higher than 2. We will discuss how to get around this problem in the next section after the discussion of rigidity. In the rest of the paper, when we say the Delaunay *graph*, we refer to the Delaunay edges and vertices. When we say the Delaunay *complex*, we also include the higher order simplices such as Delaunay triangles etc.

C. Global rigidity of combinatorial Delaunay complex

The property of the combinatorial Delaunay graph clearly depends on the selection of landmarks. The goal of this section is to show that with sufficiently dense landmarks — when there are at least 3 landmarks on each boundary cycle and they form an r-sample of $\partial\mathcal{R}$ with $r < 1$—the Delaunay graph is rigid (no continuous deformation possible if the edges are of fixed lengthes) and the Delaunay complex is globally rigid (it admits a unique realization). An example when the combinatorial Delaunay graph is not rigid due to insufficient sampling is shown in Figure 5 (iii). Now we prepare to prove the rigidity results by first showing that the Voronoi graph (collection of points on Voronoi edges) is connected within $\mathcal{R}$. In this subsection we assume that the landmarks are selected according to the landmark selection criterion mentioned above. The proofs of the following three Lemmas are quite technical and are available in the full paper [19].

Lemma 2.7. *Two Voronoi vertices connected by a Voronoi edge correspond to two Delaunay triangles sharing an edge.*

Lemma 2.8. *For any two adjacent landmarks u, v on the same boundary cycle, there must be a Voronoi vertex inside $\mathcal{R}$ whose closest landmarks include u, v.*

Lemma 2.8 implies that the Delaunay graph has no node with degree 1 – since every node is involved in 2 triangles with its adjacent 2 nodes on the same boundary.

Lemma 2.9. *If there is a continuous curve C that connects two points on the boundary $\partial\mathcal{R}$ such that C does not contain any point on Voronoi edges, then C cuts off a topological 1-disk[2] of $\partial\mathcal{R}$ with no other landmark inside.*

Corollary 2.10. *The Voronoi graph $V(S)$ is connected.*

Now we are able to show that the combinatorial Delaunay graph is rigid. In other words, given a realization of $D(S)$ in the plane, one cannot deform its shape in the plane without changing the lengthes of the edges. To prove this, we use a seminal result about graph rigidity by G. Laman in 1970, known as the *Laman condition*. It states that generically rigid graphs in 2D can be classified by a purely combinatorial condition.

Theorem 2.11 (Laman condition [22]). *A graph G with n vertices is generically rigid in 2 dimensions if and only if it contains a Laman graph G', which has $2n-3$ edges and every subset of k vertices spans at most $2k-3$ edges.*

Theorem 2.12. *The combinatorial Delaunay graph $D(S)$ is rigid, under our sampling condition.*

Proof: In this proof we assume without loss of generality that there is no degeneracy, i.e., four or more landmarks are not co-circular. Indeed degeneracy will only put more edges to the combinatorial Delaunay graph, which only helps with graph rigidity.

From the Voronoi graph $V(S)$, we extract a subgraph V' that contains all Voronoi vertices and the Voronoi edges that connect these Voronoi vertices. Some Voronoi edges end at points on the boundary $\partial\mathcal{R}$ and we ignore those. By Corollary 2.10 this graph V' is connected. Now we find a spanning tree T in V' that connects all Voronoi vertices. Take the corresponding subgraph D' of the combinatorial Delaunay graph $D(S)$ such that an edge exists between two landmarks in D' if and only if there is a point in T that certifies it. D' is a subgraph of $D(S)$. Now we argue that D' is Laman.

First the number of landmarks is n. We argue that the number of edges in D' is $2n-3$. Assuming the number of Voronoi vertices is m, T has $m-1$ Voronoi edges. We start from a leaf node on T and sweep along the edges on T. Each time we add one new vertex that is connected to the piece that we have explored though an edge. During the sweep we count the number of landmarks and the number of Delaunay edges that we introduce. To start, we have T' initialized with one Voronoi vertex, thus we have three landmarks and three Delaunay edges. The new Voronoi vertex x we introduce is adjacent to one and only one vertex in T'—if x is adjacent to two vertices in T', then there is a cycle since T' is connected. This will contradict with the fact that T is a tree. Thus in each additional step we will introduce one Voronoi vertex that is connected to T' through one Voronoi edge. This will introduce one new landmark and two new Delaunay edges. When we finish exploring all Voronoi vertices we have a total of $3+(m-1)=m+2=n$ landmarks, and $3+2(m-1)=2n-3$ Delaunay edges between them. Thus D' has n landmarks and $2n-3$ edges.

With the same argument we can show that any subgraph of D' with k landmarks, denoted by S', has at most $2k-3$ edges. This is because a Delaunay edge is certified by a Voronoi edge. Thus we take the Voronoi edges of T whose corresponding landmarks all fall inside S'. These Voronoi edges span at most a tree between Voronoi vertices involving only landmarks in S', because they are a subset of a tree T. By the same argument there are at most $2k-3$ edges between landmarks in S'. Thus the graph D' is a Laman graph. By the Laman condition the combinatorial Delaunay graph $D(S)$ is rigid. ■

The above theorem shows the rigidity of the combinatorial Delaunay graph, but not the global rigidity yet—there might be several different realizations of the graph in the plane. Indeed for an arbitrary triangulation one may flip one triangle against another adjacent triangle one way or the other to create different embeddings. However, this is no longer possible if we embed the *combinatorial Delaunay complex*, induced from the Voronoi diagram $V(S)$. The intuition is that when the triangles are 'solid' and two triangles cannot share interior points there is only one way to embed the Delaunay complex. Thus the recovered Delaunay complex does reflect the true layout of the sensor field $\mathcal{R}$.

Recall that we want to find an embedding of the abstract Delaunay complex in 2D. That is, find a mapping φ of the vertices in the plane such that any abstract simplex $\sigma \in DC(S)$ is mapped as a simplex $\operatorname{conv}\varphi(\sigma) \in \mathbb{R}^2$. Notice that in the case of degeneracy there are high-order k-simplices, $k \geq 3$, for which a geometric realization requires embedding into a space of dimension k or higher. However, this is not really a problem if we force the dimension to be 2. Indeed, look at all the edges of a k-simplex, $k \geq 3$, they form a complete graph of $k+1 \geq 4$ vertices. Thus it is a 3-connected graph and redundantly rigid (a graph remains rigid upon removal of any single edge). Existing results in rigidity theory [23], [24] show that a graph is globally rigid (uniquely realizable) in 2D under edge lengths constraints if and only if it is tri-connected and is redundantly rigid. Thus all high-order simplices have unique embeddings in the plane (up to global translation and rotation). In this paper we find a geometric realization of the abstract Delaunay complex in the plane, such that for all the simplices with dimension 2 or smaller, they are mapped to simplices in the plane; for simplices of dimension 3 or higher, the induced *graph* is globally rigid and subject to a unique embedding, as explained above.

Now the Delaunay complex is composed of a set of Delaunay triangles (2-simplices) and high-order simplices (and their sub-simplices, of course). We already know that the high-order simplices are embedded in the plane as globally rigid components. The Delaunay 2-simplices/triangles are embedded as geometric complex, i.e., the geometric realization of the abstract Delaunay complex. What is left is to show that given two Delaunay triangles $\triangle uvw$ and $\triangle uvp$ sharing an edge, there is only one way to embed them in the plane as

[2]Intuitively, a topological 1-disk can be continuously deformed into a straight unit length line segment, without cutting or gluing.

required by the definition of simplical complex—that is w and p are on opposite sides of the shared edge uv, as in Figure 7(i). Otherwise, w and p are embedded on the same side of uv.

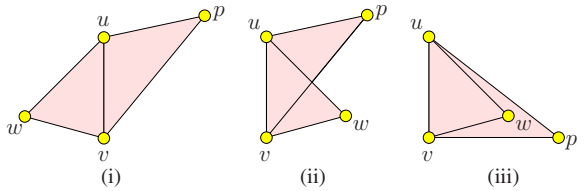


Fig. 7. Two Delaunay triangles $\triangle uvw$ and $\triangle uvp$ sharing an edge. (i) is the only valid valid embedding with the two triangles not sharing any interior points.

Then either w is inside $\triangle uvp$ (as in Figure 7 (iii)), or p is inside $\triangle uvw$, or two edges intersect at a non-vertex point (as in Figure 7 (ii)). This will violate the properties of a simplicial complex that any two simplices are either disjoint or meet at a common face. If w is inside $\triangle uvp$, then the two simplices, a 0-simplex w and a 2-simplex $\triangle uvp$ intersect at a vertex w which is not a face of $\triangle uvp$. In the other case, if two edges intersect at a non-vertex point, this intersection is not a face of either edge.

Now we can conclude with the main theoretical result:

Theorem 2.13. *Under our landmark selection criterion, the combinatorial Delaunay complex $DC(S)$ has a unique embedding in the plane up to a global translation and rotation.*

III. Algorithm Description

We assume a large number of sensor nodes scattered in a geometric region. In general nearby nodes can directly talk to each other and far away nodes can not but the algorithm does not strictly enforce a unit disk graph model. The algorithm basically realizes the landmark selection and embedding suggested in the previous section. Thus we will not re-iterate many things said already and instead focus on the implementation and robustness issues, for the geodesic distance is only poorly approximated by the minimum hop count between two nodes.

A. Select landmarks

We use a distributed boundary detection algorithm that identifies nodes on both outer and inner boundaries and connects them into boundary cycles [12]. With the boundary detected we can identify the medial axis of the sensor field, defined as the set of nodes with at least two closest boundary nodes [14]. The boundary nodes flood inward at roughly the same time [25], [26]. The flooding messages are suppressed by the hop count from the boundary nodes to reduce message complexity. Each node learns its closest boundary node. The nodes at which the flooding frontiers collide are nodes on the inner medial axis.

With the boundary and medial axis identified, we select landmarks from boundary nodes such that for any node p on the boundary, there is a landmark within distance $ILFS(p)$, where $ILFS(p)$ is the inner local feature size of p defined as the hop count distance from p to its closest node on the inner medial axis. In order to find the local feature size of each node on the boundary, nodes on the medial axis flood the network at roughly the same time with proper message suppression. Each boundary node learns its local feature size as the hop count to its closest node on the medial axis.

Now, landmark selection can be performed by a message traversing along the boundary cycles and select landmarks along the way in a greedy fashion to guarantee the sampling criterion. Alternatively, we can let each boundary node p wait for a random period of time and select itself as a landmark. Then p sends a suppression message with TTL as $ILFS(p)$ to adjacent boundary nodes. A boundary node receiving this suppression message will not further select itself as landmarks. Thus landmarks are selected with the required density.

B. Compute Voronoi diagram and combinatorial Delaunay complex

The landmark Voronoi diagram is computed in a distributed way as in [13]. Essentially all the landmarks flood the network simultaneously and each node records the closest landmark(s). Again a node p will not forward the message if it carries a hop count larger than the closest hop count p has seen. So the propagation of messages from a landmark ℓ is confined within ℓ's Voronoi cell. All the nodes with the same closest landmark are naturally classified to be in the same cell of the Voronoi diagram. Nodes with more than one closest landmarks stay on Voronoi edges or vertices. Due to the discreteness of hop count values, we re-define Voronoi vertices.

Definition 3.1. *An* interior node *is a node p with distance to its closest landmark strictly smaller than its distances to all the other landmarks. A* border node *is a node that is not an interior node.*

Figure 3 (i) is an example of the landmark Voronoi diagram with different Voronoi cells colored differently. Border nodes are colored black. We group these border nodes into Voronoi edges and vertices, i.e., the k-witnesses of $(k-1)$-simplices.

Definition 3.2. *A k-witness is a border node which is within 1-hop from interior nodes of k different Voronoi cells. The border nodes that witness the same set of Voronoi cells are grouped into connected* clusters.

One subtle robustness issue, due to the discreteness of sensor nodes, is that there might not be a node that qualifies for the witness defined above (especially for high-order simplices). Thus we propose a *merge* operation: For two clusters A and B that are both k-witnesses, if there exists a node p in cluster A, or exists a node q in cluster B, and all nodes in cluster B are neighbors of p or all nodes in cluster A are neighbors of q, then we merge cluster A and B into one cluster that certifies the union of their corresponding landmarks. The benefit of doing so is to generate high order Delaunay simplices even when there are no corresponding witnesses due to the discrete resolution.

The witnesses certify the existence of Delaunay simplices and by definition can be identified locally. A k-witness node w, after it identifies itself, reports to the corresponding landmarks. Such a report contains the IDs of the landmarks involved in this dimension $k-1$ Delaunay simplex, together with the

distance vector from the witness node w to each of the k landmarks. Remember that nodes in a Voronoi cell store their minimum hop count distances to their home landmark. Thus, the report just follows the natural shortest path pointer to the landmarks involved (so routing is simple). It can happen that multiple witnesses certify the same Delaunay simplex (say, in the case of a Delaunay edge) and they individually report to the same landmark. These report messages are again suppressed during routing. If a node sees a report about a previously received Delaunay simplex, it will not forward it. Naturally the report from the witness with the smallest hop count to its landmarks will arrive the earliest. With these reports, a landmark learns the combinatorial Delaunay simplices it is involved in, and in addition, an approximate hop count to the other landmarks in those simplices through the distance vectors carried in the reports. In particular, a landmark p estimates the hop count distance to landmark q as the minimum of the sum of distances from the witness node to p and q, over all reports received with q involved. This distance estimation can be directly used to embed the Delaunay simplices. Or, if the minimum hop count distances between neighboring landmarks are desired, one can let the messages initiated by the landmarks earlier travel to the adjacent Voronoi cells. Thus each landmark learns the minimum hop count to all neighboring landmarks.

We remark that in the protocol we aggressively use message suppression to reduce the communication cost. With reasonable synchronization most of the flood messages are pruned and the average number of messages transmitted by each node is within a small constant. We also remark that local synchronization (with possible global clock drifts) is sufficient as message suppression occurs mostly among neighboring landmarks.

C. Embed Delaunay complex

Now we are ready to glue the simplices together to embed the landmarks and generate the network layout. Since there is only one way to glue two adjacent simplices (to keep their interiors disjoint, as shown by Theorem 2.13), the embedding is unique. We first embed one simplex S_1 arbitrarily. Then we can embed its neighbor S_2 as follows: Let ℓ_1 and ℓ_2 be the landmarks they share in common. For each landmark ℓ_i in S_2 not yet embedded, we compute the 2 points that are with distance $d(\ell_1, \ell_i)$ from ℓ_1 and $d(\ell_2, \ell_i)$ from ℓ_2, where $d(\cdot, \cdot)$ is the hop-count distance between landmarks, estimated in the previous section. Among the two possible locations we take the one such that the orientation of points $\{\ell_1, \ell_2, \ell_i\}$ is different from the orientation of $\{\ell_1, \ell_2, \ell_r\}$, where ℓ_r is any landmark of S_1, other than ℓ_1 and ℓ_2. Thus ℓ_i and ℓ_r lie on opposite sides of edge $\ell_1\ell_2$.

In some cases one landmark may have two or more neighboring simplices that are already embedded and is thus given multiple coordinate assignments. A natural solution is to take ℓ at the centroid of the different positions. After we have a rough embedding of the entire Delaunay complex, we apply a mass-spring algorithm [27]–[31] to "smooth out" the disfigurements caused by the conflicting node assignments. It is important to recognize however, that mass-spring plays a minor role in our algorithm and its utility is only apparent here because we initially start with topologically correct landmarks positions, i.e., no global flips. Without this initial configuration with good layout a naive mass-spring algorithm can easily gets stuck at local minima, as observed by many [27], [31].

Briefly, the idea of mass-spring embedding is to think of the landmarks as masses and each edge as a spring, whose length is equal to the estimated hop count distance between two landmark nodes. The springs apply forces on the nodes and make them move until the system stabilizes. The objective is to have the measured distances (based on their current locations) between landmarks match as closely as possible the expected distances (indicated by hop count values).

In a distributed environment the embedding of the Delaunay simplices can be done incrementally with message passing. Or, alternatively the combinatorial Delaunay complex can be collected at a central station where the embedding is performed and disseminated to the remaining nodes. As the number of landmarks is only dependent on the geometric complexity of the sensor field, it is much smaller than the total number of nodes. Thus a centralized collection and dissemination of the landmark positions is manageable.

D. Network localization

Since the locations of the landmarks are known, each non-landmark node just runs a tri-lateration algorithm to find its location (e.g., the atomic trilateration in [32]) by using the hop count estimation to 3 or more landmarks.

IV. Simulations

We conducted simulations on various network topologies and node densities to evaluate our algorithm and compare with existing solutions. In the simulations we use a unit disk graph model for the network connectivity. The average node degree varies between 6$\sim$10. The network setup and parameters are shown in the caption for each topology.

Multi-dimensional scaling (MDS). Multidimensional scaling has been used by Shang et al. [6] for sensor network localization with connectivity information only. It is also the only anchor-free localization algorithm so far using connectivity information. For n nodes, the input to MDS is the pairwise distance estimation of size $O(n^2)$. In this case, since only rough hop-count distances are known, MDS has trouble capturing a twist within the graph, making a long narrow graph not differentiable from a spiral-shaped graph. In addition, at the heart of MDS is singular value decomposition (SVD) which has a complexity of $O(n^3)$, making MDS sluggish. In our simulation we tested MDS in two cases, once on all the nodes and once on the landmarks only. They produce similar layout results. MDS on all nodes is very slow. For some experiments with 5000 nodes the matrix operation involved in MDS requires more than 1GB memory.

Simulation results. We observed that the performance of our algorithm is fairly stable for all kinds of shapes, but the performance of MDS depends a lot on the shape of the sensor field. We thus include here a few representative pictures

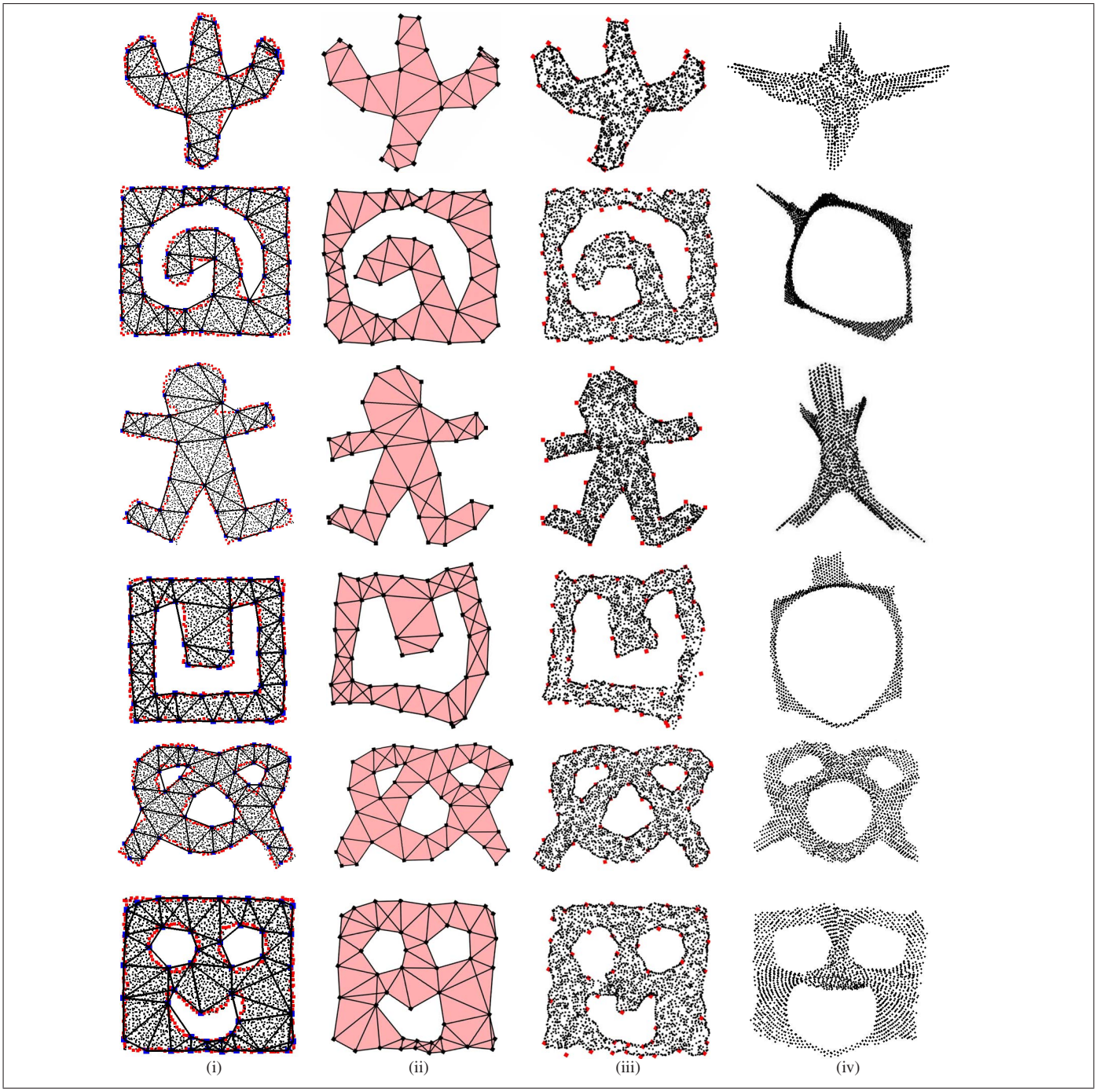


Fig. 8: From left to right, we have: (i) the true sensor locations and extracted combinatorial Delaunay complex; (ii) embedding of the combinatorial Delaunay complex; (iii) localization of all nodes by our algorithm; (iv) the results produced by MDS on all nodes in the network. The connectivity network is generated with unit disk graph model on nodes placed at perturbed grid points. First row: Cactus, 1692 nodes with average degree of 6.9. Second row: Spiral in a box, 2910 nodes with average degree of 9.5. Third row: Ginger man, 2807 nodes with average degree of 10. Fourth row: Square with a concave hole, 2161 nodes with average degree of 10.4. Fifth row: Pretzel, 2993 nodes with average degree of 9.1. Sixth row: Smiley face, 2782 nodes with average degree of 9.5.

in Figure 8. MDS gives reasonable results for some cases (the 5th and 6th example) but performs quite poorly when the real network has curved pieces (like spirals), and may even introduce an incorrect global flip, as in the 2nd and 4th examples. We have computed the average distance error between the true location and our localization result and that of MDS, scaled by the communication range[3]. In all cases we are consistently better. In some cases when MDS does not produce the correct network layout, we are $4 \sim 7$ times better.

Topology	concave	face	man	pretzel	spiral	cactus	star
Our Alg	1.88	0.91	1.94	0.95	1.11	2.39	2.16
MDS	4.42	2.78	3.24	1.45	7.10	2.82	3.24

TABLE I. Average location error, scaled by communication range.

V. Conclusion

The novelty of our localization scheme is to extract high-order topological information to solve the notoriously difficult problem of resolving flip ambiguities. While geometric information of sensor nodes (e.g. node locations) has been recognized as an important character in sensor networks, the global topology of the sensor field can be greatly helpful. In this case, the topology helps in recovering the geometry.

The algorithm to identify the abstract Delaunay complex in a discrete sensor network is a heuristic that uses the intuition from the continuous case. Rigorously we can use the notion the witness complex. This is explored in a later paper [33].

Acknowledgement. This work is supported by NSF CAREER Award CNS-0643687. We thank Alexander Kröller, Joe Mitchell, Rik Sarkar, Xianjin Zhu for various discussions on this problem.

References

[1] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004, pp. 50–61.

[2] J. E. Graver, B. Servatius, and H. Servatius, *Combinatorial Rigidity*. Graduate Studies in Math., AMS, 1993.

[3] T. Eren, D. Goldenberg, W. Whitley, Y. Yang, S. Morse, B. Anderson, and P. Belhumeur, "Rigidity, computation, and randomization of network localization," in *Proceedings of IEEE INFOCOM*, 2004.

[4] D. Goldenberg, A. Krishnamurthy, W. Maness, Y. R. Yang, A. Young, A. S. Morse, A. Savvides, and B. Anderson, "Network localization in partially localizable networks," in *Proceedings of IEEE INFOCOM*, 2005.

[5] D. Goldenberg, P. Bihler, M. Cao, J. Fang, B. D. Anderson, A. S. Morse, and Y. R. Yang, "Localization in sparse networks using sweeps," in *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 2006, pp. 110–121.

[6] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, 2003, pp. 201–212.

[7] S. Funke, "Topological hole detection in wireless sensor networks and its applications," in *DIALM-POMC '05: Proceedings of the 2005 Joint Workshop on Foundations of Mobile Computing*, 2005, pp. 44–53.

[8] S. Funke and C. Klein, "Hole detection or: "how much geometry hides in connectivity?"," in *SCG '06: Proceedings of the twenty-second annual symposium on Computational geometry*, 2006, pp. 377–385.

[9] S. P. Fekete, A. Kröller, D. Pfisterer, S. Fischer, and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *ALGOSENSORS*, ser. LNCS, vol. 3121, 2004, pp. 123–136.

[3]For alignment, we take three arbitrary landmarks and compute a rotation matrix for both results.

[10] S. P. Fekete, M. Kaufmann, A. Kröller, and N. Lehmann, "A new approach for boundary recognition in geometric sensor networks," in *Proceedings 17th Canadian Conference on Computational Geometry*, 2005, pp. 82–85.

[11] A. Kröller, S. P. Fekete, D. Pfisterer, and S. Fischer, "Deterministic boundary recognition and topology extraction for large sensor networks," in *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2006, pp. 1000–1009.

[12] Y. Wang, J. Gao, and J. S. B. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 2006, pp. 122–133.

[13] Q. Fang, J. Gao, L. Guibas, V. de Silva, and L. Zhang, "GLIDER: Gradient landmark-based distributed routing for sensor networks," in *Proc. of the 24th Conference of the IEEE Communication Society (INFOCOM)*, vol. 1, 2005, pp. 339–350.

[14] J. Bruck, J. Gao, and A. Jiang, "MAP: Medial axis based geometric routing in sensor networks," in *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 2005, pp. 88–102.

[15] Q. Fang, J. Gao, and L. J. Guibas, "Landmark-based information storage and retrieval in sensor networks," in *The 25th Conference of the IEEE Communication Society (INFOCOM'06)*, 2006.

[16] S. Funke and N. Milosavljevic, "Network sketching or: "how much geometry hides in connectivity? - part II"," in *18th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

[17] S. Funke and N. Milosavljević, "Guaranteed-delivery geographic routing under uncertain node locations," in *Proceedings of IEEE INFOCOM 2007*, 2007.

[18] A. Rao, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in *Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 96–108.

[19] S. Lederer, Y. Wang, and J. Gao, "Connectivity-based localization of large scale sensor networks with complex shape," http://www.cs.sunysb.edu/∼jgao/paper/layout.pdf.

[20] N. Amenta, M. Bern, and D. Eppstein, "The crust and the β-skeleton: Combinatorial curve reconstruction," *Graphical Models and Image Processing*, vol. 60, pp. 125–135, 1998.

[21] H. Edelsbrunner, *Geometry and Topology for Mesh Generation*. Cambridge Univ. Press, 2001.

[22] G. Laman, "On graphs and rigidity of plane skeletal structures," *J. Engineering Math.*, no. 4, pp. 331–340, 1970.

[23] B. Hendrickson, "Conditions for unique graph realizations," *SIAM J. Comput.*, vol. 21, no. 1, pp. 65–84, 1992.

[24] A. R. Berg and T. Jordán, "A proof of connelly's conjecture on 3-connected generic cycles," *J. Comb. Theory B*, vol. 88, no. 1, pp. 17–37, 2003.

[25] J. Elson, "Time synchronization in wireless sensor networks," Ph.D. dissertation, University of California, Los Angeles, May 2003.

[26] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003, pp. 138–149.

[27] S. G. Kobourov, A. Efrat, D. Forrester, and A. Iyer, "Force-directed approaches to sensor network localization," in *8th Workshop on Algorithm Engineering and Experiments (ALENEX)*, 2006.

[28] A. Howard, M. Mataric, and G. Sukhatme, "Relaxation on a mesh: a formalism for generalized localization," in *In Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pp. 1055–1060.

[29] T. Kamada and S. Kawai, "An algorithm for drawing general undirected graphs," *Inf. Process. Lett.*, vol. 31, no. 1, pp. 7–15, 1989.

[30] T. M. J. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Softw. Pract. Exper.*, vol. 21, no. 11, pp. 1129–1164, 1991.

[31] N. B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller, "Anchor-free distributed localization in sensor networks," MIT LCS, Tech. Rep. TR-892, 2003.

[32] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in *ad-hoc* networks of sensors," in *MobiCom '01: Proceedings of the 7th ACM Annual International Conference on Mobile Computing and Networking*, 2001, pp. 166–179.

[33] J. Gao, L. Guibas, S. Oudot, and Y. Wang, "Geodesic delaunay triangulation and witness complex in the plane," in *Proc. 18th ACM-SIAM Sympos. on Discrete Algorithms*, 2008.