Chapter 13 in Sensor Networks and Configuration: Fundamentals, Techniques, Platforms and Experiments, N.P. Mahalik (ed.), Springer-Verlag, pp. 281-315, 2006

Localization

Guoqiang Mao, Barış Fidan and Brian D. O. Anderson

1 INTRODUCTION

Most wireless sensor network applications require knowing or measuring locations of thousands of sensors accurately. In environmental sensing applications such as bush fire surveillance, water quality monitoring and precision agriculture, for example, sensing data without knowing the sensor location is meaningless [1]. In addition, location estimation may enable applications such as inventory management, intrusion detection, road traffic monitoring, health monitoring, etc.

Sensor network localization refers to the process of estimating the locations of sensors using measurements between neighbouring sensors such as distance measurements and bearing measurements. In sensor network localization, it is typically assumed that a small portion of sensors, called anchors, have *a priori* information about their coordinates. These anchor nodes serve to fix the location of the sensor network in the global coordinate system. In applications which do not require a global coordinate system (e.g., monitoring in an office building or home environment), these anchor nodes define the reference coordinate system in which all other sensors are referred to. The coordinates of the anchor nodes may be obtained by using a global positioning system (GPS) or by installing the anchor nodes at fixed points with known coordinates. However due to constraints on cost and size of sensors, energy, implementation environment (e.g., GPS receivers cannot detect the satellites' transmission indoors) or the deployment of sensors (e.g., sensor nodes may be randomly deployed in the region), most sensors do not have *a priori* coordinate information. These sensor nodes without *a priori* coordinate information are referred to as the non-anchor nodes and their coordinates are to be estimated by the sensor network localization algorithm. In this chapter, we shall provide an overview of sensor network localization techniques as well as an introduction to the fundamental theory underpinning the sensor network localization. While many techniques covered in this chapter can be applied in both 2-dimensions ($\Re^2$) and 3-dimensions ($\Re^3$), we choose to focus on 2-dimensional localization problems. The rest of the chapter is organized as follows. In Section 2, we shall provide an overview of measurement techniques and the corresponding localization algorithms. In Section 3 we shall focus on connectivity-based localization algorithms. In Section 4 we shall focus on distance-based localization techniques. Section 5 introduces the fundamental theory of the various distance-based localization techniques and Section 6 summarizes current research problems in distance-based localization. Finally a summary is provided in Section 7.

2 MEASUREMENT TECHNIQUES

Most wireless sensor network localization algorithms rely on measurements between neighbouring sensors for location estimation. Measurement techniques in sensor network localization can be broadly classified into three categories: received signal strength (RSS) measurements, angle of arrival (AOA) measurements and propagation time based measurements. The propagation time based measurements can be further divided into three subclasses: one-way propagation time measurements, roundtrip propagation time measurements and time-difference-of-arrival (TDOA) measurements.

2.1 Received signal strength measurements

Received signal strength indicator (RSSI) has become a standard feature in most wireless devices and the RSS based localization techniques have attracted considerable attention in the literature for obvious reasons. The RSS based localization techniques eliminate the need for additional hardware, and exhibit favourable properties with respect to power consumption, size and cost. As such, the research community

It should be noted that P_{ij} and $P_0(d_0)$ in (1.4) are measured in watts instead of dB milliwatts.

Using (1.3) and (1.4), a simple calculation will show that the expected value of $\hat{d}_{ij}$ is related to the true distance d_{ij} by:

$$E(\hat{d}_{ij}) = d_{ij} e^{\frac{\sigma^2}{2\eta^2 n_p^2}} \tag{1.5}$$

where $\eta = \frac{10}{\ln(10)}$. That is, the maximum likelihood estimate in (1.4) is a biased estimate of the true distance and the bias can be removed when σ and n_p are known.

Given the distances between neighbouring sensors, algorithms can be developed to estimate the coordinates of the non-anchor sensor nodes. A detailed discussion on these algorithms can be found in Section 4.

2.1.2 RSS profiling based localization techniques

The RSS profiling based localization techniques have been mainly used in location estimation in wireless local area networks (WLANs). In the RSS profiling based localization techniques, a large number of sample points are distributed throughout the coverage area of the sensor network. One or more signal strength measurements are taken from all visible anchor nodes (e.g., access points in WLAN, sniffing devices [7]) at each sample point. Based on the collected information, an RSS model is established where each sample point is mapped to either a signal strength vector or a signal strength probability distribution. The RSS model constructed is essentially a map of the signal strength behaviour in the coverage area. The RSS model generated is unique with respect to the anchor locations and the sensor network environment. The RSS model is then stored in a central location.

A non-anchor node that is not aware of its location collects one or more signal strength measurements from all the visible anchor nodes and creates its own RSS fingerprint which is sent to the central station. The central station matches the presented signal strength vector to the RSS model using either nearest neighbour based methods [5] or probabilistic techniques, from which an estimate of the non-anchor node's location can be obtained. The estimated location is then sent to the non-anchor node by the central station. Alternatively, a non-anchor node may request the RSS model from the central station and perform the location estimation by itself.

The accuracy of the RSS profiling based localization techniques depends on the particular technique used to build the model and the algorithm used to match the measured signal strength to the model. Generally speaking, the RSS profiling based localization techniques produce relatively small location estimation errors in comparison with the distance estimation based techniques [5]. Elnahrawy *et al.* proposed several area-based localization algorithms based on the RSS profiling [3]. Different from the earlier point-based algorithms which estimate the exact location of the non-anchor node, the area based algorithms estimate the possible area that might contain the non-anchor node. The performance of the area based algorithms is measured by two parameters: accuracy and precision, where accuracy is the likelihood the object is within the area and precision is the size of the area. Three techniques were proposed for the area based algorithms, i.e., single point matching, area based probability and Bayesian networks. By comparing the performance of these area based algorithms with the point based algorithm in [5], they observed that the performance of these algorithms is quite similar and there is a fundamental limit in the RSS profiling based localization algorithms. Elnahrawy *et al.* also found a general rule of thumb is that using 802.11 technology, with dense sampling and a good algorithm, one can expect a median localization error of about 3 m and a 97^{th} percentile error of about 9 m. With relatively sparse sampling, every 6 m, or 37 $m^2/sample$, one can still

Typically four or more antennas are used, with coarse tuning being done by simply measuring which antenna has the strongest signal, followed by fine tuning by comparing the amplitude responses.

The second category of measurement techniques is based on the principle that a transmitter's bearing with regards to the receiver can be obtained from measurements of the phase differences in the arrival of a wave front. The measurements typically require a large receiver antenna (in comparison with the wavelength of the transmitter signal) or an antenna array. Fig. x.2 shows an antenna array of N antenna elements. The adjacent antenna elements are separated by a uniform distance d. For a transmitter far away from the antenna array, the distance between the transmitter and the i^{th} antenna element is approximately given by $R_i \simeq R_0 - id\cos\theta$, where R_0 is the distance between the transmitter and the 0^{th} antenna element and θ is the bearing of the transmitter with regards to the antenna array. Therefore the transmitter signal received by adjacent antenna elements will have a phase difference of $2\pi \times \frac{d\cos\theta}{\lambda}$. By measuring the phase difference, the bearing of the transmitter can be obtained.

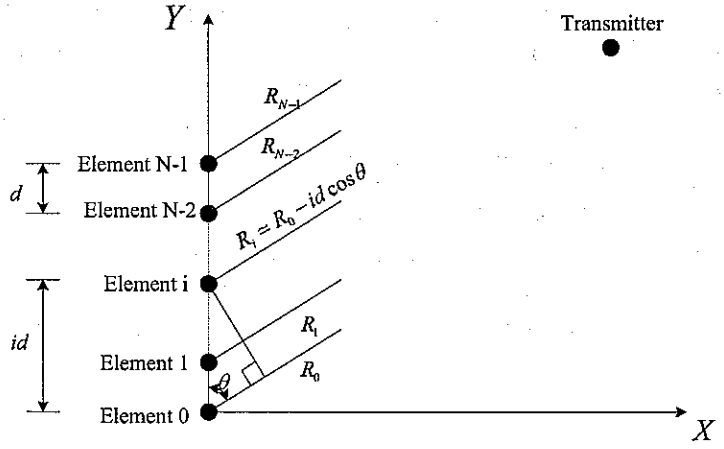

Fig. x.2. Antenna array of N antenna elements

In the absence of noise and interference, bearing lines from two or more receivers will intersect to determine a unique location, which becomes the location estimate of the transmitter. In the presence of noise, more than two bearing lines will not intersect at a single point. The reader may refer to [10] for a detailed discussion on location estimation techniques with noisy measurements.

2.3 Propagation time based measurements

Propagation time based measurements can be further divided into three subclasses: one-way propagation time, roundtrip propagation time and time-difference-of-arrival (TDOA) measurements.

2.3.1 One-way propagation time and roundtrip propagation time measurements

Both one-way propagation time and roundtrip propagation time measurements have been widely used for estimating distance between neighbouring sensors. The principle of one-way propagation time measurements is straightforward. By measuring the difference between the sending time and the receiving time of a signal, the distance between the transmitter and the receiver can be obtained. However, one-way propagation time measurements require very accurate synchronization between the transmitter and the receiver. This requirement significantly adds to the cost of the sensor. Moreover, due to the high propagation speed of wireless signals, a small error in propagation time measurements may cause a large error in the distance estimate. These disadvantages make one-way propagation time measurements a less attractive option. In [11], Priyantha *et al.* obtained the one-way propagation time measurements by using a combination of RF and ultrasound hardware. On each transmission, a transmitter concurrently sends an RF

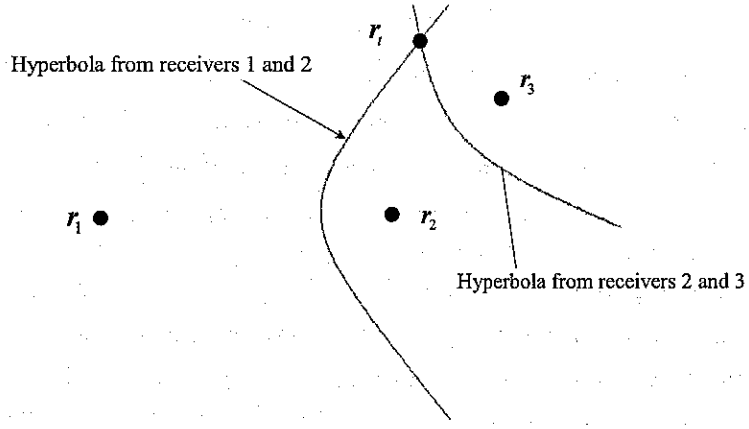

Fig. x.4. Intersecting hyperbolas from three receivers. The transmitter is located at the intersection of hyperbola obtained from TDOA measurement of receivers 1 and 2 and hyperbola obtained from TDOA measurement of receivers 2 and 3

Given the TDOA measurement Δt_{ij} and the coordinates of receivers i and j, (1.7) defines one branch of a hyperbola whose foci are at the locations of receivers i and j and on which $\boldsymbol{r}_t$ must lie. In $\Re^2$, measurements from a minimum of three receivers are required to uniquely determine the location of the transmitter. This is illustrated in Fig. x.4. In practice, Δt_{ij} is not available; instead we have the noisy TDOA measurement $\Delta \tilde{t}_{ij}$ defined by:

$$\Delta \tilde{t}_{ij} = \Delta t_{ij} + n_{ij} = \frac{1}{c}\left(\| \boldsymbol{r}_i - \boldsymbol{r}_t \| - \| \boldsymbol{r}_j - \boldsymbol{r}_t \|\right) + n_{ij} \tag{1.8}$$

where n_{ij} denotes an additive noise, which is usually assumed to be a Gaussian distributed random variable with zero mean. In a system consisting of N receivers, there are $N-1$ linearly independent TDOA equations like (1.8). A maximum likelihood solution to the transmitter location can be obtained numerically by using an iterative gradient descent algorithm [10]. Recently, Doğançay developed a closed-form transmitter location estimator using TDOA measurements based on triangulation of hyperbolic asymptotes [13].

In a wireless sensor network, which measurement technique to use for location estimation will depend on the specific application. Typically, both AOA and propagation time based measurements are able to achieve better accuracy than RSS based measurements. However, that accuracy is achieved at the expense of higher equipment cost. Patwati *et al.* gave a Cramér-Rao lower bound for location estimation using some measurement techniques in [14].

3 Connectivity-based Localization Algorithms

It is worth noting that there is a class of localization algorithms that do not rely on any of the measurement techniques in the last section. They are the so-called connectivity-based or "range free" localization algorithms. They use the connectivity information, i.e., "who is within the communications range of whom" [15] to derive the locations of the non-anchor nodes.

In [16], Doherty *et al.* formulated the connectivity-based localization problem as a convex optimization problem and locations of the non-anchor nodes are obtained using polynomial-time numerical algorithms based on interior point methods for solving linear programs or semidefinite programs (SDP). A linear program is a program of the form:

and four randomly placed anchors showed a localization error of 0.35. The average node degree is 10. The MDS algorithm can be easily extended to incorporate distance measurements for location estimation. Shang *et al.* further improved the algorithm in [19]. The main idea is to compute a local map using MDS for each node consisting only of nearby nodes, and then to merge these local maps together to form a global map. The improved algorithm avoids using the shortest path between far away nodes and can be implemented in a distributed fashion. It performs better than the original method on irregularly-shaped networks.

Generally, the connectivity-based localization algorithms are able to obtain a coarse grained estimate of each node's location. The localization error decreases with increasing density of the network and the number of anchors. The localization error is smaller in a regular network topology and becomes much larger in an irregular network topology. Although not able to achieve an accurate estimate of location, the simplicity of the connectivity-based localization algorithms makes them attractive for applications requiring an approximate location estimate only.

4 Distance-Based Localization Algorithms

In the remaining sections, we shall focus on distance-based localization algorithms. Distance-based localization algorithms can be divided into centralized algorithms and distributed algorithms. In centralized algorithms, all measurements are sent to a central processor where location estimation is performed. In distributed algorithms, there is no central processor. Each non-anchor node estimates its location using measurements between neighbouring nodes and the location estimates of its neighbours.

Centralized algorithms can be used in applications where a centralized architecture already exists. For example, in applications such as health monitoring, road traffic monitoring and control, environmental monitoring and precision agriculture, a central system already exists, which gathers information from all nodes in the network. In that case, it is convenient to piggyback the distance measurements onto the monitoring information sent to the central processor. Centralized algorithms are likely to provide more accurate location estimates than distributed algorithms. However centralized algorithms suffer from the scalability problem and generally are not suitable for a large scale sensor network. Centralized algorithms also suffer from higher computational complexity, and lower reliability which is caused by multi-hop transmission over a wireless network. In comparison, distributed algorithms are more scalable and have lower computational complexity. However distributed algorithms are difficult to design because of the potentially complicated relationship between local behaviour and global behaviour. Algorithms that are locally optimal may not perform well in a global sense. How to optimally distribute the computation of a centralized algorithm in a distributed implementation continues to be a research problem. Moreover, distributed algorithms generally require multiple iterations to converge to a stable solution. The relatively slow convergence speed of distributed algorithms may be a concern in some applications. Error propagation is also a potential problem in distributed algorithms.

Another important factor affecting the choice between centralized algorithms and distributed algorithms is communication cost. Sensors are energy constrained. Depending on the hardware and the transmission range, the energy required for transmitting a single bit could be used to execute 1,000 to 2,000 instructions [20]. Centralized algorithms in large networks require each sensor's measurements to be sent over many hops to a central processor. Distributed algorithms require only local information exchange between neighboring nodes but possibly many such exchanges are required, depending on the number of iterations needed to arrive at a stable solution. The energy efficiency of centralized and distributed algorithms was compared in [21]. In general, when the average number of hops to the central processor exceeds the necessary number of iterations, distributed algorithms will likely save communication energy costs.

normalized by the transmission range. When the transmission range is 1.1, 1.2, 1.4, 1.6, 1.8 and 2.0, the corresponding average node degree is 6.86, 8.19, 10.87, 13.96, 17.87 and 21.36 respectively. The following noise model is used in the simulation:

$$\tilde{d}_{ij} = d_{ij}\left(1+\rho\times n\right) \tag{1.12}$$

where n is a random noise with normal distribution and $\rho = 0.1$.

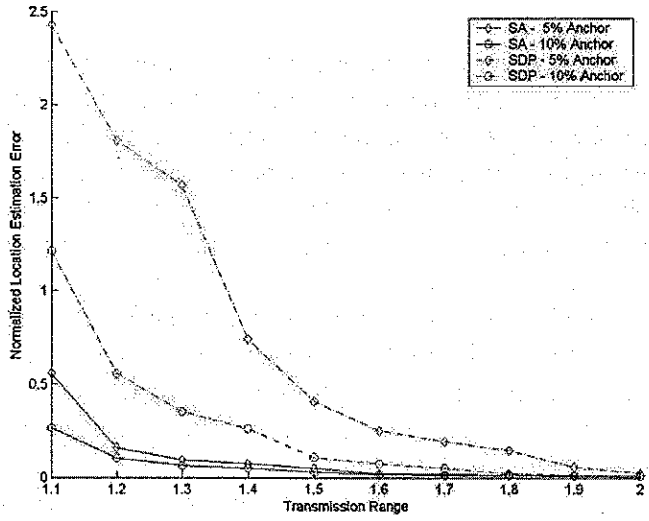

Fig. x. 5. Performance of SA algorithm with flip ambiguity mitigation and SDP algorithm with gradient search improvement

As shown in the figure, the SA algorithm has better accuracy than the SDP algorithm with gradient search however this improvement is achieved at the expense of higher computation cost. A possible reason for the better performance is the SA algorithm is robust against being trapped into a local minimum.

4.2 Distributed algorithms

The "DV-hop" algorithm developed by Niculescu *et al.* [18] can be readily modified to include distance measurements into the location estimation. This is done by letting neighbouring nodes propagate measured distance instead of hop count. The corresponding algorithm is referred to as the "DV-distance" algorithm [18].

Savarese *et al.* developed a two-stage localization algorithm in [27]. In the first stage, a "Hop-terrain" algorithm, which is similar as the "DV-hop" algorithm, is used to obtain an initial estimate of the nodes' locations. In the second stage, the measured distances between neighbouring nodes are used to refine the initial location estimate. To mitigate location estimation errors caused by error propagation and unfavourable network topologies, a confidence value is assigned to each node's position. An anchor has a higher confidence value (close to 1) and a non-anchor node with few neighbours and poor constellation has lower confidence value (close to 0). Simulation using 400 nodes uniformly placed in a 100 by 100 square showed that the algorithm is able to achieve an average location estimation error of less than 33% of the transmission range in the presence of 5% distance measurement error (normalized by the transmission range) and 5% anchors. The average node degree is greater than 7.

The localization algorithm developed by Savvides *et al.* [28] is divided into four stages. In the first stage, those nodes whose locations are "tentatively unique" are selected based on the assumption that solution to a node location is "tentatively unique" if it has at least three neighbours that are either non-collinear anchors or their solutions are "tentatively unique". The locations of these tentatively uniquely localizable nodes are estimated in stage two and three. In the second stage, each non-anchor node obtains the estimated distances to at least three anchors using a "DV-distance" like algorithm. An estimated distance to an anchor node

$N(S,D,S_a)$ with a set S of sensor nodes, a set D of known distances d_{ij} between certain pairs of nodes $s_i, s_j \in S$, and a set $S_a \subset S$ of ***anchor*** nodes s_{a_i} ($a_i \in \{1,\ldots,|S|\}, i \in \{1,\ldots,|S_a|\}$)[1], whose coordinates $\boldsymbol{r}_i$ are known. Find a mapping $p: S \to \Re^d$ ($d \in \{2,3\}$) that assigns a d-dimensional coordinate to each node $s_i \in S$ such that $\|p(s_i) - p(s_j)\| = d_{ij}$ holds for all node pairs $s_i, s_j \in S$ for which d_{ij} is given (i.e., $d_{ij} \in D$), and the assignment is consistent with any (***anchor***) node assignments provided, i.e., $p(s_{a_i}) = \boldsymbol{r}_i, \forall s_{a_i} \in S_a$.

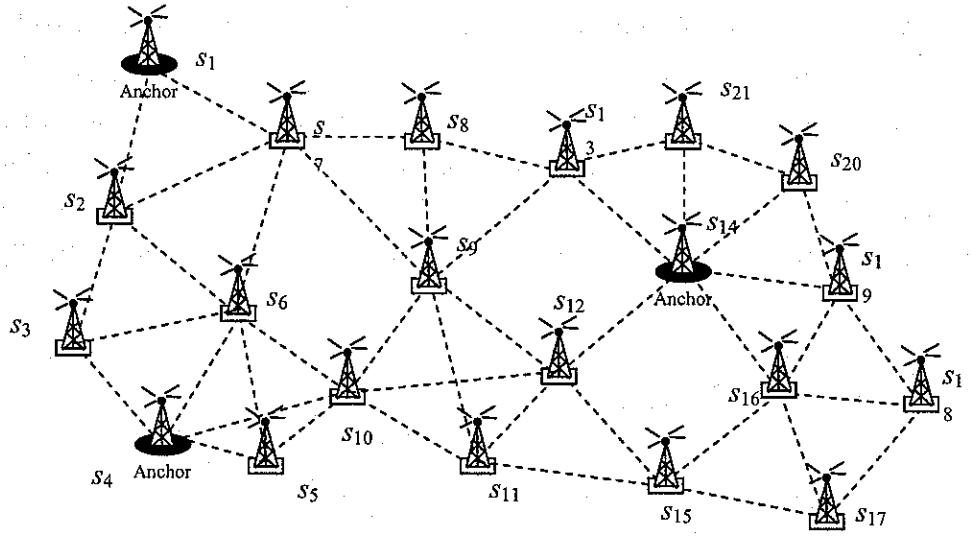

Fig. x. 6. A two-dimensional sensor network localization problem: Absolute positions $\boldsymbol{r}_i$ of the three anchor sensors $s_{a_i}, a_1 = 1, a_2 = 4, a_3 = 14$ are known. The distance d_{ij} between each sensor pair s_i, s_j connected with a (dashed) line segment is measurable. The task is to estimate the absolute position of every sensor s_i in the network.

The sensor network localization problem, which is formulated in Problem 1 and illustrated in Fig. x.6, can be split up into an analytic existence/solvability problem and an algorithmic problem. The analytic problem is to determine the properties of a sensor network that ensure unique solvability of the localization problem. The algorithmic problem is to find a method of solving the localization problem and determine the computational complexity of this method. A specific pair of questions to be answered related to the algorithmic problem are how to deal with the presence of errors in the inter-sensor measurements and how such errors translate into errors in the algorithm's output of sensor coordinates.

5.2 A Graph Theoretical Framework

In the sequel, both the analytic and the algorithmic problems above will be studied in the framework of graph theory, particularly using the notions of ***rigid*** graph theory. A sensor network $N(S,D,S_a)$, as described in Problem 1, can be represented by a graph $G = (V,E)$ with a vertex set V and an edge set E, where each vertex $i \in V$ is associated with a sensor node s_i in the network, and each edge $(i,j) \in E$ corresponds to a sensor pair s_i, s_j for which the inter-sensor distance d_{ij} is known. In this case, we call $G = (V,E)$ the ***underlying graph*** of the sensor network $N(S,D,S_a)$.

A d-dimensional ($d \in \{2,3\}$) ***representation*** of a graph $G = (V,E)$ is a mapping $\overline{p}: V \to \Re^d$. Given a

[1] In this chapter, $|\overline{S}|$ for a given set $\overline{S}$ denotes the number of elements in $\overline{S}$.

$(G,\overline{p})$ is called ***rigid*** if there exists a sufficiently small positive constant ε such that if $(G,\overline{q})$ is equivalent to $(G,\overline{p})$ and $\|\overline{p}(i)-\overline{q}(i)\|<\varepsilon$ for all $i\in V$ then $(G,\overline{q})$ is congruent to $(G,\overline{p})$. Intuitively, a rigid framework cannot flex, and if a framework $(G,\overline{p})$ (with a distance set $\overline{D}$) is non-rigid then continuous deformations can be applied to produce an infinite number of different realizations of G (for $\overline{D}$).

Note that there exist rigid frameworks $(G,\overline{p})$ and $(G,\overline{q})$ which are equivalent but not congruent, as demonstrated in Fig. x.8. A framework $(G,\overline{p})$ is ***globally rigid*** if every framework which is equivalent to $(G,\overline{p})$ is congruent to $(G,\overline{p})$. It is easy to see that if G is a complete graph then the framework $(G,\overline{p})$ is necessarily globally rigid.

If a framework $(G,\overline{p})$ (with a distance set $\overline{D}$) is rigid but not globally rigid, like the ones in Fig. x.8, although a continuous deformation does not exist as in the case of non-rigid frameworks, there are two types of discontinuous deformations that can prevent a realization of G (for the distance set $\overline{D}$) from being unique (in the sense that it differs from the other possible realizations of the same graph at most by translation, rotation or reflection) [37, 38]: ***flip*** and ***discontinuous flex ambiguities***.

In ***flip ambiguities*** in $\Re^d$ $(d\in\{2,3\})$, a vertex (sensor node) v has a set of neighbours which span a $(d-1)$-dimensional subspace, e.g., v has only d neighbours; which leads to the possibility of the neighbours forming a mirror through which v can be reflected. Fig. x.8(a) depicts an example of flip ambiguity. In ***discontinuous flex ambiguities*** in $\Re^d$ $(d\in\{2,3\})$, the removal of an edge allows the remaining part of the graph to be flexed to a different realization (which cannot be obtained from the original realization by translation, rotation or reflection) and the removed edge reinserted with the same length. Fig. x.8(b) depicts an example.

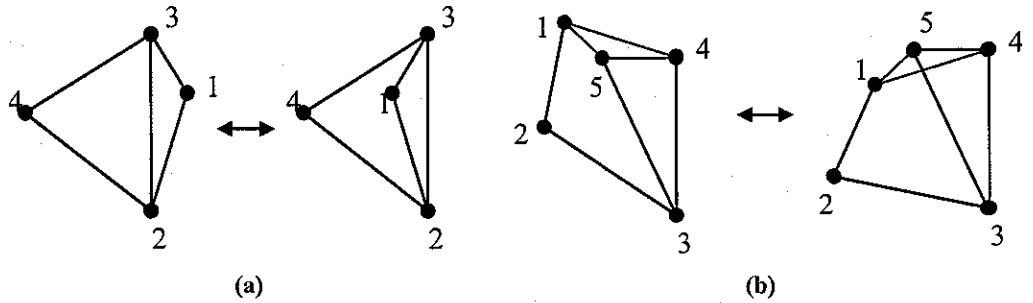

Fig. x.8. Two pairs of equivalent rigid frameworks in $\Re^2$ which are not congruent: **(a)** The frameworks suffer from flip ambiguity: Vertex 1 can be reflected across the edge (2,3) to a new position without violating the distance constraints. **(b)** The frameworks suffer from discontinuous flex ambiguity: Removing the edge (1,4), flexing the edge triple (1,5),(1,2),(2,3), and reinserting the edge (1,4) so that the distance constraints are not violated in the end, we obtain a new realization

Knowing the graph and distance set of a globally rigid framework is not sufficient to position the framework absolutely in $\Re^d$ $(d\in\{2,3\})$. In order to do this, we need the absolute positions of at least three vertices in $\Re^2$ or four vertices in $\Re^3$ as well, and in fact they must be ***generically positioned***, i.e., these vertices should not lie on the same line in $\Re^2$ or the same plane in $\Re^3$. Fig. x.9 illustrates a case in $\Re^2$, where the three vertices with known absolute positions are collinear. For this example, there are two possible absolute positions for the whole framework, which are symmetric with respect to the line passing through the three collinear vertices with known absolute positions.

to have a finite number of solutions. However, in some cases where $(G, \overline{p})$ is rigid but not globally rigid, some additional *a priori* information may compensate the need for global rigidity. For example, assume that a sensor network can be represented by a unit disk graph, where there is an edge between two nodes (the nodes can sense each other) if and only if the distance between them is less than a certain threshold $R > 0$. Then the ambiguities due to the non-globally rigid nature of the underlying graph may sometimes be eliminated using the unit disk graph properties as demonstrated in Fig. x.10.

Given a rigid framework (G, p) in $\Re^d$ $(d \in \{2,3\})$, if p is ***generic***, for example, if no combination of $d+1$ vertices in G lies on a $d-1$ dimensional hyperplane (which is a line in $\Re^2$ and a plane in $\Re^3$), then the graph G alone determines the rigidity of the framework, and so we can define ***generic rigidity of a graph***. Note that there is a linear algebraic test for rigidity based on the rank of a matrix, called the rigidity matrix, whose entries are formed from the coordinates of the vertices. In $\Re^2$, there exists the following alternative combinatorial (essentially graph theoretic) necessary and sufficient condition for rigidity, termed Laman's Theorem [39, 40].

Theorem 2 (*Laman's Theorem* [39, 40]): A graph $G = (V, E)$ in $\Re^2$, where $|V| > 1$, is generically rigid if and only if there exists a subset $E' \subseteq E$ satisfying the following two conditions:

(i) $|E'| = 2|V| - 3$

(ii) For any non-empty subset $E'' \subseteq E'$, $|E''| \leq 2|V(E'')| - 3$, where $V(E'')$ denotes the set of all end-vertices of the edges in E''.

No such combinatorial necessary and sufficient condition is currently available in $\Re^3$, where the obvious generalizations of the two dimensional results are only necessary, but not sufficient [40], i.e., we only have the following necessary counterpart in $\Re^3$:

Theorem 3 [40]: If a graph $G = (V, E)$ in $\Re^3$, where $|V| > 2$, is generically rigid, then there exists a subset $E' \subseteq E$ satisfying the following two conditions:

(i) $|E'| = 3|V| - 6$

(ii) For any non-empty subset $E'' \subseteq E'$, $|E''| \leq 3|V(E'')| - 6$, where $V(E'')$ denotes the set of all end-vertices of the edges in E''.

We can define ***generic global rigidity of a graph*** in $\Re^2$ in a similar way [41]. In order to state a combinatorial test for generic global rigidity in $\Re^2$, we need to define two other terms: For any positive integer k, a graph is called ***k-connected*** if between any two vertices of the graph, there exist at least k paths which have no edge or vertex in common, or equivalently, it is not possible to find $k-1$ vertices whose removal (together with the removal of the edges incident on them) would render the graph unconnected [42]. A graph is termed ***generically redundantly rigid*** if with the removal of any edge, it remains generically rigid [35, 43]. In $\Re^2$, there is a variant of Laman's Theorem for checking generic redundant rigidity; the reader may refer to [37, 43] for details. Based on these definitions, the following theorem states an elegant necessary and sufficient condition for ***generic global rigidity*** of a framework (i.e., global rigidity of a generic framework) in $\Re^2$:

Theorem 4 [43]: A graph $G = (V, E)$ in $\Re^2$ with $|V| \geq 4$ vertices is generically globally rigid if and only if it is 3-connected and redundantly rigid.

In $\Re^3$, it is necessary that a graph be 4-connected and generically redundantly rigid for the graph to be generically globally rigid. However, these conditions are known to be insufficient [41]. No necessary and sufficient condition for generic global rigidity has been established in $\Re^3$ yet. In fact, in contrast to

law is proposed in [33] for the cases where each sensor can determine the angle between two of its neighbours in addition to the distances to those neighbours. In such a case, noting that the cosine law allows determination of the distance between any two neighbours of each sensor and that any pair of neighbours of a given sensor are either neighbours of each other or at a two hop distance from each other, all two-hop distances can be determined.

5.6 Localization of Large-Scale Sensor Networks

Modelling sensor networks with a large number of sensor nodes using the deterministic graph settings of the previous subsections is not feasible in general, due to various sources of uncertainties. An often appropriate way of abstracting such large-scale networks is found to be via ***random geometric graph*** notions [34, 35].

A 2-dimensional ***random geometric graph*** $G_n(R)$ parameterized by a pair of positive real numbers n, R is a random graph framework $((V,E),\overline{p})$ where V is a random vertex set and $\overline{p}$ is a random realization such that each vertex $i \in V$ with realization $\overline{p}(i)$ is a random point in $[0,1]^2$ generated by a 2-dimensional Poisson process of intensity n, and E is a random edge set, for any pair $i \neq j \in V$, satisfying $(i,j) \in E$ if and only if $|\overline{p}(i) - \overline{p}(j)| < R$. The parameters n and R are called the ***sensor density*** and the ***sensing radius***, respectively.

Modelling of a large-scale sensor network using a random geometric graph allows analysis of the probability (of the underlying graph) of the network being connected, k-connected, globally rigid, etc. It further allows one to analyze the effects of changing the sensing radius on such probabilities. The results of a series of such analyses can be found in [34, 35, 46]. Next, we present some of these results together with the implications of the earlier theorems in the section for random geometric graphs.

Remark 3: Theorem 5 implies that (in $\Re^2$), for a given sensor density – sensing radius pair n, R, the probability of $G_n(2R)$ to be globally rigid is greater than or equal to the probability of $G_n(R)$ being 2-connected.

Theorem 8 [34, 35]**:** Consider a set of random graphs $G_n(R)$ with n varying, and with R dependent on n and such that for all n, there holds $\frac{nR^2}{8\log n} \geq 1+\varepsilon$ for some positive ε. Then the probability that $G_n(R)$ is not a trilateration graph tends to zero as $n \to \infty$.

Theorem 9 [34, 35]**:** For any positive integer (sensor density) n, there exists some sensing radius $R \in O\left(\sqrt{\frac{\log n}{n}}\right)$ such that a realization of the 2-dimensional random geometric graph $G_n(R)$ is computable in linear time if the positions of three vertices in $G_n(R)$ connected with edges to each other are known.

Further discussions about sensor network localization protocols based on the above results as well as the analysis of 3-dimensional large-scale sensor networks, as opposed to 2-dimensional ones, can be found in [34, 35]. As a relevant issue an analysis of the localization of so-called ***partially localizable sensor networks***, where some of the sensors in the network cannot be uniquely localized, can be seen in [36].

5.7 Complexity of Rigid Graph Theory Based Localization Algorithms

The computational complexity of distance-based network localization algorithms has been investigated in the literature (see, e.g., [34, 35, 45]). In general, the computational complexity of localization is exponential in the number of sensor nodes [45]. As mentioned in Section 5.3, this conclusion is particularly valid for sensor networks whose underlying graphs are unit disk graphs. Nevertheless, there are exceptional classes of sensor network underlying graphs, for which this conclusion does not apply. The trilateration and

localization problems as indicated in Theorem 1, the above inaccuracy sources may cause a localization algorithm to suffer from certain ambiguities described in Section 5.3, which are typical in non-globally rigid (but rigid) settings. Examples of such situations are given in [38]. A sample situation where measurement errors may cause flip ambiguity problems is depicted in Fig. x.11.

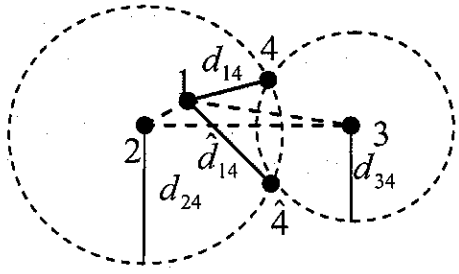

Fig. x.11. Measurement noises causing a flip ambiguity in a globally rigid setting: Sensor node 4 is desired to be localized employing the known positions of nodes 1,2,3 and measurements of the distances between node pairs (4,1), (4,2), and (4,3). The distance measurements d_{24} and d_{34} constrain the position of node 4 to lie on one of the two intersection points of the circles shown in the figure. An erroneous measurement of the distance d_{14} as $\hat{d}_{14}$ will cause node 4 to be located as $\hat{4}$

A recent work focusing on robust distributed localization of sensor networks with certain measurement/estimation errors of the above type and ambiguities caused by these errors is presented in [38]. In this paper, certain criteria are provided in selection of the subgraphs of the underlying graph of a network to be used in a localization algorithm robust against such errors. The analysis in [38], however, is not complete and there may be other criteria that may better characterize robustness of a given sub-network against distance measurement and location estimation errors in localization, which both show future research directions in the area. A closely related current problem is to understand how graph properties characterize sensor networks with robust localization to (complete) sensor failures rather than measurement and estimation inaccuracies.

Another area of active research is understanding and utilizing the data and error propagation characteristics in a sensor network. Some of the available tools related to this area can be seen in [32]. Using the tools here together with some other methods, approaches can be developed to localization based on analysis of accumulation of gross (ambiguity-type) errors of low probability and small errors of high probability.

It is worth to recall the idea of using trilateration islands stated in Section 5.7 to reduce the computational complexity of localization of networks having certain classes of globally rigid underlying graphs which are not trilateration graphs. Given such an underlying graph, the problem of partitioning the graph to trilateration islands in an optimal way is open. Once such a partitioning is known, development of an optimal algorithm employing trilateration ideas is another future research direction.

7 SUMMARY

Most wireless sensor network applications require knowing or measuring locations of thousands of sensors accurately. Sensor network localization is an important area that attracted significant research interest recently. This interest is expected to grow further with the proliferation of wireless sensor network applications.

In this chapter we discussed three major categories of measurement techniques in sensor network localization, i.e., received signal strength measurements, angle of arrival measurements, and propagation time based measurements. The localization algorithms built on these measurement techniques were also discussed. In a wireless sensor network, which measurement technique and algorithm to use for location

[23] Biswas P and Ye Y (2004) Semidefinite programming for ad hoc wireless sensor network localization. Third International Symposium on Information Processing in Sensor Networks, pp 46-54
[24] Liang T-C, Wang T-C, and Ye Y (2004) A gradient search method to round the semidefinite programming relaxation for ad hoc wireless sensor network localization Standford University, Report No. August 26
[25] Kannan AA, Mao G, and Vucetic B (2005) Simulated Annealing based Localization in Wireless Sensor Network. The 30th IEEE Conference on Local Computer Networks, pp 513-514
[26] Kannan AA, Mao G, and Vucetic B (2006) Simulated Annealing based Wireless Sensor Network Localization with Flip Ambiguity Mitigation. to appear in IEEE Vehicular Technology Conference 2006,
[27] Savarese C and Rabaey J (2002) Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks. Proceedings of the General Track: 2002 USENIX Annual Technical Conference, pp 317-327
[28] Savvides A, Park H, and Srivastava MB (2002) The bits and flops of the n-hop multilateration primitive for node localization problems. International Workshopon Sensor Networks Application, pp 112-121
[29] Langendoen K and Reijers N (2003) Distributed localization in wireless sensor networks: a quantative comparison, Computer Networks, vol. 43, pp 499-518
[30] Capkun S, Hamdi M, and Hubaux J (2001) GPS-free positioning in mobile ad-hoc networks. 34th Hawaii International Conference on System Sciences, pp 3481-3490
[31] KWOK C, FOX D, and MEILA M (2004) Real-time particle filters, Proceedings of the IEEE, vol. 92, pp 469-484
[32] Ihler AT, Fisher JW, III, Moses RL, and Willsky AS (2005) Nonparametric belief propagation for self-localization of sensor networks, IEEE Journal on Selected Areas in Communications, vol. 23, pp 809-819
[33] Anderson BDO, Belhumeur PN, Eren T, Goldenberg D, Morse AS, Whiteley W, and Yang RY (2005) Graphical properties of easily localizable sensor networks, submitted for publication in Wireless Networks
[34] Eren T, Goldenberg D, Whiteley W, Yang RY, Morse AS, Anderson BDO, and Belhumeur PN (2004) Rigidity and randomness in network localization. IEEE INFOCOM, pp 2673-2684
[35] Aspnes J, Eren T, Goldenberg DK, Morse AS, Whiteley W, Yang YR, Anderson BDO, and Belhumeur PN (2005) A theory of network localization, submitted to IEEE Transactions on Mobile Computing
[36] Goldenberg DK, Krishnamurthy A, MAness WC, Yang RY, Young A, Morse AS, Savvides A, and Anderson BDO (2005) Network localization in partially localizable networks. IEEE INFOCOM, pp 313 - 326
[37] Hendrickson B (1992) Conditions for unique graph realizations, SIAM J. Comput., vol. 21, pp 65-84
[38] Moore D, Leonard J, Rus D, and Teller S (2004) Robust distributed network localization with noisy range measurements. the 2nd ACM Conf. on Embedded Networked Sensor Systems (SenSys'04), pp 50-61
[39] Laman G (1970) On graphs and rigidity of plane skeletal structures, Journal of Engineering Mathematics, vol. 4, pp 331-340
[40] Whiteley W (1996) Some matroids from discrete applied geometry in Contemporary Mathematics. vol. 197, J. E. Bonin, J. G. Oxley, and B. Servatius, Eds.: American Mathematical Society
[41] Connelly R (2005) Generic global rigidity, Discrete and Computational Geometry, vol. 33, pp 549-563
[42] Godsil C and Royle G (2001) Algebraic Graph Theory. New York: Springer-Verlag
[43] Jackson B and Jordan T (2005) Connected rigidity matroids and unique realizations of graphs, J. Combinatorial Theory Series B, vol. 94, pp 1-29
[44] Fang J, Cao M, Morse AS, and Anderson BDO (2006) Sequential Localization of Networks. submitted to the 17th International Symposium on Mathematical Theory of Networks and Systems- MTNS 2006,
[45] Saxe J (1979) Embeddability of weighted graphs in k-space is strongly NP-hard. 17th Allerton Conference in Communications, Control and Computing, pp 480-489
[46] Penrose MD (1999) On k-connectivity for a geometric random graph, Random Structures and Algorithms, vol. 15, pp 145-164
[47] Cao M, Anderson BDO, and Morse AS (2005) Localization with imprecise distance information in sensor networks. Proc. Joint IEEE Conf on Decision and Control and European Control Conf., pp 2829-2834
[48] Havel TF, Kuntz ID, and Crippen GM (1983) The Theory and Practice of Distance Geometry, Bulletin of Mathematical Biology, vol. 45, pp 665-720