

# Sequential Localization with Inaccurate Measurements

J. Fang

D. Duncan

A. S. Morse

**Abstract**—The sensor network localization problem with distance information is to determine the positions of all sensors in a network given the positions of some sensors and the distances between some pairs of sensors. We present a sequential algorithm for estimating sensor positions when only inaccurate distance measurements are available, and evaluate the performance of the algorithm on simulations of randomly deployed networks of 100 sensors.

## I. INTRODUCTION

In many situations where wireless sensor networks are used, only the positions of some of the sensors are known, and the positions of the remaining sensors must be inferred from the known locations and available inter-sensor distance measurements. More formally, consider  $n$  sensors in the plane labelled 1 through  $n$ , where the positions of some sensors are known, and the measured distances between some pairs of sensors are known. The sensors with known positions are called *anchors*. Since ranging devices are never exact, we consider the following model for the type of distance measurements obtained. For each inter-sensor distance measurement  $\tilde{d}$ , we assume that an *accuracy guarantee*, denoted by  $\varepsilon > 0$ , of  $\tilde{d}$  is given such that the actual inter-sensor distance is within  $\varepsilon$  of the measured distance  $\tilde{d}$ . In general, sensor positions are not uniquely determined by inaccurate distance measurements, and there may not even exist a set of positions in the plane whose induced inter-sensor distances match the measured distances exactly. In [1], [2], it is pointed out that it is more important to obtain position estimates which reflect certain geometric properties of the configuration of the actual sensor positions, rather than simply position estimates whose induced inter-sensor distances are within some desired tolerance of the given distance measurements. In [2], a modified spring based relaxation method is used to obtain sensor estimates, and it is shown via experimental evaluations that the estimated positions reflect the general layout of the actual sensor positions. Our work is most closely related to [1] where the aim was to compute position estimates for subnetworks called “robust quadrilaterals” with correctness guarantees. More specifically, a *robust quadrilateral* is defined as a subnetwork of four sensors such that distance measurements among all four sensors are obtained, and an algorithm is given which assigns position estimates to the sensors in a robust quadrilateral only if the position estimates can

be guaranteed to be free of “flip ambiguities” with high probability [1].

Roughly speaking, position estimates are desired so that the configuration of the estimated positions are approximately congruent to that of the actual positions. We capture this notion using the concept of “correctly oriented” position estimates which we define as follows. For two points  $p$  and  $q$  in  $\mathbb{R}^2$ , let  $l(p, q)$  denote the line segment with endpoints  $p$  and  $q$ . For  $m \geq 4$  sensors labelled 1 through  $m$  and  $i \in \{1, \dots, m\}$ , let  $p_i$  and  $\hat{p}_i$  denote the actual position and estimated position of sensor  $i$  respectively. The estimated positions are said to be *correctly oriented* if for all distinct  $i, j, k, l \in \{1, \dots, m\}$ , the line segments  $l(p_i, p_j)$  and  $l(p_k, p_l)$  intersect if and only if the line segments  $l(\hat{p}_i, \hat{p}_j)$  and  $l(\hat{p}_k, \hat{p}_l)$  intersect. In Section V, we will show that correctly oriented position estimates can be used to deduce important properties of the configuration of actual sensor positions. The key difficulty lies in determining if a set of position estimates are correctly oriented without knowing the corresponding actual sensor positions.

In [3] a sequential localization algorithm for exact distance measurements was proposed in which the sensors of the network are processed one by one in a pre-determined order. That work was extended in [4], [5], [6] to a sequential localization algorithm called Sweeps again for the case of *exact* distance measurements. In this work, we present an algorithm based on Sweeps, which we call “modified Sweeps,” for estimating sensor positions of a network when only inaccurate distance measurements can be obtained. Furthermore, for each position estimate  $p$  computed by modified Sweeps, an error bound  $e(p)$  is also computed such that the maximum distance between the position estimate  $p$  and the actual position is at most  $e(p)$ . The error bounds can be used by the final application to determine which sensor estimates are precise enough to be useful. More importantly, we give a sufficient condition for determining if a set of position estimates with error bounds are correctly oriented *without* knowing the corresponding actual sensor positions.

## II. HIGH LEVEL DESCRIPTION OF MODIFIED SWEEPS

A *candidate regions set* of a sensor is a set consisting of a finite number of regions in the plane with the property that the actual position of the sensor is in one of the regions. A candidate regions set is required to consist of either all polygon regions or all disks, and we call each disk or polygon in a candidate regions set of a sensor a *candidate region* of the sensor. A candidate region of a sensor is *false* if the region does not contain the sensor’s position. Roughly speaking, modified Sweeps first computes

J. Fang (jia.fang@yale.edu), D. Duncan (dominique.duncan@yale.edu) and A. S. Morse (as.morse@yale.edu) are with the Department of Electrical Engineering at Yale University. The work of Fang and Morse was supported in part, by grants from the U.S. Army Research Office and the U.S. National Science Foundation and by a gift from the Xerox Corporation. The work of Duncan was supported by a Yale Graduate Fellowship.

a candidate regions set for each sensor by processing the sensors one by one in some order, and then refines each set by processing the sensors in a different ordering. More specifically, to determine a candidate regions set for each sensor, an ordering of the sensors is first determined so that the anchors precede all other sensors in the ordering, and each non-anchor sensor has at least one “predecessor” in the ordering. A *predecessor* of a sensor is any other sensor preceding it in the ordering such that the measured distance between the two sensors is obtained. Assuming such an ordering exists, the algorithm “sweeps” through the network by processing the sensors sequentially according to the ordering, beginning with the first non-anchor sensor in the ordering. For each non-anchor sensor, a candidate regions set is computed for the sensor using the measured distances between the sensor and its predecessors, and the candidate regions sets, or known positions, of its predecessors. Once a candidate regions set has been computed for each sensor, subsequent “refining” sweeps are performed to remove, if possible, false candidate regions from each candidate regions set. To perform a refining sweep, the sensors are again processed sequentially according to a new ordering. In the following, we will use *singleton* to refer to a set consisting of exactly one element. For each non-anchor sensor  $v$  with a non-singleton candidate regions set, the measured distances between  $v$  and its predecessors, and the minimum and maximum distances between sensor  $v$ ’s candidate regions and the candidate regions of its predecessors in the new ordering are used for identifying false candidate regions of the sensor. When two candidate regions are both disks or both polygons, the minimum and maximum distances between them can be efficiently computed, and it is for this reason that candidate regions sets are constrained to consist of either all disks or all polygons.

A sensor is localized if its candidate regions set consists of a point, i.e. a disk or polygon with zero diameter. In general, exact positions cannot be computed when distance measurements are inaccurate, so the desired outcome for each sensor is that after a finite number of sweeps through the network, the candidate regions set of the sensor contains just one candidate region with a “small” diameter. Whether this will be the case will depend on the configuration of the actual sensor positions and the accuracy guarantees of the measured distances.

We now illustrate two key aspects of modified Sweeps, namely generating and refining candidate regions sets, by using modified Sweeps to estimate sensor positions of a simple five node network. The network consists of five sensors labelled  $a, b, c, u, v$  and positioned at points  $\pi(a)$ ,  $\pi(b)$ ,  $\pi(c)$ ,  $\pi(u)$  and  $\pi(v)$  respectively, so that no three of the sensor positions are collinear. Sensors labelled  $a$ ,  $b$  and  $c$  are anchors, and measurements  $\tilde{d}_{au}$ ,  $\tilde{d}_{bu}$ ,  $\tilde{d}_{av}$ ,  $\tilde{d}_{cv}$  and  $\tilde{d}_{uv}$  are obtained with guaranteed accuracies  $\epsilon_{au}$ ,  $\epsilon_{bu}$ ,  $\epsilon_{av}$ ,  $\epsilon_{cv}$  and  $\epsilon_{uv}$ , respectively. An ordering of the sensors is first determined so that the anchors precede all other sensors, and each non-anchor sensor has at least one predecessor. One such ordering is  $a, b, c, u, v$ . The first “sweep” begins

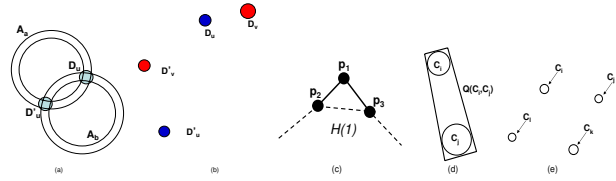


Fig. 1. (a) A five node network (b) Candidate regions of sensors  $u$  and  $v$  (c) The region  $H(1)$  (d) The bounding quadrilateral of two circles (e) Four circles satisfying Condition 1

by determining a candidate regions set for the first non-anchor sensor, which in this case is  $u$ . Let  $\mathbb{A}_a$  denote the ring centered at  $\pi(a)$  with inner radius  $\tilde{d}_{au} - \epsilon_{au}$  and outer radius  $\tilde{d}_{au} + \epsilon_{au}$ , and let  $\mathbb{A}_b$  denote the ring centered at  $\pi(b)$  with inner radius  $\tilde{d}_{bu} - \epsilon_{bu}$  and outer radius  $\tilde{d}_{bu} + \epsilon_{bu}$ . Since  $d_{xu} \in [\tilde{d}_{xu} - \epsilon_{xu}, \tilde{d}_{xu} + \epsilon_{xu}]$ ,  $x \in \{a, b, v\}$ , it follows that  $\pi(u) \in \mathbb{A}_a \cap \mathbb{A}_b$ . Suppose the two rings intersect in two disjoint regions. A disk approximation of the ring intersection is computed which consists of two disks whose union is required to contain the regions of intersection. Let  $\mathbb{D}_u$  and  $\mathbb{D}'_u$  denote the two disks in the approximation. See Figure 1(a). Ideally, the disks in the approximation should be as small as possible; however, we do not require this so. By construction, the set  $\{\mathbb{D}_u, \mathbb{D}'_u\}$  is a candidate regions set for sensor  $u$ . When distance measurements are exact, the two rings will actually be circles and intersect in at most two points, in which case the disk approximation should simply be the set of intersection points. The candidate regions set for sensor  $v$  is computed similarly using the positions of anchors  $a$  and  $c$ , and the measured distances between sensor  $v$  and those anchors. Suppose that the candidate regions set computed for sensor  $v$  also consists of two disjoint disks denoted  $\mathbb{D}_v$  and  $\mathbb{D}'_v$ . In the actual modified Sweeps algorithm, the candidate regions set of sensor  $u$  is also used in the computation of the candidate regions set for sensor  $v$  since  $u$  precedes  $v$  in the chosen ordering. However, we skip this step in an effort to keep this example simple.

Since neither the candidate regions sets of sensors  $u$  nor  $v$  are singletons, a refining sweep will be performed to identify false candidate regions. Without loss of generality, suppose  $\pi(v) \in \mathbb{D}_v$  and  $\pi(u) \in \mathbb{D}_u$ . The idea behind identifying false candidate regions is based on the simple observation that if  $\pi(v) \in \mathbb{D}^*$ , where  $\mathbb{D}^*$  is a candidate region in the candidate regions set of sensor  $v$ , then there must be at least one candidate region  $\mathbb{D}$  in the candidate regions set of sensor  $u$ , namely the candidate region which contains  $\pi(u)$ , such that:  $[d_{\min}(\mathbb{D}^*, \mathbb{D}), d_{\max}(\mathbb{D}^*, \mathbb{D})] \cap [\tilde{d}_{uv} - \epsilon_{uv}, \tilde{d}_{uv} + \epsilon_{uv}] \neq \emptyset$ . Hence, if for some candidate region  $\mathbb{D}^*$  of  $v$ , we have that  $[d_{\min}(\mathbb{D}^*, \mathbb{D}), d_{\max}(\mathbb{D}^*, \mathbb{D})] \cap [\tilde{d}_{uv} - \epsilon_{uv}, \tilde{d}_{uv} + \epsilon_{uv}] = \emptyset$  for all disks  $\mathbb{D}$  in the candidate regions set of  $u$ , then it cannot be the case that  $\pi(v) \in \mathbb{D}^*$ . In this case  $\mathbb{D}^*$  can be removed from the candidate regions set of  $v$  to obtain a smaller candidate regions set.

To refine the candidate regions set computed for sensor  $v$ , we process the sensors in the ordering  $u, v, a, b, c$ . Suppose the disks  $\mathbb{D}_u$ ,  $\mathbb{D}'_u$ ,  $\mathbb{D}_v$ , and  $\mathbb{D}'_v$  are positioned in the plane as

shown in Figure 1(b). If  $\varepsilon_{uv}$  is not too “large”, then  $[d_{\min}(\mathbb{D}'_v, \mathbb{D}_u), d_{\max}(\mathbb{D}'_v, \mathbb{D}_u)]$  and  $[d_{\min}(\mathbb{D}'_v, \mathbb{D}'_u), d_{\max}(\mathbb{D}'_v, \mathbb{D}'_u)]$  are both disjoint from the interval  $[\tilde{d}_{uv} - \varepsilon_{uv}, \tilde{d}_{uv} + \varepsilon_{uv}]$ . This implies  $\pi(v) \notin \mathbb{D}'_v$  and so  $\mathbb{D}'_v$  can be removed from the candidate regions set of  $v$  to obtain the new candidate regions set  $\{\mathbb{D}_v\}$ .

After the second refining sweep,  $\{\mathbb{D}_u, \mathbb{D}'_u\}$  and  $\{\mathbb{D}_v\}$  are the new candidate regions sets of  $u$  and  $v$  respectively. A different ordering is now chosen so as to refine the candidate regions set of sensor  $u$  by a similar procedure in a third refining sweep. Whether false candidate regions of sensors  $u$  and  $v$  can be identified depends on the configuration of the actual sensor positions, and the guaranteed accuracies of the distance measurements. Intuitively, and as have been confirmed by experimental evaluations, the more accurate the distance measurements are, i.e. as  $\varepsilon_{ij} \rightarrow 0$  for each measured distance  $\tilde{d}_{ij}$ , the more likely it is that the candidate regions sets can be refined to be a singleton.

### III. BACKGROUND

A *multi-point*  $p = \{p_1, \dots, p_n\}$  in  $d$ -dimensional space is a set of  $n$  points in  $\mathbb{R}^d$  labelled  $p_1, \dots, p_n$ . A multi-point is *generic* if the elements of the set consisting of the coordinates of its points do not satisfy a non-zero polynomial equation with rational coefficients. Because we are only concerned with networks in the plane, we will henceforth restrict our attention to the case of  $d = 2$ . A graph with vertex set  $\mathcal{V}$  and edge set  $\mathcal{E}$  is denoted by  $(\mathcal{V}, \mathcal{E})$ . A *point formation* in  $\mathbb{R}^2$  of  $n$  points at a multi-point  $p = \{p_1, \dots, p_n\}$  consists of  $p$  and a simple undirected graph  $\mathbb{G}$  with vertex set  $\mathcal{V} = \{1, \dots, n\}$ , and is denoted by  $(\mathbb{G}, p)$ . If  $(i, j)$  is an edge in  $\mathbb{G}$ , then the *length of edge*  $(i, j)$  in the point formation  $(\mathbb{G}, p)$  is the distance between  $p_i$  and  $p_j$ . Two point formations with the same graph have the same edge lengths if the length of each edge in the graph is the same in both point formations. Two point formations with the same graph are congruent if all inter-vertex distances are the same. A point formation  $(\mathbb{G}, p)$  in  $\mathbb{R}^2$  is *globally rigid* in  $\mathbb{R}^2$  if each point formation in the plane with the same graph and edge lengths is congruent to  $(\mathbb{G}, p)$ . For any multi-point  $p = \{p_1, \dots, p_n\}$  in  $\mathbb{R}^2$  and  $\varepsilon > 0$ , let  $\mathcal{B}_p(\varepsilon)$  denote the set of all multi-points  $q = \{q_1, \dots, q_n\}$  in  $\mathbb{R}^2$  where  $\|p_i - q_i\| < \varepsilon$  for all  $i \in \{1, \dots, n\}$ . A graph  $\mathbb{G}$  is said to be *globally rigid* in  $\mathbb{R}^2$  if there exist multi-point  $p$  in  $\mathbb{R}^2$  and  $\varepsilon > 0$  such that  $(\mathbb{G}, q)$  is globally rigid in  $\mathbb{R}^2$  for all  $q \in \mathcal{B}_p(\varepsilon)$ . It is known that if a multi-point  $p$  in  $\mathbb{R}^2$  is generic, then the point formation  $(\mathbb{G}, p)$  is globally rigid in  $\mathbb{R}^2$  if and only if  $\mathbb{G}$  is globally rigid in  $\mathbb{R}^2$  [7], [8].

A network with  $n$  sensors is modelled by a point formation  $(\mathbb{G}, p)$  where each sensor corresponds to exactly one vertex of  $\mathbb{G}$ , and vice versa, with  $(i, j)$  being an edge of  $\mathbb{G}$  if the sensors corresponding to  $i$  and  $j$  are both anchors or if  $i$  and  $j$  are distinct and the distance measurement between the corresponding sensors is obtained, and  $p = \{p_1, \dots, p_n\}$  where  $p_i$  is the position of the sensor corresponding to vertex  $i$ . We say that  $\mathbb{G}$  is the graph of the network, and  $p$  is the multi-point of the network. Since *almost all* multi-points are generic, we will restrict our attention to those networks with generic multi-points. In particular, this implies no two

sensors occupy the same point and no three sensors are collinear in the networks we consider.

A network in which all distance measurements are exact is *localizable* if there corresponds exactly one position to each non-anchor sensor so that the given inter-sensor distances are satisfied. We consider the natural extension of this definition to networks with inaccurate distance measurements where computing exact sensor positions is (in general) impossible. We say that a network of  $n$  sensors positioned at  $\pi(1), \dots, \pi(n)$  respectively is *localizable with precision*  $\rho$  just in case for all points  $p_1, \dots, p_n \in \mathbb{R}^2$  where  $p_i = \pi(i)$  for all anchors  $i$  and  $\|p_i - p_j\| \in (\tilde{d}_{ij} - \varepsilon_{ij}, \tilde{d}_{ij} + \varepsilon_{ij})$  for each distance measurement  $\tilde{d}_{ij}$  with guaranteed accuracy  $\varepsilon_{ij}$ , we have that  $\|p_i - \pi(i)\| \leq \rho$  for all  $i \in \{1, \dots, n\}$ . When distance measurements are exact, a network is localizable if and only if the network has three anchors and the graph of the network is globally rigid in  $\mathbb{R}^2$  [9]. When the distance measurements in a network are inaccurate, it is straightforward to show from the definitions that if the graph of a network is not globally rigid in  $\mathbb{R}^2$ , then there exists  $\rho > 0$  such that the network is not localizable with precision  $\rho$ .

### IV. MODIFIED SWEEPS

In the following, let  $\mathbb{N}$  be a network of  $n$  sensors labelled 1 through  $n$  where each sensor  $i$  is positioned at  $\pi(i)$ . Let  $\mathbb{G} = (\mathcal{V}, \mathcal{E})$  denote the graph of  $\mathbb{N}$  where  $\mathcal{V} = \{1, \dots, n\}$ , and each vertex  $i$  corresponds to sensor  $i$ . For each vertex  $v$ , let  $\mathcal{N}(v)$  denote the set of vertices adjacent to  $v$  in  $\mathbb{G}$ . We assume there are at least three anchors and that  $\mathbb{G}$  is connected. For each  $(i, j) \in \mathcal{E}$  where at least one of  $i$  or  $j$  is a non-anchor sensor, let  $\tilde{d}_{ij}$  denote the distance measurement obtained between sensors  $i$  and  $j$ , and let  $d_{ij}$  denote the actual distance between sensors  $i$  and  $j$ . For each measured distance  $\tilde{d}_{ij}$ , let  $\varepsilon_{ij}$  denote the guaranteed accuracy of the measured distance. This implies that  $d_{ij} \in [\tilde{d}_{ij} - \varepsilon_{ij}, \tilde{d}_{ij} + \varepsilon_{ij}]$  for each distance measurement  $\tilde{d}_{ij}$ . To avoid degenerate cases, we assume that  $\varepsilon_{ij} < \tilde{d}_{ij}$  for all measured distances  $\tilde{d}_{ij}$ .

A *mapping*  $\alpha$  has as its domain a non-empty subset  $\mathcal{U}$  of  $\mathcal{V}$ , and  $\alpha(u)$ ,  $u \in \mathcal{U}$ , are either all disks or all polygons in the plane. Given any mapping  $\alpha$  with domain  $\mathcal{U}$ ,  $\alpha$  is called a *disk (polygon) mapping* if  $\alpha(u)$  is a disk (polygon) for all  $u \in \mathcal{U}$ . For mapping  $\alpha$ , let  $\Delta(\alpha)$  denote the domain of  $\alpha$ . Two mappings  $\alpha$  and  $\beta$  are said to be *consistent*, and we write  $\alpha \sim \beta$ , if there does not exist  $u \in \Delta(\alpha) \cap \Delta(\beta)$  such that  $\alpha$  and  $\beta$  do not map  $u$  to the same region in the plane. For any positive integer  $k$ , consider a collection of  $k$  pairwise consistent mappings  $\alpha_1, \dots, \alpha_k$  which are either all disk mappings or all polygon mappings. Let  $u_k(\alpha_1, \dots, \alpha_k)$  denote the mapping with domain  $\bigcup_{i \in \{1, \dots, k\}} \Delta(\alpha_i)$  whose restriction to  $\Delta(\alpha_i)$  is equal to  $\alpha_i$  for each  $i \in \{1, \dots, k\}$ . For a disk or polygon  $P$  in the plane, let *centroid*( $P$ ) denote the centroid of the convex hull of the points in  $P$ , and let *radius*( $P$ ) denote the maximum distance between *centroid*( $P$ ) and the boundary of  $P$ . For positive reals  $d$  and  $\varepsilon$ , let  $\mathbb{A}(P, d, \varepsilon)$  denote the ring centered at *centroid*( $P$ ) with outer radius  $d + \varepsilon + \text{radius}(P)$ , and inner radius  $d - \varepsilon - \text{radius}(P)$  if  $d - \varepsilon - \text{radius}(P) > 0$ , and zero otherwise.

Consider sensor  $v$  and suppose that the measured distances between  $v$  and sensors  $u_1, \dots, u_m$  are given. For  $i \in \{1, \dots, m\}$ , let  $d_i$  denote the measured distance between sensor  $v$  and sensor  $u_i$ , and let  $\varepsilon_i$  denote the accuracy guarantee of  $d_i$ . Suppose  $\alpha_1, \dots, \alpha_m$  are  $m$  pairwise consistent mappings such that  $u_i \in \Delta(\alpha_i)$  for each  $i \in \{1, \dots, m\}$ , and  $v \notin \bigcup_{i \in \{1, \dots, m\}} \Delta(\alpha_i)$ . The mappings  $\alpha_1, \dots, \alpha_m$  are also required to be either all disk mappings or all polygon mappings. If  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i) \neq \emptyset$ , then any collection of disks or polygon regions in the plane containing  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i)$  can be considered a candidate regions set of sensor  $v$  assuming  $\alpha_i(u_i)$  contains the position of  $u_i$  for each  $i \in \{1, \dots, m\}$ . We aim to define the set  $\mathcal{M}(\alpha_1, \dots, \alpha_m, v, u_1, \dots, u_m)$  with the goal of keeping track of the candidate regions of sensor  $v$  assuming the position of each  $u_i$  is contained in the region  $\alpha_i(u_i)$ . If  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i) = \emptyset$ , then let  $\mathcal{M}(\alpha_1, \dots, \alpha_m, v, u_1, \dots, u_m) = \emptyset$ . Now suppose  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i)$  consists of  $c > 0$  pairwise disjoint regions. Let  $P_{v1}, \dots, P_{vc}$  be  $c$  regions whose union contains  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i)$ . If  $\alpha_1, \dots, \alpha_m$  are all disk (polygon) mappings, then we require that  $P_{v1}, \dots, P_{vc}$  all be disks (polygons). Ideally, each  $P_{vj}$ ,  $j \in \{1, \dots, c\}$ , would be the disk or polygon with the smallest diameter which contains one of the contiguous region in  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i)$ . However, the modified Sweeps algorithm does not require this to be so.<sup>1</sup> If  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i)$  consists of a finite number of points, then  $P_{v1}, \dots, P_{vc}$  should simply be the set of intersection points. Let  $\beta_1, \dots, \beta_c$  denote mappings with domain  $\{v\} \cup \bigcup_{i \in \{1, \dots, m\}} \Delta(\alpha_i)$  defined as follows. For each  $j \in \{1, \dots, c\}$ , let  $\beta_j(v)$  be  $P_{vj}$ , and for  $i \in \{1, \dots, m\}$  and each  $u \in \Delta(\alpha_i)$ , let  $\beta_j(u) = \alpha_i(u)$ . Define  $\mathcal{M}(\alpha_1, \dots, \alpha_m, v, u_1, \dots, u_m)$  as  $\{\beta_1, \dots, \beta_c\}$ . Modified Sweeps will use  $\mathcal{M}$  as the basis for computing candidate regions sets of sensors.

Let  $P_1$  and  $P_2$  be either two disks or two polygons in the plane. Let  $\min(P_1, P_2)$  and  $\max(P_1, P_2)$  denote the minimum and maximum distance, respectively, between  $P_1$  and  $P_2$ . Let  $I(P_1, P_2)$  denote the real line interval with left endpoint  $\min(P_1, P_2)$  and right endpoint  $\max(P_1, P_2)$ .

#### A. Algorithm

For notational convenience, we will in the following only consider polygon mappings, and any reference to mappings will mean polygon mappings. Let  $[v] = v_1, v_2, v_3, \dots, v_n$  be an ordering of  $\mathbb{N}$ 's sensors where  $v_1, v_2, v_3$  are anchors and each  $v_i$  where  $i > 3$  is adjacent to at least one vertex  $v_j$  where  $j < i$ . For  $v_i$ ,  $i \in \{1, 2, 3\}$ , let  $\alpha_i$  be the mapping with domain  $\{v_i\}$  where  $\alpha_i(v_i)$  is the given position of  $v_i$ . For  $i \in \{1, 2, 3\}$ , let  $\mathcal{S}(v_i, 1) = \{\alpha_i\}$ . The sets  $\mathcal{S}(v_i, 1)$ ,  $i > 3$ , are computed iteratively as follows. For  $v_i$ ,  $i > 3$ , let  $u_1, \dots, u_m$  denote the vertices in  $\mathcal{N}(v_i) \cap \{v_1, \dots, v_{i-1}\}$ . Define  $\mathcal{S}(v_i, 1)$  as the union of all  $\mathcal{M}(\alpha_1, \alpha_2, \dots, \alpha_m, v_i, u_1, u_2, \dots, u_m)$ , where

<sup>1</sup>In the instance of the modified Sweeps algorithm we implemented, we used a simple algorithm to determine a set of polygons whose union contains  $\bigcap_{i \in \{1, \dots, m\}} \mathbb{A}(\alpha_i(u_i), d_i, \varepsilon_i)$ , which our experimental evaluations on randomly deployed networks suggest is both computationally efficient and adequate.

$\alpha_j \in \mathcal{S}(u_j, 1)$  for each  $j \in \{1, \dots, m\}$  and  $\alpha_j \sim \alpha_k \forall j, k \in \{1, \dots, m\}$ . Each  $\mathcal{S}(v, 1)$  consists of a finite number of disk mappings, and for each sensor  $v$ , the set  $\{\alpha(v) \mid \alpha \in \mathcal{S}(v, 1)\}$  is a candidate regions set for sensor  $v$ . We call  $\{\alpha(v) \mid \alpha \in \mathcal{S}(v, 1)\}$  the candidate regions set of sensor  $v$  obtained by the first sweep.

Suppose for some  $k \geq 1$  that  $\mathcal{S}(v, k)$ ,  $v \in \mathcal{V}$ , have been computed, and that for each sensor  $v$ , the set  $\{\alpha(v) \mid \alpha \in \mathcal{S}(v, k)\}$  is a candidate regions set for sensor  $v$ . Let  $u_1, \dots, u_n$  be any ordering of the vertices (not identical to the ordering used to compute  $\mathcal{S}(v, k)$ ,  $v \in \mathcal{V}$ ) such that at least one vertex  $v$  where  $\mathcal{S}(v, k)$  is a singleton precede all vertices  $u$  where  $\mathcal{S}(u, k)$  is not a singleton. For each vertex  $u_i$ , let  $\mathcal{P}(u_i)$  denote the set  $\mathcal{N}(u_i) \cap \{u_1, \dots, u_{i-1}\}$ . Let  $s$  denote the number of vertices  $v$  for which  $\mathcal{S}(v, k)$  is a singleton. If  $i \in \{1, \dots, s\}$  or  $\mathcal{P}(u_i) = \emptyset$ , then define  $\mathcal{S}(u_i, k+1) = \mathcal{S}(u_i, k)$ . For  $i \in \{s+1, \dots, n\}$  where  $\mathcal{P}(u_i) \neq \emptyset$ , the idea behind obtaining  $\mathcal{S}(u_i, k+1)$  from  $\mathcal{S}(u_i, k)$  and  $\mathcal{S}(w, k+1)$ ,  $w \in \mathcal{P}(u_i)$ , is as follows. By assumption, the set  $\{\alpha(u_i) \mid \alpha \in \mathcal{S}(u_i, k)\}$  is a candidate regions set of sensor  $u_i$ . Let  $P$  be any candidate region of sensor  $u_i$  from the set. Suppose that for all  $w \in \mathcal{P}(u_i)$ ,  $\{\alpha(w) \mid \alpha \in \mathcal{S}(w, k+1)\}$ , is a candidate regions set of  $w$ . This implies that for all  $w \in \mathcal{P}(u_i)$ , there is a region  $P_w^*$  in  $\{\alpha(w) \mid \alpha \in \mathcal{S}(w, k+1)\}$  which contains the position of  $w$ . Suppose that for some sensor  $w \in \mathcal{P}(u_i)$  that  $I(P, P_w^*)$  is disjoint from  $[\tilde{d}_{wu_i} - \varepsilon_{wu_i}, \tilde{d}_{wu_i} + \varepsilon_{wu_i}]$  for all  $P' \in \{\alpha(w) \mid \alpha \in \mathcal{S}(w, k+1)\}$ . This implies  $P$  cannot contain the position of sensor  $u_i$ , for if  $P$  did contain the position of sensor  $u_i$ , then  $I(P, P_w^*)$  is not disjoint from  $[\tilde{d}_{wu_i} - \varepsilon_{wu_i}, \tilde{d}_{wu_i} + \varepsilon_{wu_i}]$ . In this case we say that  $P$  is an identified false candidate region of sensor  $u_i$ . To obtain the set  $\mathcal{S}(u_i, k+1)$ , we remove all mappings  $\beta$  from  $\mathcal{S}(u_i, k)$  where  $\beta(u_i)$  is an identified false candidate region of sensor  $u_i$ . Since only mappings  $\beta$  where  $\beta(u_i)$  is a false candidate region of  $u_i$  is removed from  $\mathcal{S}(u_i, k)$  to obtain  $\mathcal{S}(u_i, k+1)$ , it follows that  $\{\alpha(u_i) \mid \alpha \in \mathcal{S}(u_i, k+1)\}$  must still be a candidate regions set for sensor  $u_i$ . In the following, for notational convenience, let  $w_1, \dots, w_m$  be the elements of  $\mathcal{P}(u_i)$ , and define  $\mathcal{S}(u_i, k+1)$  as:

$$\begin{aligned} \mathcal{S}(u_i, k+1) = \{ & u_{m+1}(\alpha, \alpha_1, \dots, \alpha_m) \mid \alpha \in \mathcal{S}(u_i, k), \\ & \alpha_j \in \mathcal{S}(w_j, k+1) \quad \forall j \in \{1, \dots, m\}, \\ & \beta \sim \gamma \quad \forall \beta, \gamma \in \{\alpha, \alpha_1, \dots, \alpha_m\}, \\ & I(\alpha(u_i), \alpha_j(w_j)) \cap [\tilde{d}_{u_i w_j} - \varepsilon_{u_i w_j}, \tilde{d}_{u_i w_j} + \varepsilon_{u_i w_j}] \neq \emptyset \\ & \quad \forall j \in \{1, \dots, m\} \} \quad (1) \end{aligned}$$

It can be shown that for each  $v \in \mathcal{V}$ , the set  $\{\alpha(v) \mid \alpha \in \mathcal{S}(v, k+1)\}$  is a candidate regions set for sensor  $v$ , and we call it the candidate regions set of sensor  $v$  obtained by the  $(k+1)$ th sweep. For each sensor  $v$ , the candidate regions set of sensor  $v$  obtained by the  $(k+1)$ th sweep is a subset, not necessarily proper, of the candidate regions set of sensor  $v$  obtained by the  $k$ th sweep.

Suppose that for sensor  $v$  and some  $k \geq 1$  that  $\mathcal{S}(v, k)$  has been computed, and either  $\mathcal{S}(v, k)$  is a singleton, or for all mappings  $\alpha, \beta \in \mathcal{S}(v, k)$ ,  $\alpha$  and  $\beta$  both map  $v$  to the same

region in the plane. Furthermore, suppose there is at least one mapping  $\alpha \in \mathcal{S}(v, k)$  for which there exist  $u, w \in \Delta(\alpha)$  where distance measurements  $\tilde{d}_{uv}$  and  $\tilde{d}_{vw}$  are obtained, i.e.  $u, w \in \mathcal{N}(v)$ , and  $\mathbb{A}(\alpha(u), \tilde{d}_{uv}, \varepsilon_{uv}) \cap \mathbb{A}(\alpha(w), \tilde{d}_{vw}, \varepsilon_{vw})$  consists of two disjoint regions. When the previous hold, the position estimate of sensor  $v$  is taken to be the centroid of  $\alpha(v)$  and the error bound is taken to be  $\text{radius}(\alpha(v))$ , where  $\alpha$  is any mapping in  $\mathcal{S}(v, k)$ .

## V. CORRECTLY ORIENTED POSITION ESTIMATES

Consider in the plane a set of four sensors labelled 1, 2, 3, 4 such that no three sensors are collinear. For  $i \in \{1, 2, 3, 4\}$ , let  $p_i$  and  $\hat{p}_i$  denote the actual and estimated position of sensor  $i$  respectively, and suppose the estimated positions are correctly oriented and no three of the estimated positions are collinear. Let  $H(1)$  denote the region of the plane which does not contain  $p_1$  and is bounded by the line segment  $l(p_2, p_3)$  and the two half-lines both with origin at  $p_1$ , and containing the points  $p_2$  and  $p_3$ , respectively. The regions  $H(2), H(3)$  are defined analogously. See Figure 1(c) for an illustration of  $H(1)$  which is the region of the plane bounded by the dotted lines. The point  $p_4$  lies in  $H(1)$  if and only if the line segments  $l(p_1, p_4)$  and  $l(p_2, p_3)$  intersect. Since the estimated positions are correctly oriented, it follows that  $l(p_1, p_4)$  intersects  $l(p_2, p_3)$  if and only if  $l(\hat{p}_1, \hat{p}_4)$  intersects  $l(\hat{p}_2, \hat{p}_3)$ . Therefore, using the estimated positions, it can be determined if the actual position of sensor 4 lies in  $H(1)$ . By similar reasoning, the estimated positions can be used to determine if  $p_4$  lies in  $H(2)$  and  $H(3)$ . If the actual position of sensor 4 does not lie in any of the  $H_i$ ,  $i \in \{1, 2, 3\}$ , then the actual position of one of the sensors must lie in the convex hull determined by the actual positions of the other three sensors. Hence, a set of correctly oriented sensor position estimates can be used to deduce properties of the configuration of actual sensor positions.

In the following, we give a sufficient condition for determining if a set of position estimates (with error bounds) are correctly oriented *without* knowing the corresponding actual sensor positions. Let  $\mathcal{C}_i$  and  $\mathcal{C}_j$  be two circles in the plane, and let  $\mathcal{Q}(\mathcal{C}_i, \mathcal{C}_j)$  denote the smallest quadrilateral containing both circles  $\mathcal{C}_i$  and  $\mathcal{C}_j$  as shown in Figure 1(d). Consider  $t > 3$  sensors  $u_1, \dots, u_t$  with estimated positions  $\hat{p}_1, \dots, \hat{p}_t$  respectively. For  $i \in \{1, \dots, t\}$ , let  $e_i$  denote the error bound of  $\hat{p}_i$ , and let  $\mathcal{D}_i$  and  $\mathcal{C}_i$  denote respectively the disk and circle in the plane centered at  $\hat{p}_i$  with radius  $e_i$ . Consider the following condition on the geometry of  $\mathcal{C}_1, \dots, \mathcal{C}_t$ :

*Condition 1:* For all distinct  $i, j, k, l \in \{1, \dots, t\}$ , the quadrilaterals  $\mathcal{Q}(\mathcal{C}_i, \mathcal{C}_j)$  and  $\mathcal{Q}(\mathcal{C}_k, \mathcal{C}_l)$  are either disjoint, or  $\mathcal{Q}(\mathcal{C}_i, \mathcal{C}_j)$  and  $\mathcal{Q}(\mathcal{C}_k, \mathcal{C}_l)$  intersect in a quadrilateral  $\mathcal{Q}$  such that  $\mathcal{Q}$  is disjoint from each of the circles  $\mathcal{C}_i, \mathcal{C}_j, \mathcal{C}_k, \mathcal{C}_l$ , and if  $e$  and  $e'$  are opposite edges of  $\mathcal{Q}$ , then both edges are contained in  $\mathcal{Q}(\mathcal{C}_i, \mathcal{C}_j)$  or  $\mathcal{Q}(\mathcal{C}_k, \mathcal{C}_l)$ .

Figure 1(e) shows the relative positions of four circles in the plane which satisfy Condition 1. Suppose Condition 1 holds for  $\mathcal{C}_i$ ,  $i \in \{1, \dots, t\}$ . Then for all distinct  $i, j, k, l \in \{1, \dots, t\}$ , and  $q_i \in \mathcal{D}_i$ ,  $q_j \in \mathcal{D}_j$ ,  $q_k \in \mathcal{D}_k$ ,  $q_l \in \mathcal{D}_l$ , we have

that  $l(q_i, q_j)$  intersects  $l(q_k, q_l)$  if and only if  $l(\hat{p}_i, \hat{p}_j)$  intersects  $l(\hat{p}_k, \hat{p}_l)$ . The previous implies  $\hat{p}_1, \dots, \hat{p}_t$  are correctly oriented since the actual position of sensor  $u_i$  is contained in  $\mathcal{D}_i$  for each  $i \in \{1, \dots, t\}$ .

## VI. EXPERIMENTAL EVALUATIONS

In our evaluations, we used Matlab to generate a random network of 100 nodes using three input parameters  $R$ ,  $m$  and  $p$ , where  $R$  is the sensing range,  $m$  is the number of anchors, and  $p$  is the noise factor. Sensor positions are randomly generated from the distribution that is uniformly distributed on the  $1 \times 1$  two dimensional space, and  $m$  of those sensors are randomly chosen to be anchors. For each pair of sensors within sensing range  $R$ , a noisy distance measurement is generated using the input parameter noise factor  $p$ . More specifically,  $p$  is specified to be between zero and one, and for each pair of sensors within sensing range, a distance measurement  $\tilde{d}$  is generated such that the actual inter-sensor distance is within  $p \cdot \tilde{d}$  of the distance measurement. In other words, the guaranteed accuracy of each generated distance measurement is  $p \cdot 100$  percent of the distance measurement. This corresponds to the notion that the distance measurement between two sensors become less accurate as the inter-sensor distance increases. For ease of implementation, we used convex polygons to approximate ring intersection regions via a simple algorithm using tangent lines and convex hulls.

We evaluated modified Sweeps on networks whose graphs have “augmented bilateration” orderings. A graph is said to have a *bilateration ordering* if its vertices can be ordered so that  $v_1, v_2, v_3$  induce a complete subgraph, and each  $v_i$ ,  $i > 3$ , is adjacent to at least two vertices  $v_j$  where  $j < i$ . A graph has an *augmented bilateration ordering* if it has a bilateration ordering  $v_1, \dots, v_n$  where  $v_1, v_2, v_3$  are anchors, and if  $v_i$ ,  $i > 3$ , is adjacent to only two vertices  $v_j$  where  $j < i$ , then  $v_i$  must be also adjacent to  $v_{i+1}$  and  $v_{i+1}$  is adjacent to two vertices  $v_j$  where  $j < i$ , at least one of which is not adjacent to  $v_i$ . A graph has a *trilateration ordering* if its vertices can be ordered so that  $v_1, v_2, v_3$  induce a complete subgraph, and each  $v_i$ ,  $i > 3$ , is adjacent to at least *three* vertices  $v_j$  where  $j < i$ . If the graph of a network in the plane with a generic multi-point and three anchors has either an augmented bilateration ordering or a trilateration ordering, then the network is localizable. A graph’s augmented bilateration orderings are said to be *untrilaterable* if no such ordering is also a trilateration ordering.<sup>2</sup> In our evaluations, we consider both networks whose graphs have augmented bilateration orderings, and untrilaterable augmented bilateration orderings. The ordering chosen for the first sweep is an augmented bilateration ordering of the network. For evaluation purposes, we implemented a slightly altered version of modified Sweeps which we have found to be more computationally efficient. More specifically, if  $\mathcal{S}(v, 1)$  is a singleton for some  $v \in \mathcal{V}$ , and  $\alpha$  is the mapping in  $\mathcal{S}(v, 1)$ , then before proceeding with the first sweep, each

<sup>2</sup>It can be shown by an edge counting argument that a graph can have an augmented bilateration ordering without possessing any trilateration orderings.

$\mathcal{S}(u, 1)$ ,  $u \in \mathcal{P}(v, 1)$ <sup>3</sup>, is refined by removing all mappings  $\beta$  from  $\mathcal{S}(u, 1)$  where  $\beta(u) \neq \alpha(u)$ . Each  $\{\gamma(u) \mid \gamma \in \mathcal{S}(u, 1)\}$ ,  $u \in \mathcal{P}(v, 1)$ , remains a candidate region set for sensor  $u$  after the refinement step. Using this slightly altered version of modified Sweeps, we sweep through the network only once.

We now discuss two scenarios in detail. We first consider networks of 100 sensors whose graphs have an augmented bilateration ordering for which the input parameters are:  $R = 0.2$ ,  $m = 15$ , and  $p = 0.08$ . We averaged the results of modified Sweeps over 100 randomly deployed instances of such networks with the aforementioned parameters. We found that on average 47 of the 85 non-anchor sensors are assigned a position estimate, and the average error bound was 0.0303, which is less than  $\frac{1}{6}$  of the sensing range. Hence, on average, the actual position of a sensor can be guaranteed to be within 0.0303 of its estimated position. The average of the distance between each position estimate and the actual sensor position did not exceed 0.02, and was in general far less than the average error bound. As expected, when the noise factor is decreased to 0.05, more sensors on average are assigned position estimates, and the corresponding error bounds are also lower. We next considered networks generated by the same input parameters as the previous scenario. However, after the network is generated, the sensors which are adjacent to more than two anchors are identified, and for each such sensor, distance measurements are generated only between the sensor and two of the anchors the sensor is adjacent to. Any augmented bilateration ordering of the graph of the resulting network can thus be guaranteed to be unilaterable. We average the results over 100 instances of randomly deployed networks with the aforementioned properties. Less sensors in total are assigned position estimates by Sweeps as compared to the previous scenario. More specifically, 40 of the sensors, as opposed to the previous 47, are assigned position estimates. The average error bound of the position estimates was 0.0366, which is just slightly higher than the previous scenario.

Generally speaking, we found that as the sensing range or the number of anchors of the network increased, the number of sensors for which a position estimate was computed increased. To illustrate this trend, we considered networks of 100 sensors whose graphs have an augmented bilateration ordering for which the input parameters are:  $R = 0.2$ , and  $p = 0.05$ . By varying the number of anchors, we see what effect this has on the number of sensors for which a position estimate was computed and on the average error bound of the position estimates. We found the most meaningful results to occur when 10-25 anchors were used. There is a significant amount of variation in the case when there are less than 10 anchors, due to the fact that the sample size is too small, and 25 anchors resulted in almost perfect localization, so it is unnecessary to consider cases of more than 25 anchors. Twenty simulations were run for each of the following cases:

<sup>3</sup> $\mathcal{P}(v, 1)$  denotes the set of sensors adjacent to  $v$  and preceding  $v$  in the ordering of the first sweep.

10 anchors, 15 anchors, 20 anchors and 25 anchors, and averaged in the Monte Carlo method. In Figure 2 on the left, we see an increase in the number of sensors for which a position estimate is computed as the number of anchors increases from 10 to 25. In Figure 2 on the right, we see that the mean error decreases from 0.0054 to 0.004 as the number of anchors increases from 10 to 25, which is what we expect.

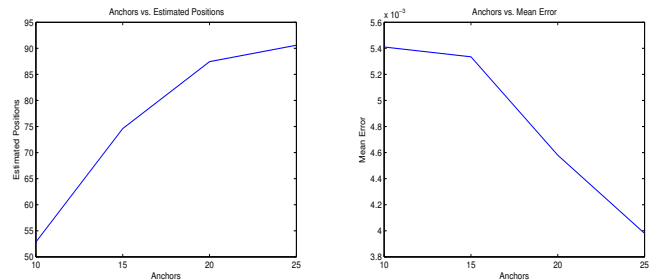


Fig. 2. Number of anchors versus the number of estimated positions and mean error

## VII. CONCLUSION

In this paper we presented a sequential algorithm called modified Sweeps for estimating sensor positions, together with error bounds, of a network when only inaccurate distance measurements and some anchor positions are available. We defined the concept of correctly oriented position estimates, and give a sufficient condition on the estimated positions and the corresponding error bounds in order to guarantee that the estimated positions are correctly oriented. For future work, we will evaluate the algorithm using different techniques for determining orderings. We also aim to extend the proposed algorithm to a decentralized setting and carry out more extensive experimental evaluations using actual sensor data.

## REFERENCES

- [1] David Moore, John Leonard, Daniela Rus, and Seth Teller. Robust distributed network localization with noisy range measurements. In *Proc. 2nd ACM SenSys*, pages 50–61, Baltimore, MD, November 2004.
- [2] N. B. Priyantha, H. B., E. Demaine, and S. Teller. Poster abstract: anchor-free distributed localization in sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 340–341, New York, NY, USA, 2003. ACM.
- [3] B. D. O. Anderson, P. N. Belhumeur, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, and Y. R. Yang. Graphical properties of easily localizable networks. *Wireless Networks*, 2007.
- [4] J. Fang, M. Cao, A.S. Morse, and B.D.O. Anderson. Localization of sensor networks using sweeps. In *Proceedings of CDC*, pages 4645–4650, San Diego, CA, 2006.
- [5] D.K. Goldenberg, P. Bihler, M. Cao, J. Fang, B.D.O. Anderson, A.S. Morse, and Y.R. Yang. Localization in sparse networks using sweeps. pages 110–121, 2006. *Proceedings of Mobicom*.
- [6] J. Fang, M. Cao, A.S. Morse, and B.D.O. Anderson. Sequential localization of sensor networks. *SIAM J. on Control and Optimization*, 2008. To appear.
- [7] B. Hendrickson. Conditions for unique graph realizations. *SIAM J. Comput.*, 21(1):65–84, 1992.
- [8] B. Jackson and T. Jordán. Connected rigidity matroids and unique realizations of graphs. *J. Comb. Theory Ser. B*, 94(1):1–29, 2005.
- [9] T. Eren, D. Goldenberg, W. Whiteley, Y.R. Yang, A.S. Morse, B.D.O. Anderson, and P.N. Belhumeur. Rigidity, computation, and randomization in network localization. In *Proc. of IEEE INFOCOM*, 2004.