Location, Localization, and Localizability

Yunhao Liu (刘云浩), *Member, ACM, Senior Member, IEEE*, Zheng Yang (杨　铮), *Student Member, ACM, IEEE*
Xiaoping Wang (王小平), *Student Member, IEEE*, and Lirong Jian (简丽荣), *Student Member, IEEE*

Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, China

E-mail: {liu, yangzh, xiaopingwang, jlrphx}@cse.ust.hk

Abstract   Location-aware technology spawns numerous unforeseen pervasive applications in a wide range of living, production, commence, and public services. This article provides an overview of the location, localization, and localizability issues of wireless ad-hoc and sensor networks. Making data geographically meaningful, location information is essential for many applications, and it deeply aids a number of network functions, such as network routing, topology control, coverage, boundary detection, clustering, etc. We investigate a large body of existing localization approaches with focuses on error control and network localizability, the two rising aspects that attract significant research interests in recent years. Error control aims to alleviate the negative impact of noisy ranging measurement and the error accumulation effect during cooperative localization process. Network localizability provides theoretical analysis on the performance of localization approaches, providing guidance on network configuration and adjustment. We emphasize the basic principles of localization to understand the state-of-the-art and to address directions of future research in the new and largely open areas of location-aware technologies.

Keywords   location-based services (LBS), localization, error control, localizability, wireless ad-hoc and sensor networks

1   Location

The proliferation of wireless and mobile devices has fostered the demand for context-aware applications, in which location is viewed as one of the most significant contexts. For example, pervasive medical care is designed to accurately record and manage patient movements[1-2]; smart space enables the interaction between physical space and human activities[3-4]; modern logistics has major concerns on goods transportation, inventory, and warehousing[5-6]; environmental monitoring networks sense air, water, and soil quality and detect the source of pollutants in real time[7-11]; and mobile peer-to-peer computing encourages content sharing and contributing among mobile hosts in the vicinity[12-13]. In brief, location-based service (LBS) is a key enabling technology of these applications and widely exists in nowadays wireless communication networks from the short-range Bluetooth to the long-range telecommunication networks, as illustrated in Fig.1.

Recent technological advances have enabled the development of low-cost, low-power, and multifunctional sensor devices. These nodes are autonomous devices with integrated sensing, processing, and communication capabilities. With the rapid development of wireless sensor networks (WSNs), location information
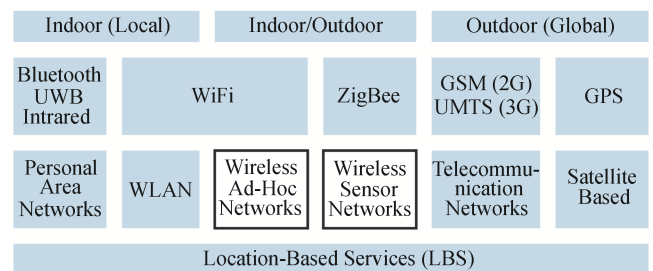

Fig.1. Location-based services for a wide range of wireless networks.

becomes critically essential and indispensable. The overwhelming reason is that WSNs are fundamentally intended to provide information on spatial-temporal characteristics of the physical world; hence, it is important to associate sensed data with locations, making data geographically meaningful. For example, a number of applications, such as object tracking and environment monitoring, inherently rely on location information. A detailed survey on location-based applications can be found in [14-15].

Location information also supports many fundamental network services, including network routing, topology control, coverage, boundary detection, clustering, etc. We give a brief overview as follows.

- *Routing*

Routing is a process of selecting paths in a network along which to send data traffic. Most routing protocols for multi-hop wireless networks utilize physical locations to construct forwarding tables and deliver messages to the node closer to the destination in each hop[16]. Specifically, when a node receives a message, local forwarding decisions are made according to the positions of the destination and its neighboring nodes. Such geographic routing schemes require localized information, making the routing process stateless, scalable, and low-overhead in terms of route discovery.

- *Topology Control*

Topology control is one of the most important techniques used in wireless ad-hoc and sensor networks for saving energy and eliminating radio interference[17-18]. By adjusting network parameters (e.g., the transmitting range), energy consumption and interference can be effectively reduced; meanwhile some global network properties (e.g., connectivity) can still be well retained. Importantly, using location information as a priori knowledge, geometry techniques (e.g., spanner subgraphs and Euclidean minimum spanning trees) can be immediately applied to topology control[17].

- *Coverage*

Coverage reflects how well a sensor network observes the physical space; thus, it can be viewed as the quality of service (QoS) of the sensing function. Previous designs fall into two categories. The probabilistic approaches[19-21] analyze the node density for ensuring appropriate coverage statistically, but essentially have no guarantee on the result. In contrast, the geometric approaches[22] are able to obtain accurate and reliable results, in which locations are essential.

- *Boundary Detection*

Boundary detection is to figure out the overall boundary of an area monitored by a WSN. There are two kinds of boundaries: the outer boundary showing the under-sensed area, and the inner boundary indicating holes in a network deployment. The knowledge of boundary facilitates the design of routing, load balancing, and network management[23]. As direct evidence, location information helps to identify border nodes and further depict the network boundary.

- *Clustering*

To facilitate network management, researchers often propose to group sensor nodes into clusters and organize nodes hierarchically[24]. In general, ordinary nodes only talk to the nodes within the same cluster, and the inter-cluster communications rely on a special node in each cluster, which is often called cluster head. Cluster heads form a backbone of a network, based on which the network-wide connectivity is maintained.

Clustering brings numerous advantages on network operations, such as improving network scalability, localizing the information exchange, stabilizing the network topology, and increasing network life time. Among all possible solutions, location-based clustering approaches are greatly efficient by generating non-overlapped clusters. In addition, location information can also be used to rebuild clusters locally when new nodes join the network or some nodes suffer from hardware failure[24].

2 Localization

Network localization has attracted a lot of research efforts in recent years. One method to determine the location of a device is through manual configuration, which is often infeasible for large-scale deployments or mobile systems. As a popular system, Global Positioning System (GPS) is not suitable for indoor or underground environments and suffers from high hardware cost. Local Positioning Systems (LPS) rely on high-density base stations being deployed, an expensive burden for most resource-constrained wireless ad hoc networks.

The limitations of existing positioning systems motivate a novel scheme of network localization, in which some special nodes (a.k.a. anchors or beacons) know their global locations and the rest determine their locations by measuring the geographic information of their local neighboring nodes. Such a localization scheme for wireless multi-hop networks is alternatively described as "cooperative", "ad-hoc", "in-network localization", or "self localization", since network nodes cooperatively determine their locations by information sharing.

In this section, we first review the state-of-the-art localization approaches from two aspects: physical measurements and network-wide localization algorithms. We then discuss a number of techniques for controlling localization errors caused by noisy physical measurements and algorithmic defects.

Almost all existing localization algorithms consist of two stages: 1) measuring geographic information from the ground truth of network deployment; 2) computing node locations according to the measured data. Geographic information includes a variety of geometric relationships from coarse-grained neighbor-awareness to fine-grained inter-node rangings (e.g., distance or angle). Based on physical measurements, localization algorithms solve the problem that how the location information from beacon nodes spreads network-wide. Generally, the design of localization algorithms largely depends on a wide range of factors, including resource availability, accuracy requirements, and deployment restrictions; and no particular algorithm is an absolute favorite across the spectrum.

276

J. Comput. Sci. & Technol., Mar. 2010, Vol.25, No.2

2.1 Physical Measurements and Single-Hop Positioning

Clearly, it is difficult, if not infeasible, to do localization without knowledge of the physical world. According to the capabilities of diverse hardwares, we classify the measuring techniques into six categories (from fine-grained to coarse-grained): location, distance, angle, area, hop count, and neighborhood, as shown in Fig.2.

Fine-Grained   Coarse-Grained

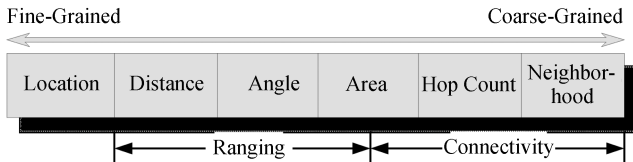| Location	Distance	Angle	Area	Hop Count	Neighbor-hood

←—— Ranging ——→ ←—— Connectivity ——→

Fig.2. Physical measurements.

Among them, the most powerful physical measurement is directly obtaining the position without any further computation. GPS is such a kind of infrastructure. The other five measurements are used in the scenarios of positioning an unknown node by given some reference nodes. The terms of "reference" and "unknown" nodes refer to the nodes being aware and being NOT aware of their locations, respectively. Distance and angle measurements are obtained by ranging techniques. Hop count and neighborhood are basically based on radio connectivity. In addition, area measurement relies on either ranging or connectivity, depending on how the area constrains are formed.

2.1.1 Distance Measurements

The distances from an unknown node to several references constrain the presence of this node, which is the basic idea of the so called multilateration. Fig.3 shows an example of trilateration, a special form of multilateration which utilizes exact three references. A to-be-located node (node 0) measures the distances from itself to three references (nodes 1, 2, 3). Obviously, node 0 should locate at the intersection of three circles centered at each reference position. The result of trilateration is unique as long as three references are non-linear.

Suppose the location of the unknown node is (x_0, y_0) and it is able to obtain the distance estimates d_i' to the i-th reference node locating at (x_i, y_i), $1 \leqslant i \leqslant n$. Let d_i be the actual Euclidean distance to the i-th reference node, i.e.,

$$d_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}.$$

The difference between the measured and the actual distances can be represented by $\rho_i = d_i' - d_i$. Owing to ranging noises in d_i', ρ_i is often non-zero in practice.

The least squares method is used to assign a value to (x_0, y_0) that minimizes $\sum_{i=1}^{n} \rho_i^2$. This problem can be solved by a numerical solution to an over-determined linear system[25].
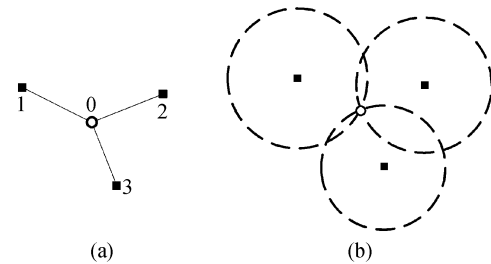

Fig.3. Trilateration. (a) Measuring distance to 3 reference nodes. (b) Ranging circles.

The over-determined linear system can be obtained as follows. Rearranging and squaring terms of the actual distances, we have n such equations:

$$x_i^2 + y_i^2 - d_i^2 = 2x_0 x_i + 2y_0 y_i - (x_0^2 + y_0^2).$$

By subtracting out the n-th equation from the rest, we have $n-1$ equations of the following form:

$$x_i^2 + y_i^2 - x_n^2 - y_n^2 - d_i^2 + d_n^2 = 2(x_i - x_n)x_0 + 2(y_i - y_n)y_0$$

which yields the linear relationship

$$\boldsymbol{Ax} = \boldsymbol{B}$$

where $\boldsymbol{A}$ is an $(n-1) \times 2$ matrix, such that the i-th row of $\boldsymbol{A}$ is $[2(x_i - x_n) \ 2(y_i - y_n)]$, $\boldsymbol{x}$ is the column vector representing the coordinates of the unknown location $[x_0 \ y_0]^{\mathrm{T}}$, and $\boldsymbol{B}$ is the $(n-1)$ element column vector whose i-th term is $(x_i^2 + y_i^2 - x_n^2 - y_n^2 - d_i^2 + d_n^2)$. Practically, we cannot determine $\boldsymbol{B}$, since the real distances are not known to us, so computation is performed on $\boldsymbol{B}'$, which is the same as $\boldsymbol{B}$ with d_i' substituting for d_i. Now the least square solution is an estimate for $\boldsymbol{x}'$ that minimizes $\|\boldsymbol{Ax}' - \boldsymbol{B}'\|^2$, which is provided by $\boldsymbol{x}' = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{B}'$.

So far, for distance-based positioning, the only thing omitted is how to measure distances in the physical world. Many ranging techniques are proposed and developed; among them, the radio signal strength based and time based ranging are two of the most widely used ones in existing designs.

(a) *Radio Signal Strength Based Distance Measurement*

Radio Signal Strength (RSS) based ranging techniques are based on the fact that the strength of radio signal diminishes during propagation. As a result, the understanding of radio attenuation helps to map the signal strength to the physical distance. In theory, radio signal strengths diminish with distance according to

a power law. A generally employed model for wireless radio propagation is as follows[26]:

$$P(d) = P(d_0) - \eta 10 \log\left(\frac{d}{d_0}\right) + X_\sigma$$

where $P(d)$ is the received power at distance d, $P(d_0)$ the received power at some reference distance d_0, η the path-loss exponent, and X_σ a log-normal random variable with variance σ^2 that accounts for fading effects. Hence, if the path-loss exponent for a given environment is known, the received signal strength can be translated to the signal propagation distance.

In practice, however, RSS-based ranging measurements contain noises on the order of several meters. The ranging noise occurs because radio propagation tends to be highly dynamic in complicated environments.

On the whole, RSS based ranging is a relatively "cheap" solution without any extra devices, as all network nodes are supposed to have radios. It is believed that more careful physical analysis of radio propagation may allow better use of RSS data. Nevertheless, the breakthrough technology is not there today.

(b) *Time Difference of Arrival (TDoA)*

A more promising technique is the combined use of ultrasound/acoustic and radio signals to estimate distances by determining the Time Difference of Arrival (TDoA) of these signals[25,27-28]. In such a scheme, each node is equipped with a speaker and a microphone, as illustrated in Fig.4. Some systems use ultrasound while others use audible frequencies. The general ranging technique, however, is independent of particular hardware.
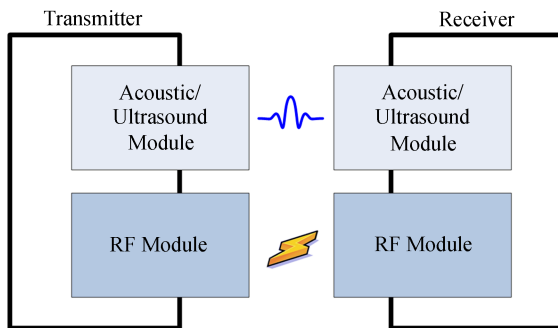

Fig.4. TDoA hardware model.

The idea of TDoA ranging is conceptually simple, as illustrated in Fig.5. The transmitter first sends a radio signal. It waits for some fixed internal of time, t_{delay} (which might be zero), and then produces a fixed pattern of "chirps" on its speaker. When receivers hear the radio signal, they record the current time, t_{radio}, and then turn on their microphones. When their

microphones detect the chirp patter, they again record the current time, t_{sound}. Once they have t_{radio}, t_{sound}, and t_{delay}, the receivers can compute the distance d to the transmitter by

$$d = \frac{v_{\text{radio}} \cdot v_{\text{sound}}}{v_{\text{radio}} - v_{\text{sound}}} \cdot (t_{\text{sound}} - t_{\text{radio}} - t_{\text{delay}}),$$

where v_{radio} and v_{sound} denote the speeds of radio and sound waves respectively. Since radio waves travel substantially faster than sound in air, the distance can be simply estimated as $d = v_{\text{sound}} \cdot (t_{\text{sound}} - t_{\text{radio}} - t_{\text{delay}})$. If the design is transmitting radio and acoustic signals simultaneously, i.e., $t_{\text{delay}} = 0$, the estimation can be further simplified as $v_{\text{sound}} \cdot (t_{\text{radio}} - t_{\text{sound}})$.
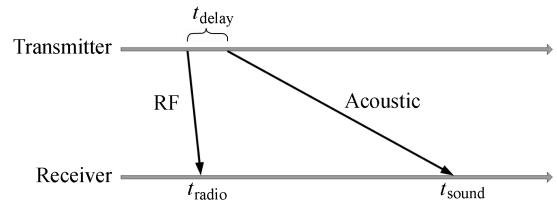

Fig.5. TDoA computation model.

TDoA methods are impressively accurate under line-of-sight conditions. For instance, it is claimed in [25] that distance can be estimated with error no more than a few centimeters for node separations under 3 meters. The cricket ultrasound system[27] can obtain close to centimeter accuracy without calibration over ranges of up to 10 meters in indoor environments.

Being accurate, TDoA systems suffer from high cost and are constrained by the line-of-sight condition, which can be difficult to meet in some environments. In addition, TDoA systems perform better when they are calibrated properly, since speakers and microphones never have identical transmission and reception characteristics. Furthermore, the speed of sound in air varies with air temperature and humidity, which introduce inaccuracy into distance estimation. Acoustic signals also show multi-path propagation effects that may affect the accuracy of signal detection. These can be mitigated to a large extent using simple spread-spectrum techniques, like those described in [29]. The basic idea is to send a pseudo-random noise sequence as the acoustic signal and use a matched filter for detection, instead of using a simple chirp and threshold detection.

Recently, researchers observe that two intrinsic uncertainties in TDoA measuring process can contribute to the ranging inaccuracy: the possible misalignment between the sender timestamp and the actual signal emission, and the possible delay of a sound signal arrival being recognized at the receiver[30]. Indeed, many factors can cause these uncertainties in a real system,

278

J. Comput. Sci. & Technol., Mar. 2010, Vol.25, No.2

such as the lack of real-time control, software delay, interrupt handling delay, system loads, etc. These two delays, if not carefully controlled, can easily add up to several milliseconds on average, which translates to several feet of ranging error. BeepBeep[30], a recently designed high-accuracy acoustic-based ranging system, achieves the localization accuracy as good as one or two centimeters within a range of more than ten meters, which is so far the best ranging result for off-the-shelf cell phones.

In conclusion, many localization algorithms use TDoA simply because it is dramatically more accurate than radio-only methods. The tradeoff is that nodes must be equipped with acoustic transceivers in addition to radio transceivers, significantly increasing both the complexity and the cost of the system.

2.1.2 Angle Measurement

Another approach for localization is the use of angular estimates instead of distance estimates. In trigonometry and geometry, triangulation is the process of determining the location of a point by measuring angles to it from two known reference points at either end of a fixed baseline, using the law of sines. Triangulation was once used to find the coordinates and sometimes the distance from a ship to the shore.

The Angle of Arrival (AoA) data is typically gathered using radio or microphone arrays, which allow a receiver to determine the direction of a transmitter. Suppose several (3~4) spatially separated microphones hear a single transmitted signal. By analyzing the phase or time difference between the signal arrivals at different microphones, it is possible to discover the AoA of the signal.

Those methods can obtain accuracy within a few degrees[31]. A straightforward localization technique, involving three rotating reference beacons at the boundary of a sensor network providing localization for all interior nodes, is described in [32]. A more detailed description of AoA-based triangulation techniques is provided in [33].

Unfortunately, AoA hardware tends to be bulkier and more expensive than TDoA ranging hardware, since each node must have one speaker and several microphones. Furthermore, the need of spatial separation between microphones is difficult to be accommodated in small size sensor nodes.

2.1.3 Area Measurement

If the radio or other signal coverage region can be described by a geometric shape, locations can be estimated by determining which geometric areas that a node is in. The basic idea of area estimation is to compute the intersection of all overlapping coverage regions and choose the centroid as the location estimate. Along with the increasing number of constraining areas, higher localization accuracy can be achieved.

According to how area is estimated, we classify the existing approaches into two categories: single reference area estimation and multi-reference area estimation.

(a) *Single Reference Area Estimation*

In this case, constraining areas are obtained according to a single reference. For instance, the region of radio coverage may be upper-bounded by a circle of radius R_{max}. In other words, if node B hears node A, it knows that it must be no more than a distance R_{max} from A. If an unknown node hears from several reference nodes, it can determine that it must lie in the geometric region described by the intersection of circles of radius R_{max} centered at these nodes, as illustrated in Fig.6(a). This can be extended to other scenarios. For instance, when both lower bound R_{min} and upper bound R_{max} can be determined, the shape of a single node's coverage is an annulus, as illustrated in Fig.6(c); when an angular sector $(\theta_{min}, \theta_{max})$ and a maximum range R_{max} can be determined for some radio antennas, the shape for a single node's coverage would be a cone with given angle and radius, illustrated in Fig.6(d).
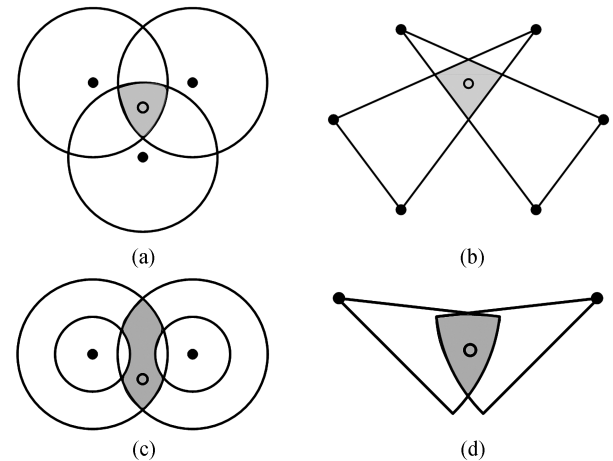

Fig.6. Area measurements.

Localization techniques using geometric regions are first described by [34]. One of the nice features of these techniques is that not only can the unknown nodes use the centroid of the overlapping region as a specific location estimate if necessary, but also they can determine a bound on the location error using the size of this region. When the upper bounds on these regions are tight, the accuracy of this geometric approach can be further enhanced by incorporating "negative information" about which reference nodes are not within range[35]. Although arbitrary shapes can be potentially

computed in this manner, a computational simplification to determine this bounded region is to use rectangular bounding boxes as location estimates. The bounding-box algorithm is a computationally efficient method to localize nodes given their ranges to several references. Essentially, it is assumed that each node lies within the intersection of its reference bounding boxes.

(b) *Multi-Reference Area Estimation*

Another approach of area estimation is the approximate point in triangle (APIT)[36]. Its novelty lies in how the regions are defined. Actually, bounding triangles are obtained according to any group of three reference nodes, rather than the coverage of a single node.

APIT consists of two key processes: triangle intersection and point in triangle (PIT) test. Nodes are assumed to hear a fairly large number of beacons. A node forms some number of "reference triangles": The triangle formed by three arbitrary references. The node then decides whether it is inside or outside a given triangle by PIT test. Once this process is complete, the node simply finds the intersection of the reference triangles that contains it and chooses the centroid as its position estimate, as illustrated in Fig.6(b). In this process, APIT does not assume that nodes can really range to these beacons.

The PIT test is based on geometry. For a given triangle with points A, B, and C, a point M is outside triangle ABC, if there exists a direction such that a point adjacent to M is further/closer to points A, B, and C simultaneously. Otherwise, M is inside triangle ABC. Unfortunately, given that typically nodes cannot move, an approximate PIT test is proposed based on two assumptions. The first one is that the range measurements are monotonic and calibrated to be comparable but are not required to produce distance estimates. The second one assumes sufficient node density for approximating node movement. If no neighbor of M is further from/closer to all three anchors A, B, and C simultaneously, M assumes that it is inside triangle ABC. Otherwise, M assumes it resides outside this triangle. In practice, however, this approximation does not realize the PIT test well. Nevertheless, APIT provides a novel point of view to conduct localization based on area estimation.

2.1.4 Hop Count Measurements

Based on the observation that if two nodes can communicate by radio, their distance from each other is less than R (the maximum range of their radios) with high probability, many delicate approaches are designed for accurate localization. In particular, researchers have found "hop count" to be a useful way to compute inter-node distances. The local connectivity information provided by the radio defines an unweighted graph, where the vertices are wireless nodes and edges represent direct radio links between nodes. The hop count h_{ij} between nodes s_i and s_j is then defined as the length of the shortest path from s_i to s_j. Obviously, the physical distance between s_i and s_j, namely, d_{ij}, is less than $R \times h_{ij}$, the value which can be used as an estimate of d_{ij} if nodes are densely deployed.

It turns out that a better estimate can be made if we know n_{local}, the expected number of neighbors per node. As shown by Kleinrock and Silvester[37], it is possible to compute a better estimate for the distance covered by one radio hop:

$$d_{\text{hop}} = R\Big(1 + e^{-n_{\text{local}}} - \int_{-1}^{1} e^{-(n_{\text{local}}/\pi)\arccos t - t\sqrt{1-t^2}}\mathrm{d}t\Big).$$

Then $d_{ij} \approx h_{ij} \times d_{\text{hop}}$. Experimental studies[38] show that the equation above can be quite accurate when n_{local} grows above 5. However, when $n_{\text{local}} > 15$, d_{hop} approaches R, so the equation of d_{hop} becomes less useful. Nagpal *et al.*[38] demonstrate by algorithm that even better hop-count distance estimates can be computed by averaging distances with neighbors. This benefit does not appear until $n_{\text{local}} \geqslant 15$; while, it can reduce hop-count error down to as little as $0.2R$.

Another method to estimate per-hop distance is to employ a number of reference nodes, as illustrated in Fig.7(a). Since the locations of reference nodes are known, the pairwise distances among them can be computed. Hence, if the hop count h_{ij} between two references (s_i and s_j) and the distance d_{ij} are available, the per-hop distance can be estimated by $d_{\text{hop}} = d_{ij}/h_{ij}$.

Due to the hardware limitations and energy constraints of wireless devices, hop count based localization approaches are cost-effective alternatives to ranging based approaches. Since there is no way to measure physical distances between nodes, existing hop-count based approaches largely depend on a high density of seeds.

Most existing approaches, however, would fail in anisotropic network topologies, where holes exist among wireless devices, as shown in Fig.7(b). In anisotropic networks[39], the Euclidean distance between a pair of nodes may not correlate closely with the hop count between them because the corresponding shortest path may have to curve around intermediate holes, leading to poor distance estimation. Unfortunately, anisotropic networks are more likely to exist in practice for several reasons. First, in many real applications, sensor nodes/seeds can rarely be uniformly deployed over the field due to the geographical obstacles. Second, even if we assume that the initial sensor

280

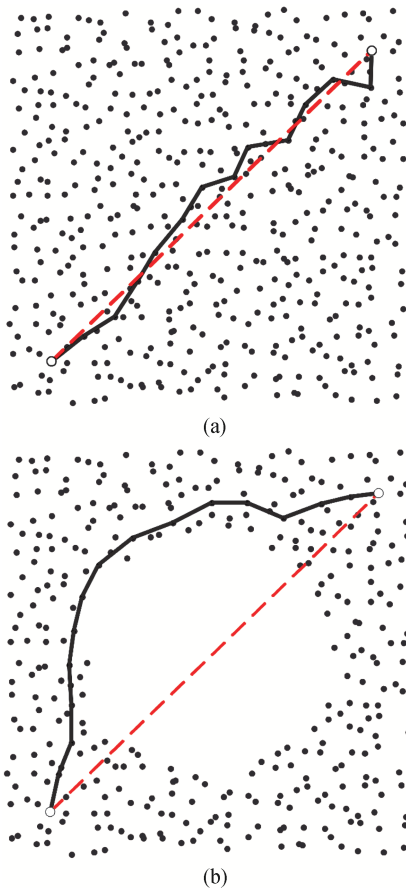*J. Comput. Sci. & Technol., Mar. 2010, Vol.25, No.2*

Fig.7. Hop count measurement. (a) Per-hop distance measurement. (b) Distance mismatch.

network is isotropic, unbalanced power consumption among nodes will easily create holes in the network. Recently, a distributed method[40] has been proposed to detect hole boundary by using only the connectivity information. Based on that work, REP[41] is proposed to deal with the "distance mismatch" problem in anisotropic networks.

2.1.5 Neighborhood Measurement

The radio connectivity measurement can be considered economic since no extra hardware is needed. Perhaps the most basic positioning technique is that of one neighbor proximity, involving a simple decision of whether two nodes are within the reception range of each other. A set of reference nodes is placed in the network with some non-overlapping (or nearly non-overlapping) sub-regions. Reference nodes periodically emit beacons including their location IDs. Unknown nodes use the received locations as their own location, achieving a course-grained localization. The major advantage of such a neighbor proximity approach is the simplicity of computation.

The neighborhood information can be more useful

when the density of reference nodes is sufficiently high so that there are often multiple reference nodes within the range of an unknown node[42]. Let there be k reference nodes within the proximity of the unknown node. As shown in Fig.8, we use the centroid of the polygon constructed by the k reference nodes as the estimated position of the unknown node. This is actually a k-nearest-neighbor approximation in which all reference nodes have equal weights.
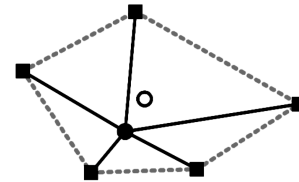

Fig.8. k-neighbor proximity.

This centroid technique has been investigated using a model with each node having a simple circular range R in an infinite square mesh of reference nodes spaced a distance d apart[43]. It is shown through simulation that, as the overlap ratio R/d is increased from 1 to 4, the average error in localization decreases from $0.5d$ to $0.25d$.

The k-neighbor proximity approach inherits the merit of computational simplicity from the single neighbor proximity approach; while at the same time, it provides more accurate localization results statistically.

2.1.6 Comparative Study and Directions of Future Research

A comparative study is presented in this subsection for existing physical measurement approaches. Table 1 provides an overview of these approaches in terms of accuracy, hardware cost, and environment requirements. All approaches have their own merits and drawbacks, making them suitable for different scenarios.

Recent technical advances foster two novel ranging

Table 1. Comparative Study of Physical Measurements

Physical Measurements		Accuracy	Hardware Cost	Computation Cost
Distance	RSS	Median	Low	Low
	TDoA	High	High	Low
Angle	AoA	High	High	Low
Area	Single reference	Median*	Median*	Median
	Multi-reference	Median*	Median*	High
Hop Count	Per-hop distance	Median	Low	Median
Neighborhood	Single neighbor	Low	Low	Low
	Multi-neighbor	Low	Low	Low

∗: depends on the diverse geometric constrains

approaches. Ultra-WideBand (UWB) is a radio technology that can be used at very low energy levels for short-range high-bandwidth communications by using a large portion of the radio spectrum[44]. It has relative bandwidth larger than 20% or absolute bandwidth of more than 500 MHz. Such wide bandwidth offers a wealth of advantages for both communications and ranging applications. In particular, a large absolute bandwidth offers high resolution with improved ranging accuracy of centimeter-level.

UWB has a combination of attractive properties for in-building location systems. First, it is a non-line-of-sight technology with a range of a few tens of meters, which makes it practical to cover large indoor areas; second, it is easy to filter the signal to minimize the multipath distortions that are the main cause of inaccuracy in RF based location systems. With conventional RF, reflections in in-building environments distort the direct path signal, making accurate pulse timing difficult; while with UWB, the direct path signal can be distinguished from the reflections. These properties provide a good cost-to-performance ratio of all available indoor location technologies.

The second promising technique is Chirp Spread Spectrum (CSS) designed by Nanotron Technologies[45] and adopted by IEEE 802.15.4a. CSS is a customized application of Multi-Dimensional Multiple Access (MDMA) for the requirements of battery-powered applications, where the reliability of the transmission and low power consumption are of special importance. CSS operates in the 2.45 GHz ISM band and achieves a maximum data rate of 2 Mbps. Each symbol is transmitted with a chirp pulse that has a bandwidth of 80 MHz and a fixed duration of 1 μs.

Nanotron Technologies have developed a ToA method that employs a ranging signal sent by a reader and an acknowledgement sent back from the tag to cancel out the requirements for clock synchronization. This solution provides protection against multi-path propagation and noise by its CSS modulation. To eliminate the effect of clock drift and offset, ranging measurements are taken by both the tag and the reader to provide two measurements that can then be averaged. This ranging result is reasonably accurate with no more than 1 meter error, even in the most challenging environments. The method is called Symmetric Double Sided Two Way Ranging, or SDS-TWR.

2.2 Network-Wide Localization

2.2.1 Computation Organization

This subsection defines taxonomy for localization algorithms based on their computational organization.

Centralized algorithms are designed to run on a central machine with powerful computational capabilities. Network nodes collect environmental data and send back to a base station for analysis, after which the computed positions are delivered back into the network. Centralized algorithms resolve the computational limitations of nodes. This benefit, however, comes from accepting the communication cost of transmitting data back to a base station. Unfortunately, communication generally consumes more energy than computation in existing network hardware platforms.

In contrast, distributed algorithms are designed to run in network, using massive parallelism and internode communication to compensate for the lack of centralized computing power, while at the same time to reduce the expensive node-to-sink communications. Distributed algorithms often use a subset of the data to locate each node independently, yielding an approximation of a corresponding centralized algorithm where all the data are considered and used to compute the positions of all nodes simultaneously. There are two important categories of distributed localization approaches. The first group, beacon-based distributed algorithms, typically starts a localization process with beacons and the nodes in vicinity of beacons. In general, nodes obtain distance measurements to a few beacons and then determine their locations. In some algorithms, the newly localized nodes can become beacons to help locating other nodes. In such iterative localization approaches, location information diffuses from beacons to the border of a network, which can be viewed as a top-down manner. The second group of approaches performs in a bottom-up manner, in which localization is originated in a local group of nodes in relative coordinates. After gradually merging such local maps, entire network localization is achieved in global coordinates.

2.2.2 Centralized Localization Approaches

(a) *Multi-Dimensional Scaling (MDS)*

Multi-Dimensional scaling (MDS)[46] was originally developed for use in mathematical psychology. The intuition behind MDS is straightforward. Suppose there are n points, suspended in a volume. We do not know the positions of the points, but we know the distances between each pair of points. MDS is an $O(n^3)$ algorithm that uses the law of cosines and linear algebra to reconstruct the relative positions of the points based on the pairwise distances. The algorithm has three stages:

1) Generate an $n \times n$ matrix $\boldsymbol{M}$, whose (i, j) entry contains the estimated distance between nodes i and j (simply run Floyd's all-pairs shortest-path algorithm).

2) Apply classical metric-MDS on $\boldsymbol{M}$ to determine a map that gives the locations of all nodes in relative

coordinates.

3) Transform the solution into global coordinates based on some number of fixed anchor nodes.

MDS performs well on RSS data, getting performance on the order of half the radio range when the neighborhood size n_{local} is higher than $12^{[47]}$. The main problem with MDS, however, is its poor asymptotic performance, which is $O(n^3)$ on account of stages 1 and 2.

(b) *SemiDefinite Programming* (*SDP*)

The semidefinite programming (SDP) approach was pioneered by Doherty *et al.*[34] In their algorithm, geometric constraints between nodes are represented as linear matrix inequalities (LMIs). Once all the constraints in the network are expressed in this form, the LMIs can be combined to form a single semidefinite program, which is solved to produce a bounding region for each node. The advantage of SDP is its elegance on concise problem formulation, clear model representation, and elegant mathematic solution.

Solving the linear or semidefinite program has to be done centrally. The relevant operation is $O(k^2)$ for angle of arrival data, and $O(k^3)$ when radial (e.g., hop count) data is included, where k is the number of convex constraints needed to describe the network. Thus, the computation complexity of SDP is likely to preclude itself in practice.

Unfortunately, not all geometric constraints can be expressed as LMIs. In general, only constraints that form convex regions are amenable to representation as an LMI. Thus, AoA data can be represented as a triangle and hop count data can be represented as a circle, but precise range data cannot be conveniently represented, as rings cannot be expressed as convex constrains. This inability to accommodate precise range data might prove to be a significant drawback.

2.2.3 Distributed Localization Approaches

(a) *Beacon Based Localization*

Beacon based localization approaches utilize estimates of distances to reference nodes that may be several hops away[48-49]. These distances are propagated from reference nodes to unknown nodes using a basic distance-vector technique. Such a mechanism can be seen as a top-down manner due to the progressive propagation of location information from beacons to an entire network. There are three types as follows.

1) DV-hop: In this approach, each unknown node determines its distance from various reference nodes by multiplying the least number of hops to the reference nodes with an estimated average distance per hop that depends upon the network density.

2) DV distance: If inter-node distance estimates are directly available for each link in the graph, the distance-vector algorithm is used to determine the distance corresponding to the shortest distance path between the unknown nodes and reference nodes.

3) Iterative localization: One variant of above approaches is indirect use of beacon nodes. Initially an unknown node, if possible, is located based on its neighbors by multilateration or other positioning techniques. After being aware of its location, it becomes a reference node to localize other unknown nodes in the subsequent localization process. This step continues iteratively, gradually turning the unknown nodes to the known. The process of iterative localization is illustrated in Fig.9.
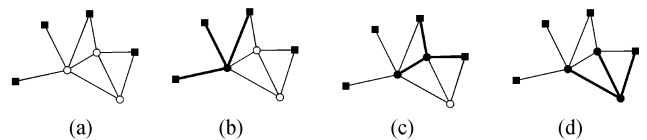

Fig.9. Iterative localization.

Iterative trilateration only involves local information (information within neighborhood) and accordingly reduces communication cost. Nevertheless, the use of localized unknown nodes as reference nodes inherently introduces substantial cumulative error. Some works characterize the error propagation in multihop localization approaches and make efforts to control error accumulation[50-51].

Experimental studies show that multilateration-based algorithms require an average node degree beyond 10 to properly localize most of the nodes in a randomly deployed network[52]. When the average degree is below 8, iterative multilateration will fail for most of the nodes, since it makes the nodes form a chain of dependence and a single-point failure on one node would lead to further failures on a set of subsequent nodes. To make localization applicable for sparse networks, Sweeps[53] partially relaxes the requirement of node dependence. In contrast to the traditional unique position computation, Sweeps introduces a novel concept of finite localization which locates a target node to a set of possible positions, called candidate positions. Finite localization guarantees that the ground truth position of a node is one of its candidate positions. Further, Sweeps adopts a new positioning scheme, called bilateration, to compute the candidate positions of a node by utilizing only two ranging measurements. As shown in Fig.10, bilateration produces two candidate positions for a node and one of them is the ground truth position. Similar to multilateration, the finitely localized node, called swept node, can act as a reference node to localize other nodes. The only difference is that all candidate positions of the swept node are enumerated for

the location computation of the target node. Moreover, after each bilateration, Sweeps checks the consistency among the candidate position sets and deletes those incompatible items. Under this mechanism, Sweeps can locate a large proposition of theoretically localizable nodes in a network. However, the worst case computation complexity of this design grows exponentially with the number of nodes.
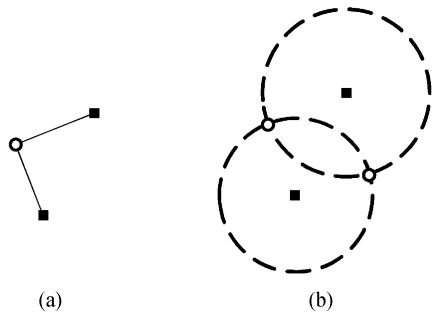

Fig.10. Bilateration. (a) Measuring distance to 2 reference nodes. (b) Bilateration creates two possible locations.

(b) *Coordinate System Stitching*

Coordinate system stitching is a different way to address the same problem. It has attracted a lot of research efforts recently[48,52,54]. It works in a bottom-up manner, in which localization is originated in a local group of nodes in relative coordinates. By gradually merging such local maps, it finally achieves entire network localization in global coordinates, as illustrated in Fig.11.
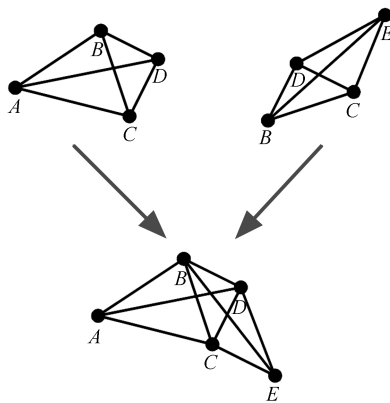

Fig.11. Coordinate system stitching.

Coordinate system stitching works as follows:

1) Split the network into small overlapping sub-regions. Very often each sub-region is simply a single node and its one-hop neighbors.

2) For each sub-region, compute a "local map", which is essentially an embedding of the nodes in the sub-region into a relative coordinate system.

3) Finally, merge sub-regions using a coordinate system registration procedure. Coordinate system registration finds a rigid transformation that maps points in one coordinate system to a different coordinate system. Thus, step 3) places all the sub-regions into a single global coordinate system. Many algorithms do this step suboptimally, since there is a closed-form, fast and least-square optimal method of registering coordinate system.
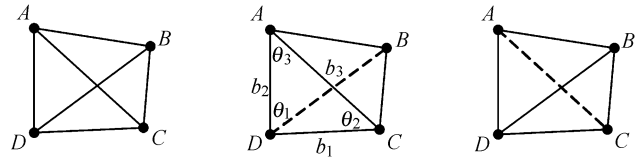

Fig.12. Robust quadrilateral.

Moore *et al.*[52] outline an approach that produces more robust local maps. Rather than using three arbitrary nodes, they use "robust quadrilateral" (robust quads) to define a map. As shown in Fig.12, a robust quad consists of four subtriangles (ΔABC, ΔADC, ΔABD, ΔBCD) that satisfy:

$$b \times \sin^2(\theta) > d_{\min},$$

where b is the length of the shortest side, θ is the smallest angle, and $d_{\min}$ is a predetermined constant based on average measurement error. The idea is that the points of a robust quad can be placed correctly with respect to each other (i.e., no "flips"[55]). Moore *et al.* demonstrate that the probability of a robust quadrilateral experiencing internal flips given zero mean Gaussian measurement error can be bounded by setting $d_{\min}$ appropriately. In effect, $d_{\min}$ filters out quads that have too much positional ambiguity to be localized with confidence. The appropriate level of filtering is based on the amount of uncertainty in distance measurements. Unfortunately, coordinate system stitching suffers from error propagation caused by local map stitching. Moore *et al.* prove the probability of their algorithm constructing correct local maps and prove error lower bound on the local map positions. Furthermore, these techniques have a tendency to orphan nodes, either because they could not be added to a local map or because their local map failed to overlap sufficiently with neighboring maps. Moore *et al.* argue that this is acceptable because the orphaned nodes are the nodes most likely to display high error. They point out that "for many applications, missing localization information for a known set of nodes is preferential to incorrect information for an unknown set". However, this answer may not be satisfactory for some applications, many of which cannot use nodes without locations for sensing, routing,

target tracking, or other tasks.

A more general form of coordinate system stitching is the component based localization[56]. A component is defined as a group of nodes that form a rigid structure. Using rigid components as basic units, the algorithm[56] merges and localizes components through inter-component distance measurements and anchor information.

As shown in Fig.13 three inter-component distance measurements constrain the relative geometric relationship between two components A and B, both of which are adjacent to two anchors. From the perspective of each single node, none of them has (at least) two neighboring anchors. In contrast, from the perspective of components, component A and component B can be merged into a bigger component, which is localizable by referring to the four anchors. Next, all nodes in the two components are localized. The component-based localization algorithms are applicable for sparse networks. Similar to Sweeps, this design cannot guarantee terminating in polynomial time either, which is a major drawback.
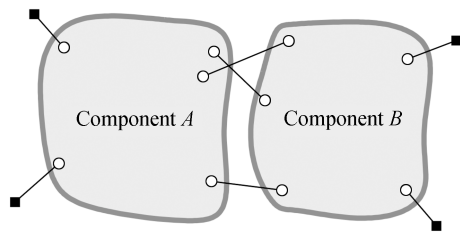

Fig.13. Component-based localization.

Coordinate system stitching techniques are quite compelling. They are inherently distributed, since sub-region and local map formation can trivially occur in the network and stitching is easily formulated as a peer-to-peer algorithm.

2.2.4 Comparative Study and Directions of Future Research

(a) Beacon Nodes

Beacon nodes (a.k.a. seeds or anchors) are necessary for localizing a network in the global coordinate system. Beacon nodes have no difference from ordinary network nodes except knowing their global locations as a priori. This knowledge can be hard-coded, or acquired through some extra hardware like a GPS receiver.

Beacon configuration has significant impact on localization. Existing work finds that higher localization accuracy can be achieved if beacons are placed in a convex hull around the network. Placing additional beacons in the center of the network is also helpful. Thus, it is necessary for system designers to plan the beacon layout before deploying a network.

(b) Node Density

Many localization algorithms are sensitive to node density. For instance, hop-count-based schemes generally require high node density so that the hop count approximation for distance is accurate. Similarly, algorithms that depend on beacon nodes fail when the beacon density is not sufficiently high in a specific region. Thus when designing or analyzing an algorithm, it is important to consider its requirement on node density, since high density may not be always true.

(c) Accuracy

Given a localization algorithm, location accuracy shows how well the computed locations match with the physical positions of the nodes. To be specific, location *accuracy* is defined as the expected Euclidean distance between the location estimate and the actual location of an unknown node, while location *precision* indicates the percentage of the results satisfying a pre-defined accuracy requirement.

For a given localization result, location accuracy trades off with location precision. If we relax the accuracy requirement, we can increase precision, and vice versa. Thus, we must put these two metrics in a common framework for comparison. We can fix location precision, say 95%, and evaluate the localization algorithms based on the corresponding accuracy achievements.

The error propagation demonstrates how location accuracy varies with the increase of measurement error. Intuitively, localization error is linear with measurement error. However, it is not true for many localization systems, especially for those sequential localization algorithms, such as trilateration and bilateration. Nodes with large location errors would contaminate their neighbors' estimates. In this scenario, measurement error is no longer the only factor contributing to localization error.

(d) Cost

In general, the cost of a localization system includes hardware cost and energy cost. Hardware cost consists of three parts: node density, beacon density, and measurement equipment. Usually, expensive equipments provide more accurate measurements. A localization procedure often involves inter-node measurement, computation and communication, among which communication consumes most energy. This is why distributed algorithms are often more compelling than centralized algorithms.

After years of extensive study on this topic, many localization solutions are presented. Table 2 presents an overview of typical approaches in terms of accuracy, node density, beacon percentage, computation cost, communication cost, and error propagation.

Table 2. Comparative Study of Localization Algorithms

Localization Algorithm		Accuracy	Node Density	Beacon Percentage	Computation Cost	Communication Cost	Error Propagation
Centralized	MDS	High	Low	Low	High	High	Low
	SDP	High	High	Median	High	High	Low
Distributed	Beacon based	Low	High	High	Low	Low	High*
	Coordinate stitching	Low	High	Low	Median	Median	High

∗: in case of iterative localization

In conclusion, a number of typical localization approaches are surveyed and evaluated with various metrics in this section. All approaches have their own merits and drawbacks, making them suitable for different applications. Hence, the design of a localization algorithm should sufficiently investigate application properties, as well as take into account algorithm generality and flexibility. In present and foreseeable future study, obtaining a Pareto improvement is a major challenge. That is, increasing the performance of one of the metrics without degradation on others.

In all localization algorithms discussed above, nodes should participate actively during a localization process, i.e., sending or receiving radio signals, or measuring physical data. For some applications, however, the to-be-locate objects cannot participate in localization and it is also difficult to attach networked nodes to them. One typical application is intrusion detection, in which it is impossible and unreasonable to equip intruders with locating devices. To tackle this issue, recently a novel concept of Device-Free Localization, also called Transceiver-Free Localization, is proposed[57-58]. Device-free localization is envisioned to be able to detect, localize, track, and identify entities free of devices, and works by processing the environment changes collected at scattering monitoring points. Existing work focuses on analyzing RSSI changes, and often suffers from high false positives. How to design a device-free localization system which can provide accurate locations is a challenging and promising research problem.

2.3 Error Control for Network Localization

2.3.1 Noisy Distance Measurement

Many localization algorithms are range-based and adopt distance ranging techniques, in which measuring errors are inevitable. Generally, these errors can be classified into two categories: *extrinsic* and *intrinsic*. The extrinsic error is attributed to the physical effects on the measurement channel, such as the presence of obstacles, multipath and shadowing effects, and the variability of the signal propagation speed due to environmental dynamics. On the other hand, the intrinsic error is caused by limitations of hardware and software. While the extrinsic one is more unpredictable and challenging during real deployments, the intrinsic one causes many complications when using multihop measurements to estimate node locations. Results from field experiments demonstrated that even relatively small ranging errors can significantly amplify the error of location estimates[52]; thus, dealing with such errors is an essential issue for high-accuracy localization algorithms.

(a) *Errors in Distance Measurements*

Table 3 lists the typical measuring (intrinsic) error of a range of nowadays ranging techniques: TDoA, RSS in AHLoS[25], Ultra Wideband system[59], RF Time of Flight ranging systems[60], and Elapsed Time between the two Time of Arrival (EToA) in BeepBeep[30]. In general, RF-based techniques, e.g., RSS, UWB and RF ToF, can achieve the meter-level accuracy in a range of tens of meters. Time-related methods have more accurate results in the order of centimeters, but require extra hardware and energy consumption.

On the other hand, extrinsic errors are caused by environmental factors or unexpected hardware malfunction, leaving difficulties on characterizing them. We will briefly discuss the state-of-the-art works on controlling the intrinsic and extrinsic errors in the following subsections of location refinement and robust localization, respectively.

Table 3. Measurement Accuracy of Different Ranging Techniques

Technology	System	Accuracy	Range
TDoA	AHLoS	2 cm	3~10 m
RSS	AHLoS	2~4 m	30~100 m
UWB	PAL UWB	1.5 m	N/A
RF ToF	RF ToF	1~3 m	100 m
EToA	BeepBeep	1~2 cm	10 m

(b) *Negative Impact of Noisy Ranging Results*

Errors in distance ranging make localization more challenging in the following three aspects:

• *Uncertainty.* Fig.14 illustrates an example of trilateration under noisy ranging measurements. Trilateration often meets the situation that the three circles do not intersect at a common point. In other words,
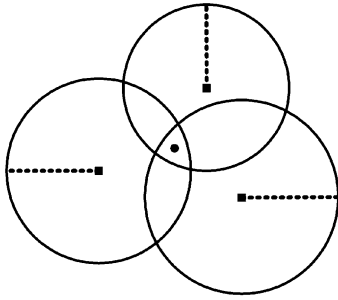
Fig.14. Trilateration under noisy ranging measurements.

there does not exist any position satisfying all distance constraints.

• *Non-Consistency.* In many cases, a single node has many reference neighbors. Any subgroup of them (no less than three) can locate this node by multilateration. The computed result, however, is varying if different groups of references are chosen, resulting in non-consistency. Thus, when alternative references are available, it is a difficult task to determine which combination of references provides the best results.

• *Error Propagation.* The results of a multihop localization process are based on a series of single hop multilaterations in an iterative manner[25]. In such a process, errors, coming from each step of multilateration, propagate and accumulate[50-51].

2.3.2 Error Characteristics of Localization

Localization error is a function of a wide range of network configuration parameters, including the numbers of beacons, the density of node deployment, network topology, etc., which constitute a complicated system. Understanding the error characteristics of localization is one essential step towards controlling errors. The Cramer Rao Lower Bound (CRLB)[61] provides a means for computing a lower bound on the covariance of any unbiased location estimate that uses distance measurements. In addition, CRLB can serve as a benchmark for localization algorithms: if the bound is closely achieved, there is little gain to continue working on improving the algorithm accuracy. Furthermore, the dependence of CRLB on network parameters helps to understand the error characteristics of network localization.

(a) *What is CRLB*

The Cramer Rao Lower Bound (CRLB) is a classic result from statistics that gives a lower bound on the error covariance for an unbiased estimate of parameter[61]. This bound provides a useful guideline to evaluate various estimators. One important and surprising advantage of CRLB is that we can calculate the lower bound without ever considering any particular estimation method. The only thing needed is

the statistical model of the random observations, i.e., $f(X|\theta)$, where X is the random observation, and θ is the parameter to be estimated. Any unbiased estimator $\hat{\theta}$ must satisfy

$$Cov(\hat{\theta}) \geqslant \{-E[\nabla_\theta(\nabla_\theta \ln f(X|\theta))^{\mathrm{T}}]\}^{-1} \qquad (1)$$

where $Cov(\hat{\theta})$ is the error covariance of the estimator, $E[\cdot]$ indicates expected value, and ∇_θ is the gradient operator with respect to θ.

The CRLB is limited to unbiased estimators that provide estimates equal to the ground truth if averaged over enough realizations. In some cases, however, biased estimators can achieve both a variance and a mean squared error that are below the CRLB.

(b) *CRLB for Multihop Localization*

In network localization, the parameter vector θ of interest consists of the coordinates of nodes to be localized, given by $\theta = [x_1, y_1, x_2, y_2, \ldots, x_L, y_L]^{\mathrm{T}}$, where L is the number of nodes to be localized. The observation vector X is formed by stacking the distance measurements $\hat{d}_{ij}$. Let M denote the size of X. We assume the distance measurement are Gaussian[52,62], so the pdf of X is vector Gaussian. According to (1), we find that $CRLB = \{\frac{1}{\sigma^2}[G'(\theta)]^{\mathrm{T}}[G'(\theta)]\}^{-1}$, where σ^2 is the variance of each distance measurement error, and $G'(\theta)$ is the $M \times 2L$ matrix whose mn-th element is

$$G'(\theta)_{mn} = \begin{cases} \dfrac{x_i - x_j}{d_{ij}}, & \text{if } \theta_n = x_i, \\[2mm] \dfrac{x_j - x_i}{d_{ij}}, & \text{if } \theta_n = x_j, \\[2mm] \dfrac{y_i - y_j}{d_{ij}}, & \text{if } \theta_n = y_i, \\[2mm] \dfrac{y_j - y_i}{d_{ij}}, & \text{if } \theta_n = y_j, \\[2mm] 0, & \text{otherwise.} \end{cases} \qquad (2)$$

The above result on CRLB is with the assumption that the location information of beacons is exact. When the beacon nodes have location uncertainty, we can also characterize localization accuracy using a covariance bound that is similar to CRLB. Both the two bounds are tight in the sense that localization algorithms achieve these bounds in case of highly accurate ranging measurements. In addition, according to (2), the CRLB can be computed analytically and efficiently, avoiding the need for expensive Monte-Carlo simulations. The computational efficiency facilitates to study localization performance of large-scale networks.

(c) *CRLB for One-Hop Localization*

One-hop multilateration is the source of the location error that could be amplified by the iterative fashion of

network localization. The CRLB for multilateration exactly demonstrates how measurement errors and node geometry affect location accuracy.

Consider the one-hop localization problem: there are m reference nodes $v_1, v_2, \ldots, v_m$ and one node v_0 to be localized. From (1) and (2), we obtain

$$\sigma_0^2 = \sigma^2 m \Big[\sum_{i=1}^{m-1} \sum_{j>i}^{m} \sin^2 \alpha_{ij} \Big]^{-1} \qquad (3)$$

where σ_0^2 is the variance of the estimate location of v_0, α_{ij} is the angle between each pair of reference nodes (i, j). According to (3), the uncertainty of location estimate consists of two parts: the ranging error (σ^2) and the geometric relationship of references and to-be-localized nodes (α_{ij}). Eliminating the impact of ranging errors, the error amplification effect caused by the node geometry has been demonstrated as the *Geographic Dilution of Precision* (GDoP)[63], which is defined as σ_0/σ.

To gain more insights of GDoP, we consider a simplified case of multilateration, where the to-be-localized node v_0 is put at the center of a circle and $m = 3$ reference nodes v_1, v_2, v_3 lie on the circumference of that circle, setting all references the same distance to v_0, as shown in Fig.15(a). Fixing v_1 at $\beta_1 = 0$, GDoP becomes a function of the locations of v_2 and v_3, denoted
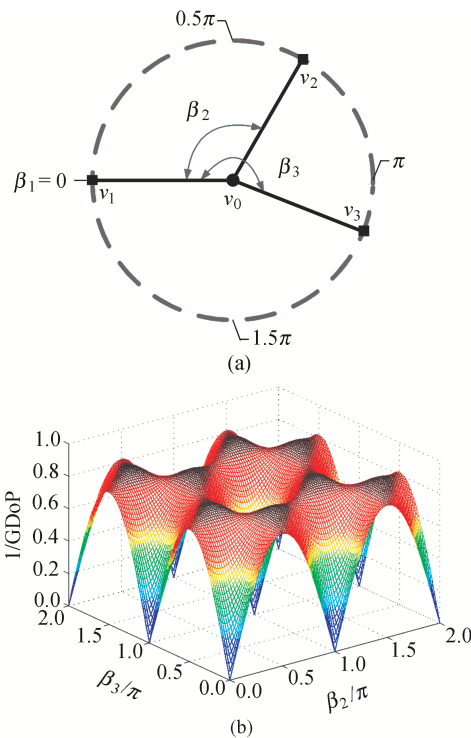

(a)

(b)

Fig.15. The impact of node geometry on the accuracy multilateration. (a) Design of experiments. (b) 3D plot of 1/GDoP.

by β_2, $\beta_3 \in [0, 2\pi]$ respectively. We plot the GDoP in Fig.15(b) and conclude that different geometric forms of multilateration provide different levels of localization accuracy. In particular, in this circular trilateration, the highest location accuracy would be achieved if reference nodes are evenly separated, namely, $\beta_1 = 0$, $\beta_2 = 2\pi/3$, and $\beta_3 = 4\pi/3$.

2.3.3 Location Refinement

Since localization is often conducted in a distributed and iterative manner, error propagation is considered as a serious problem, in which nodes with inaccurate location estimates contaminate the following localization process based on them.

(a) *Framework of Location Refinement*

To deal with error propagation, a number of location refinement algorithms have been proposed. In general, they are composed of three major components[62]:

• *Node Registry.* Each node maintains a registry which contains the node location estimate and the corresponding estimate confidence (uncertainty).

• *Reference Selection.* When redundant references are available, based on an algorithm-specified strategy, each node selects the reference combination achieving the highest estimate confidence (lowest uncertainty) to localize itself.

• *Registry Update.* In each iteration, if higher estimate confidence (lower uncertainty) is achieved, a node updates its registry and broadcasts this information to its neighbors.

Algorithm 1 outlines a framework of location refinement, in which how to select appropriate reference combinations is critical. Different strategies lead to different location refinement algorithms.

Algorithm 1. A Framework of Location Refinement

1:   Each node holds the tuple (p, e), where p is the node location estimate, e is the corresponding estimate confidence (uncertainty).

2:   Initialization step (optional):
    Each node computes an initialized location estimate.

3:   In each iteration, nodes update their registries.
    do
      for all to-be-localized node t **do**
      examine local neighborhood $N(t)$
      select the best reference combination and compute the estimate location $\hat{p}_t$ and confidence $\hat{e}_t$ decide whether to update the registry of t with the new tuple
    while the termination condition is not met.

(b) *Metrics for Location Refinement*

Although GDoP characterizes the effects of node geometry on location estimate, it cannot be directly

applied due to the need of the ground truth location of each node. This is a challenging issue and has attracted a lot of attentions.

Savarese *et al.*[64] introduce a confidence associated with each node's location, and weights multilateration results based on such confidence in the one-hop localization procedure. The estimate confidence is defined as follows. Beacons immediately start off with confidence 1; to-be-localized nodes begin with a low confidence and raise their confidences at subsequent refinement iterations. In each iteration, a node chooses those reference nodes that will raise its confidence to localize itself, and sets its confidence to the average of those references' confidences after a successful multilateration. This strategy is based on the intuition that the estimated locations of nodes close to beacons are more reliable, but puts littler emphasis on the effects of geometry on location estimate.

Through analyzing how ranging errors and reference location errors affecting localization, Liu *et al.*[62] design a location refinement scheme with error management. Each node maintains information (p, e), where p is the estimated location, and e is the corresponding estimate error, a metric reflecting the level of uncertainty. At the first beginning, each beacon is initialized with a registry ($beacon\_loc$, 0), and the to-be-localized nodes are initialized as ($unknown\_loc$, ∞). Instead of using the traditional Least Square (LS) solution $(A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}b$ mentioned previously, a robust LS (RLS) solution is adopted, $\hat{p} = (A^{\mathrm{T}}A + C_A)^{-1}A^{\mathrm{T}}b$, where $C_A = E[\Delta A^{\mathrm{T}} \cdot \Delta A]$ is the covariance matrix of perturbation of ΔA (the perturbation of A). Based on the RLS solution, the estimated location error caused by noisy distance measurements can be expressed by

$$E\|e_{\Delta b}\|^2 = E\|(A^{\mathrm{T}}A + C_A)^{-1}A^{\mathrm{T}}\Delta b\|^2 \qquad (4)$$

where Δb is the perturbation of b due to ranging errors. Similarly, the error due to location uncertainty of references is

$$E\|e_{\Delta a}\|^2 = E\|(A^{\mathrm{T}}A + C_A)^{-1}B\Delta a\|^2 \qquad (5)$$

where $a = (a_{11}, a_{21}, \ldots, a_{n1}, a_{12}, a_{22}, \ldots, a_{n2})^{\mathrm{T}}$, a vector rearranging elements in matrix A, Δa is the perturbation of a because of location uncertainty, and B is a matrix satisfying $A^{\mathrm{T}}b = Ba$. The total location error is the summation of these two terms, as they are assumed to be uncorrelated, i.e., $\hat{e} = E\|e_{\Delta b}\|^2 + \beta E\|e_{\Delta a}\|^2$, where β is a parameter to compensate for the over-estimation of the error due to a. Small value of β works well in practice[62].

By defining Quality of Trilateration (QoT)[55], the accuracy of trilateration can be characterized, enabling the comparison and selection among various geometric forms of trilateration. Assuming some probability distribution of ranging errors, probability tools are accordingly applied to quantify trilaterations. The large value of QoT indicates the estimate location is, with high probability, sufficiently close to the real location. Similar to [64], each node maintains a confidence associated with its location estimate. Let $t = Tri(s, \{s_i, i = 1, 2, 3\})$ denote a trilateration for s based on three reference nodes s_i and $Q(t)$ be the quality of t. The confidence of s (based on t) is computed according to the confidences of references $C(s_i)$:

$$C_t(s) = Q(t) \prod_{i=1}^{3} C(s_i). \qquad (6)$$

In each iteration, a to-be-localized node chooses the trilateration that achieves the highest confidence to localize itself. Different from the design in [64], which only takes the reference nodes reliability into account, QoT also considers the geometry of nodes when computing confidence. Compared to least square based approaches, QoT provides additional information that indicates how accurate a particular trilateration is. Such difference enables QoT the ability of distinguishing and avoiding poor trilaterations that are of much location uncertainty.

2.3.4 Robust Localization

Compared with intrinsic errors, extrinsic errors are more unpredictable and usually caused by non-systematic factors. Especially in some cases, the errors can be extremely large due to hardware malfunction or failure and adversary attacks. These severe errors can be seen as outliers of measurements that significantly deteriorate localization accuracy. Dealing with outliers recently becomes a hot research topic.

We classify existing outlier-resistant works into two major categories: explicitly sifting and implicitly de-emphasizing. The explicitly sifting methods are generally based on the intuition that normal ranging measurements are compatible while an outlier is likely to be inconsistent with other normal and outlier rangings. Through examining the inconsistency, we can identify and reject outlier measurements. In contrast, the implicitly de-emphasizing methods do not accept or reject a localization result by fixing a threshold; instead they employ robust statistics methods, such as high breakdown point estimators and influence functions, to mitigate the negative effects of outliers.

Based on the understanding that when there are redundant geometric constraints, there must be some inconsistency between outlier ranging results and normal

ranging results, Liu *et al.*[65] use the mean square error ζ^2 of the distance measurements based on the estimated locations as an indicator of the level of inconsistency, i.e.,

$$\zeta^2 = \frac{1}{m} \sum_{i=1}^{m} (\delta_i - \|\bar{p}_0 - p_i\|)^2 \qquad (7)$$

where m is the number of location references $\{(p_i, \delta_i), i = 1, 2, \ldots, m\}$ and $\bar{p}_0$ is the estimated location. It first estimates the location with the LS-based method, and then assesses whether the estimated location could be derived from a set of consistent location references based on ζ^2. If positive, it accepts the estimation result; otherwise, it identifies and removes the most inconsistent location reference and repeats the above process.

According to robust statistics, the least squares algorithm is sensitive to outliers, since its breakdown point is zero. One of the most commonly used high breakdown point (close to 50%) statistics algorithms is the method of least median of squares (LMS), which is adopted in [66]. It estimates the location using

$$\bar{p}_0 = \arg \min_{p_0} med_i(\delta_i - \|p_0 - p_i\|)^2. \qquad (8)$$

LMS-based localization first randomly draws M subsets of size m from the available location references (M and m are specified by application requirements), and then computes the estimated location and the residue corresponding to each subset. The final estimated location is selected according to (8).

Also deriving inspiration from robust statistics, the recent work SISR[67] uses a residual shaping influence function to de-emphasize the "bad nodes" and "bad links" during the localization procedure. To overcome the non-robustness to outliers of LS-based methods, instead of optimizing the sum of squared residues (as illustrated in Fig.16(a)), i.e., $F = \sum_{i,j} r(i,j)^2$, where $r(i,j)$ is the residue, SISR solves the optimization problem of $F = \sum_{i,j} s(i,j)$, where

$$s(i,j) = \begin{cases} \alpha r(i,j)^2, & \text{if } |r(i,j)| < \tau; \\ \ln(|r(i,j)| - u) - v, & \text{otherwise,} \end{cases} \qquad (9)$$

and $u = \tau - \frac{1}{2\alpha\tau}$, $v = \ln\left(\frac{1}{2\alpha\tau}\right) - \alpha\tau^2$, α and τ are parameters to be configured to control the shape of the influence function. Fig.17 plots a comparison between the standard squared residual used in conventional least squares and the shaped residual defined by (9). We can see that the shaped residual function is shaped like a "U" near 0 and like "wings" for value greater than τ, the reason why SISR can discount outliers from the localization procedure (as illustrated in Fig.16(b)).

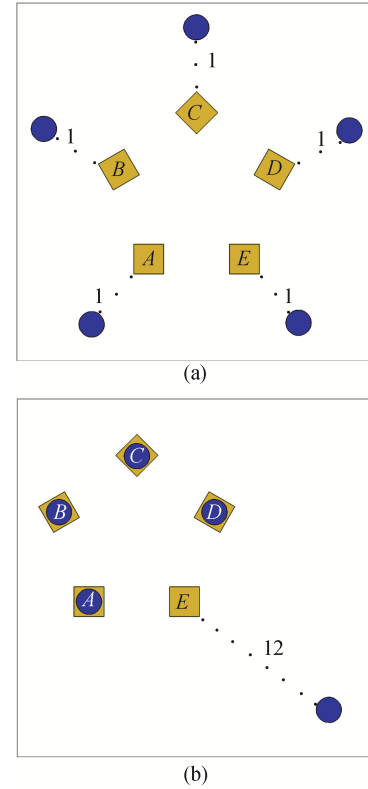Although many noise-resistant solutions have been

Fig.16. Two possible localization solutions of nodes A, B, C, D, and E, where E has large measurement errors. Squares indicate the ground truth location, and circles the computed localization solution. (a) Even; the measurement error from E is amortized over A, B, C, and D. (b) Uneven; solutions for A, B, C, and D are accurate, but that for E is very inaccurate.
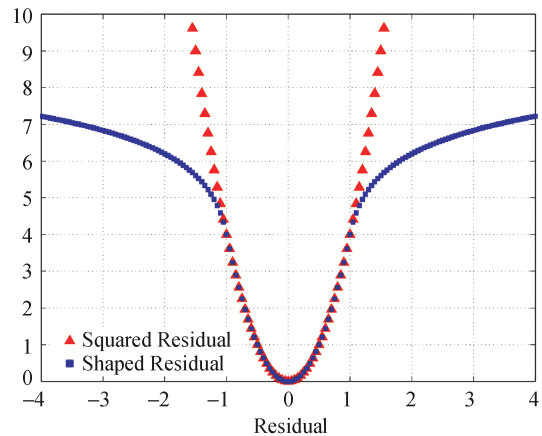

Fig.17. Comparison between the standard squared residual used in conventional least squares and the shaped residuals used in SISR with $\alpha = 4$ and $\tau = 1$.

proposed, they often emphasize on dealing with noisy distance rangings, ignoring the use of multimodal measurements. In a design based on multimodal measurements, the measuring data of distance, angle, and/or

290

J. Comput. Sci. & Technol., Mar. 2010, Vol.25, No.2

time difference are used simultaneously, providing more robust localization results. Recently, some progress from computational geometry suggests an efficient algorithm that can realize graphs with provable error bounds based on distance and angle information. This result reveals the great potential of multimodal measurements to control localization error, although its performance in practical systems is still unknown. With the rapid development of integrated circuits, multimodal measurement has been available on many wireless devices, especially sensor motes. Designing robust algorithms based multimodal measurements is another promising research issue.

Another direction of future research is detecting and sifting outlier ranging results. Typically, the quadrilateral structure is proposed for outlier detection[52]. But such a method is infeasible for sparse or moderate networks. What is the theoretical foundation of quadrilateral-based methods? Can we find any other kind of structure being able to detect outliers? How to sufficiently exploit redundant information from distance constraints[68]? To the best of our knowledge, all these issues are not well-studied.

3 Localizability

3.1 Network Localizability

Based on distance ranging techniques, the ground truth of a wireless ad-hoc network can be modeled by a distance graph $G = (V, E)$, where V is the set of wireless communication devices (e.g., laptops, RFID tags, or sensor nodes) and there is an un-weighted edge $(i, j) \in E$ if the distance between vertices i and j can be measured or both of them are at known locations, e.g., beacon nodes. Associated with each edge (i, j), we use a function $d(i, j)$: $E \to R$ to denote the measured distance value between i and j.

An essential question occurs as to whether or not a network is localizable given its distance graph. This is called the *network localizability*. A graph $G = (V, E)$ with possible additional constraints I (such as the known locations of beacon nodes) is *localizable* if there is a unique location $p(v)$ of every node v such that $d(i, j) = \|p(i) - p(j)\|$ for all links (i, j) in E and the constraint I is preserved, where $\|\cdot\|$ denotes the Euclidean distance in the 2D plane. Different from localization that determines locations of wireless nodes, localizability focuses on the location-uniqueness of a network.

On one hand, localizability assists localization fundamentally and importantly. As previously mentioned, localization often consumes a large amount of computational resource and makes sense only when networks are localizable. Hence, testing localizability before

localization can save unnecessary and meaningless computation, as well as accompanying power consumption.

On the other hand, being aware of localizability is of great benefit to many aspects of network operation and management, including topology control, network deployment, mobility control, power scheduling, and geographic routing, as shown in Fig.18. Taking deployment adjustment as an example, many measurements (e.g., augmenting communication range, increasing node or beacon density, etc.) can be taken to improve those non-localizable networks to be localizable, which can be effectively guided by the results of localizability testing.
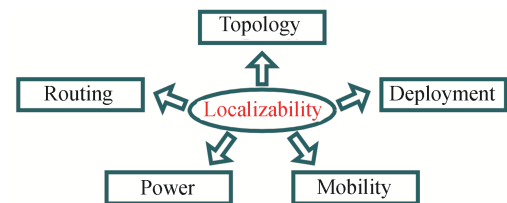

Fig.18. Localizability can assist network operation and management.

Although the network localizability is given birth by the proliferation of wireless ad-hoc/sensor networks, the problem of unique graph realization has attracted a lot of efforts made by researchers from different literatures over 30 years. An obvious requirement for a network to be localizable is the network connectivity: when a network is densely connected, it is more likely to be uniquely localizable; otherwise, it may fail the localizability testing. Besides dense connectivity, researchers also point out other requirements for localizable networks, which will be discussed in the next subsection.

3.2 Graph Rigidity

3.2.1 Globally Rigid Graphs

Previous studies have shown that the network localizability problem is closely related to graph rigidity[69-72].

A *realization* of a graph G is a function p that maps the vertices of G to the points in Euclidean space (this study assumes 2-dimension space). Generally, realizations are referred to the feasible ones that respect the pairwise distance constraints between a pair of vertices i and j if the edge $(i, j) \in E$. That is to say, $d(i, j) = \|(p(i) - p(j)\|$ for all $(i, j) \in E$. Two realization of G are equivalent if they are identical under trivial variation in 2D plane: translations, rotations, and reflections. A distance graph G has at least one feasible realization which represents the ground truth of the corresponding network. Formally, G is embeddable in 2D space and all pairwise distances are compatible,

i.e., satisfying the triangle inequality. We assume G is connected and has at least 4 vertices in the following analysis.

A graph is called *generically rigid* if one cannot continuously deform any of its realizations in the plane while preserving distance constraints[70,72]. A graph is *generically globally rigid* if it is uniquely realizable under translations, rotations, and reflections. A realization is said to be *generic* if the vertex coordinates are algebraically independent[70]. Since the set of generic realizations is dense in the space of all realizations, we omit this word for simplicity hereafter.

There are several distinct manners in which the non-uniqueness of realization can appear, as shown in Fig.19. A graph that can be continuously deformed while still satisfying all the constraints is said to be *flexible*; otherwise it is rigid. Hence, rigidity is a necessary condition for global rigidity. Rigid graphs, however, are still susceptible to discontinuous deformation. Specially, they may be subject to *flip* ambiguities in which a set of nodes have two possible configurations corresponding to a "reflection" across a set of mirror nodes (e.g., v and w in the flip example in Fig.19). This type of ambiguity is not possible in 3-connected graphs. Fig.19 further provides a 3-connected and rigid graph which becomes flexible upon removal of an edge. After the removal of the edge (u, v), a subgraph can swing into a different configuration in which the removed edge constraint is satisfied and then reinserted. Such a type of ambiguity, called *flex* deformation, is eliminated by *redundant rigidity*, the property that a graph remains rigid upon removal of any single edge.

Summarizing the conditions for eliminating ambiguities in graph realization, Jackson and Jordan provide the necessary and sufficient condition for global rigid-ity in the following theorem.

Theorem 1[71]. *A graph with $n \geqslant 4$ vertices is globally rigid in 2 dimensions if and only if it is 3-connected and redundantly rigid.*

Based on Theorem 1, the property of global rigidity can be tested in polynomial time by combining the Pebble game algorithm[73] and the network flow algorithms[70,74] for rigidity and 3-connectivity, respectively.

3.2.2 Conditions for Network Localizability

The locations of all vertices in a globally rigid graph can be uniquely determined if fixing any group of 3 vertices to avoid trivial variation in 2D plane, such as translation, rotation, or reflection. Hence, for wireless ad-hoc networks, Eren *et al.* present the following conclusion that perfectly bridges the theory of graph rigidity and the application of network localizability, as illustrated in Fig.20.

Theorem 2[69]. *A network is uniquely localizable if and only if its distance graph is globally rigid and it contains at least three anchors.*

Fig.21 shows the relationship between network connectivity and localizability (global rigidity) through extensive simulations. We generate networks of 400 nodes randomly, uniformly deployed in a unit square $[0, 1]^2$. The unit disk model with a radius is adopted for communication and distance ranging. For each evaluation, we integrate results from 100 network instances. The curve r_i denotes the percentage of i-connected networks in varied radius while r_g denotes globally rigid networks. Like many other properties for random geometric graphs, both connectivity and rigidity have transition phenomena. It can be seen that r_g lies between

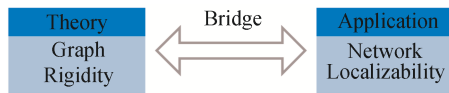| Deformation	Non-Unique Graph Realization		Solution	Resulting Graph
Continuous			G must be rigid	
Flip			G must be 3-connected.	
Flex			G must be redundantly rigid.	

Fig.19. Graph deformation and solutions.

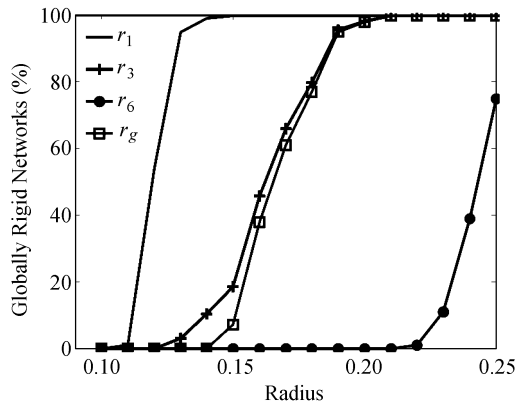Fig.20. Connection between theory and application.

Fig.21. The relationship between connectivity and rigidity.

r_3 and r_6 and is closer to r_3. This observation reflects the theoretical conclusion that 3-connectivity is a necessary condition while 6-connectivity is a sufficient one for global rigidity[75].

3.3 Inductive Construction of Globally Rigid Graphs

Inductive construction of globally rigid graphs inspires localizability testing in a distributed manner, which is highly appreciated by wireless ad-hoc/sensor networks since centralized approaches often consume large communication resource on data transmission and device synchronization.

3.3.1 Trilateration

Nowadays, trilateration is an important and well accepted scheme to inductively construct localizable networks. The basic principle of trilateration is that the position of an object can be uniquely determined by measuring the distances to three reference positions. It is widely-used in many real-world applications[25,27] as it is computationally efficient, fully distributed, and easy to implement. Importantly, the networks that can be constructed by iterative trilateration are localizable.

Theoretically, a *trilateration ordering* of a graph $G = (V, E)$ is an ordering $(v_1, v_2, \ldots, v_n)$ of V for which the first 3 vertices are pairwise connected and at least 3 edges connect each vertex v_j, $4 \leqslant j \leqslant n$, to the set of the first $j - 1$ vertices. A graph is a *trilateration extension* if it has a trilateration ordering. It is shown that trilateration extensions are globally rigid[69,76].

Trilateration based approaches, however, recognize only a subset (called trilateration extension) of globally

rigid graphs. In Fig.22(a), two globally rigid components are connected by nodes i ($i = 1, 2, \ldots, 7$). Suppose the nodes 1, 2, 3, and 4 in the left component are known as localizable. The localizability information, however, cannot propagate to the other part by trilateration since none of the nodes 5, 6, and 7 connects to three localizable nodes. Obviously, trilateration wrongly reports that nodes in the right component are not localizable, ignoring the fact that the entire graph is globally rigid.
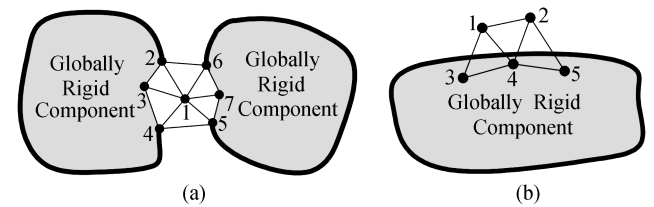

Fig.22. Deficiency of trilateration. (a) Geographical gap. (b) Border nodes.

A similar situation recurs for the border nodes, as illustrated in Fig.22(b). In this case, the border nodes 1 and 2 cannot be localized by trilateration even though nodes 3, 4, and 5 know their locations. Actually, the entire graph in Fig.22(b) is globally rigid and thus localizable. Discarding locating border nodes is unacceptable, as border nodes often play critical roles in many applications. For example, a sensor network for forbidden region monitoring has special interests on when and where intruders crash into, which are collected by border nodes only.

3.3.2 WHEEL

The limitations of trilaterations motivate another method to construct localizable networks based wheel graphs. A wheel graph W_n is a graph with n vertices, formed by connecting a single vertex to all vertices of an $(n - 1)$-cycle. The vertices in the cycle will be referred to as *rim vertices*, the central vertex as the *hub*, an edge between the hub and a rim vertex as a *spoke*, and an edge between two rim vertices as a *rim edge*. Fig.23 shows a particular realization of a wheel graph W_6, in which node 0 is the hub and others are rims.
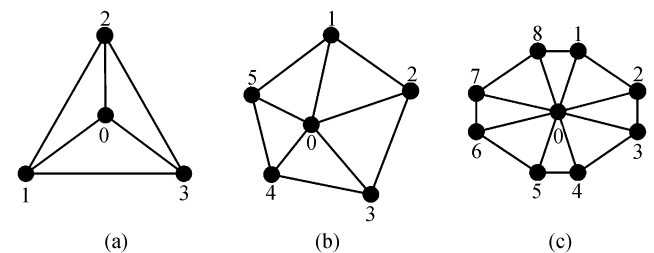

Fig.23. Wheel graphs. (a) W_4. (b) W_6. (c) W_9.

The wheel graph has many good properties. From the standpoint of the hub vertex, all elements, including vertices and edges, are in its one-hop neighborhood, which indicates that the wheel structure is fully included in the neighborhood graph of the hub vertex. Furthermore, wheel graphs are important for localizability because they are globally rigid in 2D space[77], which indicates an approach to identifying localizable vertices. Similar to the trilateration extension, the wheel extension is defined as follows.

Definition 1. *A graph G is a wheel extension if there are*

(a) *three pairwise connected vertices, say v_1, v_2, and v_3; and*

(b) *an ordering of remaining vertices as v_4, v_5, $v_6, \ldots$, such that any v_i is included in a wheel graph (a subgraph of G) containing three early vertices in the sequence.*

Theorem 3[77]. *The wheel extension is globally rigid.*

Fig.24 shows an example that is a wheel extension but not a trilateration extension. A particular iterative localization process on this example graph is illustrated in Fig.24. At beginning, three beacons are available at the bottom left. In the first iteration, nodes in the bottom left hexagon are identified because they are included in a wheel graph with 3 beacons. Such a procedure continues until all localizable nodes are marked.
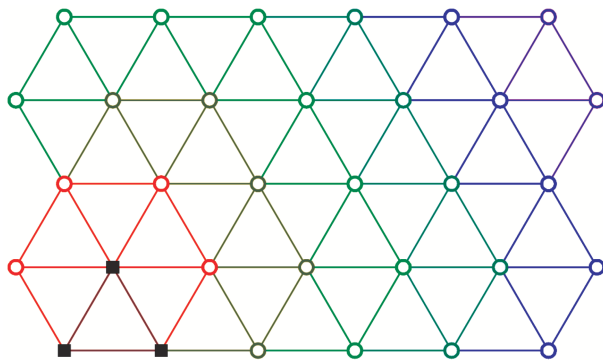

Fig.24. Wheel extension graph.

Compared to the previous trilateration (TRI) based methods, the advantages of the proposed method (WHEEL) lie in:

1) Capability: recognizing a superset of localizable nodes, as shown in Fig.25.

2) Efficiency: taking $O(n)$ running time for sparse graphs and $O(n^3)$ for dense ones, where n is the network size.

3) Low cost: introducing no extra wireless communication cost by using only localized information.
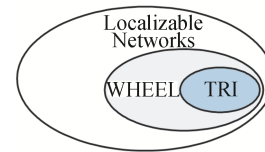

Fig.25. Trilateration extension is a subset of WHEEL extension.

Three examples are further provided to show how WHEEL outperforms TRI does. In Fig.26, a particular network with an "H" hole is generated in which 400 nodes are randomly distributed. The blue dots denote the nodes marked by TRI, while reds denote the nodes marked by WHEEL but not by TRI. Neither TRI nor WHEEL can mark the remaining blacks. WHEEL can easily step over gaps, such as borders or barriers, and recognize more nodes than TRI does. The same phenomenon recurs in all three network instances.

3.4 Node Localizability

Due to hardware or deployment constraints, for some applications, the networks are almost always not entirely localizable[78]. Indeed, theoretical analyses indicate that, in most cases, it is unlikely that all nodes in a network are localizable, but a (large) portion of nodes can be uniquely located[78]. Thus, the network localizability testing often fails unless networks are highly dense and regular.

On the other hand, nodes are not equally important since they play different roles in a network. Such differentiation can be application specific. For example, a sensor network for monitoring forbidden regions has
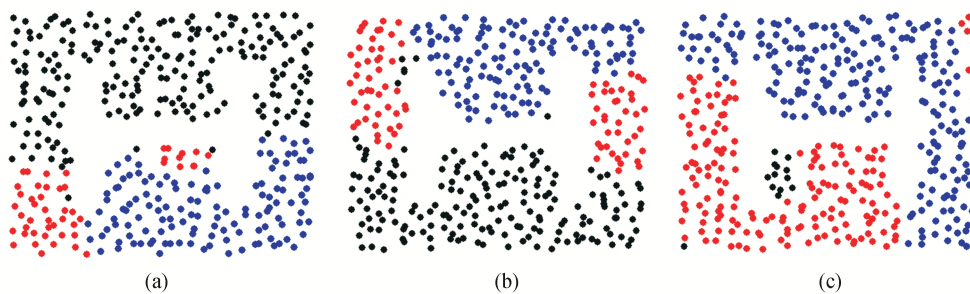

Fig.26. Networks with "H" holes. (a) Case 1. (b) Case 2. (c) Case 3.

special interest in when and where intruders enter, which are collected by border nodes only. In addition, many applications can function properly as long as a sufficient number of nodes are aware of their locations[78]. These observations motivate researchers to consider the localizability problem beyond the network localizability.

Although the theory of network localizability is complete, what we real desire is to answer the following two fundamental questions which cannot be solved by existing methods:

1) Given a network configuration, whether or not a specific node is localizable?

2) How many nodes in a network can be located and which are them?

Answering the above questions not only benefits localization, but also provides instructive directions to other location-based services. Therefore, the *node localizability* is addressed[78-79], which focuses on the location-uniqueness of every single node. Clearly, network localizability is a special case of node localizability in which all nodes are localizable. Thus, node localizability is a more general issue.

The first major challenge for studying node localizability is to identify uniquely localizable nodes. Following the results for network localizability, an obvious solution is to find a localizable subgraph from the distance graph, and identify all the nodes in the subgraph localizable. Unfortunately, such a straightforward attempt misses some localizable nodes and wrongly identifies them as non-localizable, since some conditions (e.g., 3-connectivity) essential to network localizability are no longer necessary to node localizability. As shown in Fig.27(a), the node u can be uniquely located under this network configuration but not included in the 3-connected component of beacons. The uniqueness of u's location is explained in Figs. 27(b) and 27(c) where we decompose the network into 2 subgraphs. As u connects 2 beacons in the right component, it has 2 possible locations denoted by u and u'. If we adopt u' as its location, it is impossible to embed the left subgraph into the plane. Specifically, the left subgraph has 2
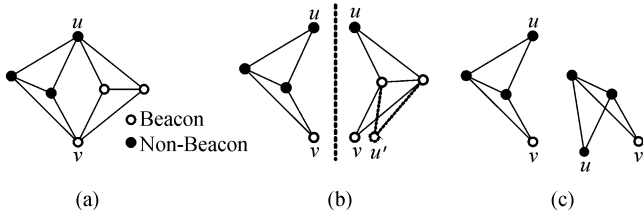

Fig.27. An example showing that the result from network localizability fails to identify node u as localizable. (a) u is localizable. (b) Graph decomposing. (c) 2 realizations of the left subgraph.

realizations, but neither of them is compatible with u'. Hence, u is uniquely localizable, although the 3-connectivity property does not hold. Motivated by the example, it is clear that the results derived for network localizability cannot be directly applied and we have to reconsider the conditions for node localizability.

The big picture of the state-of-the-arts of node localizability is shown in Fig.28. An obvious condition for a vertex to be localizable is that it must connect at least three neighbor vertices, i.e., having a degree of at least three. Goldenberg *et al.*[78] propose the first non-trivial necessary condition: if a vertex is localizable, it has 3 vertex-disjoint paths to 3 beacons. We denote such a condition as 3-paths for short. The necessity of 3-paths is obvious: if a vertex has only 2 vertex-disjoint paths to beacons, it definitely suffers from a potential flip ambiguity by reflecting along the line of a pair of cut vertices. Nevertheless, it is easy to find that some non-localizable vertices also satisfy the 3-paths condition. Redundant rigidity has been further proved essential to node localizability[79] and accordingly Yang *et al.* achieve so far the best necessary condition by combining 3P (3 vertex-disjoint paths) and redundant rigidity, which is called RR-3P for short.
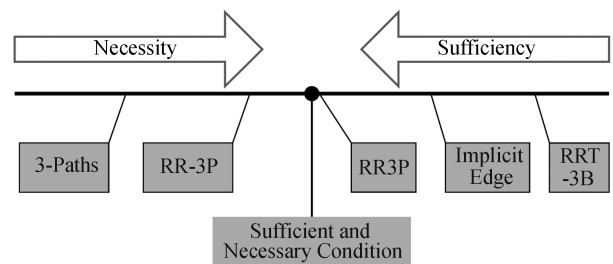

Fig.28. Evolution of conditions of node localizability.

On the other hand, for the sufficiency of node localizability, a straight-forward solution[78] is presented to identify localizable nodes by finding globally rigid subgraph (denoted by RRT-3B). An improvement has also been made by introducing the "implicit edge". Recent studies show that a vertex is localizable if it belongs to the redundantly rigid component of B in which there exists 3 vertex-disjoint paths connecting it to 3 beacon vertices (denoted by RR3P)[79]. Note that RR3P is fundamentally different from the previously mentioned RR-3P. RR3P requires the 3 paths strictly residing in the redundantly rigid component.

The RR-3P and RR3P conditions are implemented in a real-world wireless sensor network[79]. Fig.29 shows the results of node localizability testing; for the first time, it is possible to analyze how many nodes one can expect to locate in sparsely or moderately connected networks.
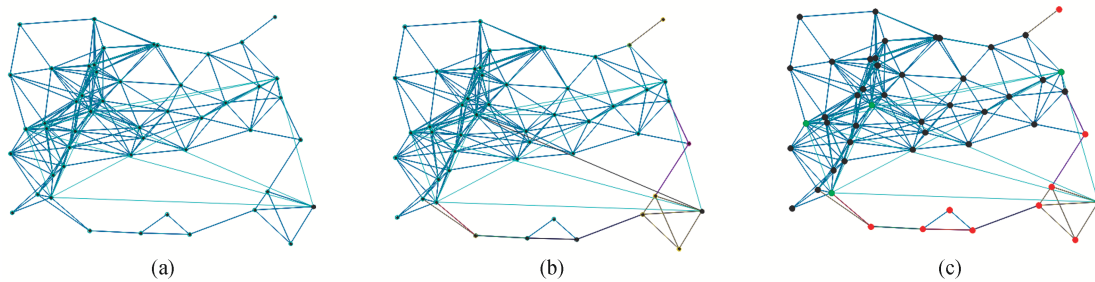
Fig.29. Localizability testing on a particular network instance in which greens are beacons, blacks are marked localizable, and reds are marked non-localizable. (a) Measured network topology. (b) RR3P subgraph. (c) Identifying localizable nodes.

Currently, the necessary and sufficient condition for node localizability is still open and researchers believe RR3P is the one. The sufficiency of RR3P is proved in [79]; while the necessity remains open, which is both challenging and worthwhile.

Another direction of future research with good potential is localizability with distance measuring errors. Previous studies have shown that measurement errors play an important role on localization. Some nodes uniquely localizable under perfect distance ranging may suffer from location ambiguities in a practical scenario of ranging errors. We envision this point in order to increase the robustness of localizability testing.

4 Conclusion

Location-awareness is a key feature of future-generation networks, enabling a large number of pervasive applications. In this article, we have provided an overview of existing localization approaches and discussed several research challenges, including location-aided network services, geographical information measuring, and design principles of algorithms. We put emphasis on error control mechanism and localizability theory, two rising aspects that attract significant research interests in recent years. We offer this survey to help researchers to understand the state-of-the-art and to address directions of future research in the new and largely open areas of location-aware technologies.

References

[1] Varshneyy U. Pervasive healthcare. *IEEE Computer*, 2003, 36(12): 138-140.

[2] Borriello G, Stanford V, Narayanaswami C, Menning W. Pervasive computing in healthcare. *IEEE Pervasive Computing*, 2007, 6(1): 17-19.

[3] Satyanarayanan M. Pervasive computing: Vision and challenges. *IEEE Personal Communications*, 2001, 8(4): 10-17.

[4] Weiser M. The computer for the twenty-first century. *Scientific American*, 1991, 265(3): 94-104.

[5] Angeles R. RFID technologies: Supply-chain applications and implementation issues. *Information Systems Management*, 2005, 22(1): 51-65.

[6] Gliddeny R. Design of ultra-low-cost UHF RFID tags for supply chain applications. *IEEE Communications Magazine*, 2004, 42(8): 140-151.

[7] Yang Z, Li M, Liu Y. Sea depth measurement with restricted floating sensors. In *Proc. IEEE RTSS*, Tucson, USA, Dec. 3-6, 2007, pp.469-478.

[8] Li M, Liu Y. Underground coal mine monitoring with wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 2009, 5(2): 10.

[9] Mo L, He Y, Liu Y *et al.* Canopy closure estimates with GreenOrbs: Sustainable sensing in the forest. In *Proc. ACM SenSys 2009*, Berkeley, USA, Nov. 4-6, 2009, pp.99-112.

[10] GreenOrbs Project. http://greenorbs.org/, 2010.

[11] OceanSense Project. http://www.cse.ust.hk/∼liu/Ocean/index.html, 2010.

[12] Li J, Jannotti J, Couto D S J D *et al.* A scalable location service for geographic ad hoc routing. In *Proc. ACM MobiCom 2000*, Boston, USA, Aug. 6-11, 2000, pp.120-130.

[13] Kortuem G, Schneider J, Preuitt D *et al.* When peer-to-peer comes face-to-face: Collaborative peer-to-peer computing in mobile ad hoc networks. In *Proc. Peer-to-Peer Computing*, Linkoping, Sweden, Aug. 27-29, 2001, pp.75-91.

[14] Patwari N, Ash J N, Kyperountas S *et al.* Locating the nodes: Cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, 2005, 22(4): 54-69.

[15] Sayed A H, Tarighat A, Khajehnouri N. Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Processing Magazine*, 2005, 22(4): 24-40.

[16] Karp B, Kungy H T. GPSR: Greedy perimeter stateless routing for wireless networks. In *Proc. ACM MobiCom 2000*, Boston, USA, Aug. 6-11, 2000, pp.243-254.

[17] Alzoubi K, Li X Y, Wang Y *et al.* Geometric spanners for wireless ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 2003, 14(4): 408-421.

[18] Li N, Hou J C. Localized topology control algorithms for heterogeneous wireless networks. *IEEE/ACM Transactions on Networking (TON)*, 2005, 13(6): 1313-1324.

[19] Cardei M, Du D Z. Improving wireless sensor network lifetime through power aware organization. *Wireless Networks*, 2005, 11(3): 333-340.

[20] Wan P J, Yi C W. Coverage by randomly deployed wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 2006, 14(SI): 2658-2669.

[21] Xue F, Kumar P R. On the θ-coverage and connectivity of large random networks. *IEEE/ACM Transactions on Networking (TON)*, 2006, 14(SI): 2289-2299.

[22] Li X Y, Wan P J, Frieder O. Coverage in wireless ad hoc sensor networks. *IEEE Transactions on Computers*, 2003, 52(6): 753-763.

296

J. Comput. Sci. & Technol., Mar. 2010, Vol.25, No.2

[23] Fang Q, Gao J, Guibas L J. Locating and bypassing routing holes in sensor networks. In *Proc. IEEE INFOCOM*, Hong Kong, China, March 7-11, 2004, pp.2458-2468.

[24] Abbasi A A, Younis M. A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, 2007, 30: 2826-2841.

[25] Savvides A, Han C, Strivastava M B. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proc. ACM MobiCom 2001*, Rome, Italy, 2001, pp.166-179.

[26] Seidel S Y, Rappaport T S. 914 MHz path loss prediction models for indoor wireless communications in multifloored buildings. *IEEE Transactions on Antennas and Propagation*, 1992, 40(2): 209-217.

[27] Priyantha N B, Chakraborty A, Balakrishnan H. The cricket location-support system. In *Proc. ACM MobiCom 2000*, Boston, USA, Aug. 6-11, 2000, pp.32-43.

[28] Whitehouse K, Culler D. Calibration as parameter estimation in sensor networks. In *Proc. ACM WSNA 2002*, Atlanta, USA, Sep. 28, 2008, pp.59-67.

[29] Girod L, Estrin D. Robust range estimation using acoustic and multimodal sensing. In *Proc. IROS 2001*, Maui, USA, Oct. 29-Nov. 3, 2001, pp.1312-1320.

[30] Peng C, Shen G, Zhang Y *et al.* BeepBeep: A high accuracy acoustic ranging system using COTS mobile devices. In *Proc. ACM SenSys 2007*, Sydney, Australia, Nov. 6-9, 2007, pp.59-72.

[31] Priyantha N B, Miu A, Balakrishnan H, Teller S. The cricket compass for context-aware mobile applications. In *Proc. ACM MobiCom 2001*, Rome, Italy, 2001, pp.1-14.

[32] Nasipuri A, Li K. A directionality based location discovery scheme for wireless sensor networks. In *Proc. WSNA 2002*, Atlanta, USA, Sep. 28, 2002, pp.105-111.

[33] Niculescu D, Nath B. Ad hoc positioning system (APS) using AoA. In *Proc. IEEE INFOCOM 2003*, San Francisco, USA, March 30-April 3, 2003, pp.1734-1743.

[34] Doherty L, Pister K S J, Ghaoui L E. Convex position estimation in wireless sensor networks. In *Proc. IEEE INFOCOM*, Anchorage, USA, April 22-26, 2001, pp.1655-1663.

[35] Galstyan A, Krishnamachari B, Lerman K, Pattem S. Distributed online localization in sensor networks using a moving target. In *Proc. ACM/IEEE IPSN*, Berkeley, USA, April 26-27, 2002, pp.61-70.

[36] He T, Huang C, Blum B M *et al.* Range-free localization schemes in large scale sensor networks. In *Proc. ACM MobiCom 2003*, San Diego, USA, Sep. 14-19, 2003, pp.81-95.

[37] Kleinrock L, Silvester J A. Optimum transmission radii for packet radio networks or why six is a magic number. In *Proc. IEEE National Telecommunications Conference*, Birmingham, USA, Dec. 3-6, 1978, pp.4.3.1-4.3.5.

[38] Nagpal R, Shrobe H, Bachrach J. Organizing a global coordinate system from local information on an ad hoc sensor network. In *Proc. ACM/IEEE IPSN 2003*, Palo Alto, USA, April 22-23, 2003, pp.333-348.

[39] Lim H, Hou J C. Localization for anisotropic sensor networks. In *Proc. IEEE INFOCOM 2005*, Miami, USA, March 13-17, 2005, pp.138-149.

[40] Wang Y, Gao J, Mitchell J. Boundary recognition in sensor networks by topological methods. In *Proc. ACM MobiCom*, Los Angeles, USA, Sept. 23-29, 2006, pp.122-133.

[41] Li M, Liu Y. Rendered path: Range-free localization in anisotropic sensor networks with holes. In *Proc. ACM MobiCom 2007*, Montreal, Canada, Sep. 9-14, 2007, pp.51-62.

[42] Ni L M, Liu Y, Lau Y C, Patil A. LANDMARC: Indoor location sensing using active RFID. *ACM Wireless Networks*, 2004, 10(6): 701-710.

[43] Bulusu N, Heidemann J, Estrin D. GPS—Less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 2000, 7(5): 28-34.

[44] Gezici S, Tian Z, Giannakis G B *et al.* Localization via ultra-wideband radios. *IEEE Signal Processing Magazine*, 2005, 22(4): 70-84.

[45] Nanotron Technologies. http://www.nanotron.com.

[46] Shang Y, Ruml W, Zhang Y, Fromherz M P J. Localization from mere connectivity. In *Proc. ACM MobiHoc 2003*, Annapolis, USA, June 1-3, 2003, pp.201-212.

[47] Bachrach J, Taylor C. Localization in Sensor Networks. Handbook of Sensor Networks: Algorithms and Architectures, Stojmenovic I (ed.), Wiley, 2005.

[48] Niculescu D, Nath B. Ad hoc positioning system (APS). In *Proc. IEEE GLOBECOM 2001*, San Antonio, USA, Nov. 25-29, 2001, pp.407-415.

[49] Niculescu D, Nath B. DV based positioning in ad hoc networks. *Journal of Telecommunication Systems*, 2003, 22(1-4): 267-280.

[50] Whitehouse K, Woo A, Karlof C, Jiang F, Culler D. The effects of ranging noise on multi-hop localization: An empirical study. In *Proc. ACM/IEEE IPSN 2005*, Los Angeles, USA, April 24-27, 2005, No.10.

[51] Niculescu D, Nath B. Error characteristics of ad hoc positioning systems (APS). In *Proc. ACM MobiHoc 2004*, Tokyo, Japan, May 24-26, 2004, pp.20-30.

[52] Moore D, Leonard J, Rus D, Teller S. Robust distributed network localization with noisy range measurements. In *Proc. ACM SenSys 2004*, Baltimore, USA, Nov. 3-5, 2004, pp.50-61.

[53] Goldenberg D, Bihler P, Cao M *et al.* Localization in sparse networks using sweeps. In *Proc. ACM MobiCom 2006*, Los Angeles, USA, Sep. 23-29, 2006, pp.110-121.

[54] Capkun S, Hamdi M, Hubaux J P. GPS-free positioning in mobile ad hoc networks. In *Proc. Hawaii International Conference on System Sciences*, Hawaii, USA, Jan. 3-6, 2001.

[55] Yang Z, Liu Y. Quality of trilateration: Confidence based iterative localization. In *Proc. IEEE ICDCS 2008*, Beijing, China, June 17-20, pp.446-453.

[56] Wang X, Luo J, Li S *et al.* Component based localization in sparse wireless ad hoc and sensor networks. In *Proc. IEEE ICNP 2008*, Orlando, USA, Oct. 19-22, pp.288-297.

[57] Youssef M, Mah M, Agrawala A K. Challenges: Device-free passive localization for wireless environments. In *Proc. ACM MobiCom 2007*, Montreal, Canada, Sept. 9-14, 2007, pp.222-229.

[58] Liu Y, Chen L, Pei J, Chen Q, Zhao Y. Mining frequent trajectory patterns for activity monitoring using radio frequency tag arrays. In *Proc. PerCom 2007*, New York, USA, Mar. 19-23, 2007, pp.37-46.

[59] Fontana R J, Gunderson S J. Ultra-wideband precision asset location system. In *Proc. Ultra Wideband Systems and Technologies*, Baltimore, USA, May 20-23, 2002, pp.147-150.

[60] Lanzisera S, Lin D, Pister K. RF time of flight ranging for wireless sensor network localization. In *Proc. WISES 2006*, Vienna, Austria, June 30, 2006.

[61] Trees H L V. Detection, Estimation and Modulation Theory, Part I. New York: Wiley, 1968.

[62] Liu J, Zhang Y, Zhao F. Robust distributed node localization with error management. In *Proc. ACM MobiHoc 2006*, Florence, Italy, May 22-25, 2006, pp.250-261.

[63] Spirito M A. On the accuracy of cellular mobile station location estimation. *IEEE Transactions on Vehicular Technology (TVT)*, 2001, 50(3): 674-685.

[64] Savarese C, Langendoen K, Rabaey J. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *Proc. USENIX Annual Technical Conference*, June 10-15, 2002, pp.317-327.

[65] Liu D, Ning P, Du W. Attack-resistant location estimation in sensor networks. In *Proc. ACM/IEEE IPSN 2005*, Los Angeles, USA, April 24-27, 2005, No.13.

[66] Li Z, Trappe W, Zhang Y, Nath B. Robust statistical methods for securing wireless localization in sensor networks. In *Proc. ACM/IEEE IPSN 2005*, Los Angeles, USA, April 24-27, 2005, No. 12.

[67] Kung H T, Lin C K, Lin T H, Vlah D. Localization with snap-inducing shaped residuals (SISR): Coping with errors in measurement. In *Proc. ACM MobiCom 2009*, Beijing, China, Sept. 20-25, 2009, pp.333-344.

[68] Jian L, Yang Z, Liu Y. Beyond triangle inequality: Sifting noisy and outlier ranging measurements. In *Proc. IEEE IN-FOCOM 2010*, San Diego, USA, March 15-19, 2010. (To Appear)

[69] Eren T, Goldenberg D K, Whiteley W *et al.* Rigidity, computation, and randomization in network localization. In *Proc. IEEE INFOCOM*, Hong Kong, China, March 7-11, 2004, pp.2673-2684.

[70] Hendrickson B. Conditions for unique graph realizations. *SIAM Journal of Computing*, 1992, 21(1): 65-84.

[71] Jackson B, Jordan T. Connected rigidity matroids and unique realizations of graphs. *Journal of Combinatorial Theory Series B*, 2005, 94(1): 1-29.

[72] Laman G. On graphs and rigidity of plane skeletal structures. *Journal of Engineering Mathematics*, 1970, 4(4): 331-340.

[73] Jacobs D J, Hendrickson B. An algorithm for two-dimensional rigidity percolation: The pebble game. *Journal of Computational Physics*, 1997, 137(2): 346-365.

[74] Hopcroft J E, Tarjan R E. Finding the triconnected components of a graph. Technical Report, Department of Computer Science, Cornell University, TR 140, 1972.

[75] Lovasz L, Yemini Y. On generic rigidity in the plane. *SIAM Journal on Algebraic and Discrete Methods*, 1982, 3(1): 91-98.

[76] Aspnes J, Eren T, Goldenberg D K *et al.* A theory of network localization. *IEEE Transactions on Mobile Computing (TMC)*, 2006, 5(12): 1663-1678.

[77] Yang Z, Liu Y, Li X Y. Beyond trilateration: On the localizability of wireless ad-hoc networks. In *Proc. IEEE INFO-COM 2009*, Rio de Janeiro, Brazil, April 19-25, 2009.

[78] Goldenberg D, Krishnamurthy A, Maness W *et al.* Network localization in partially localizable networks. In *Proc. IEEE INFOCOM 2005*, Miami, USA, March 13-17, 2005, pp.313-326.

[79] Yang Z, Liu Y. Understanding node localizability of wireless ad-hoc networks. In *Proc. IEEE INFOCOM 2010*, San Diego, USA, March 15-19, 2010. (To appear)
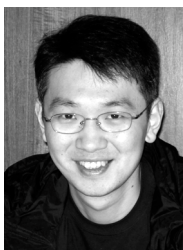
Yunhao Liu received the B.S. degree in automation from Tsinghua University, China, in 1995, the M.A. degree from the Beijing Foreign Studies University, China, in 1997, and the M.S. and Ph.D. degrees in computer science and engineering from Michigan State University, USA, in 2003 and 2004, respectively. He is now an associate professor and the postgraduate director at the Department of Computer Science and Engineering in the Hong Kong University of Science and Technology. He is also a member Tsinghua EMC Chair Professor Group. His research interests include wireless sensor network, peer-to-peer computing, and pervasive computing. He is a senior member of IEEE and a member of ACM. He is the Grand Prize Winner of Hong Kong ICT Best Innovation and Research Award 2007.
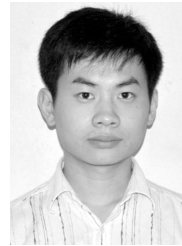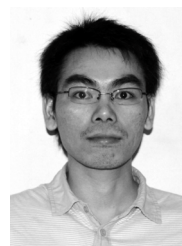
Zheng Yang received his B.S. degree in computer science, Tsinghua University, Beijing, China, in 2006. He is currently a Ph.D. candidate at the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. His research interests include wireless ad-hoc/sensor networks, and pervasive computing. He is a student member of the IEEE and the ACM.
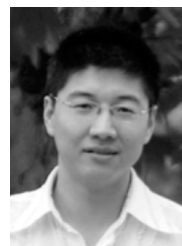
Xiaoping Wang received his B.S. and M.S. degrees from the School of Computer Science, National University of Defense Technology, China, in 2003 and 2006, respectively. He is currently a visiting Ph.D. candidate at the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong. His research interests include sensor networking and operating system. He is a student member of the IEEE.

Lirong Jian received his B.S. degree in computer science, Tsinghua University, Beijing, China, in 2008. He is currently a Ph.D. candidate at the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. His research interests include wireless networks and pervasive computing. He is a student member of the IEEE.