Available online at www.sciencedirect.com
ScienceDirect
Computer Communications 30 (2007) 2577–2592
computer communications
www.elsevier.com/locate/comcom

Path planning of mobile landmarks for localization in wireless sensor networks

Dimitrios Koutsonikolas, Saumitra M. Das, Y. Charlie Hu *

Center for Wireless Systems and Applications in the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA

Available online 27 June 2007

Abstract

Many applications of wireless sensor networks require the sensor nodes to obtain their locations. The main idea in most localization methods has been that some statically deployed nodes (landmarks) with known coordinates (e.g., GPS-equipped nodes) transmit beacons with their coordinates in order to help other nodes to localize themselves. A promising method that significantly reduces the cost is to replace the set of statically deployed GPS-enhanced sensors with one mobile landmark equipped with a GPS unit that moves to cover the entire network. In this case, a fundamental research issue is the planning of the path that the mobile landmark should travel along in order to minimize the localization error as well as the time required to localize the whole network. These two objectives can potentially conflict with each other.

In this paper, we first study three different trajectories for the mobile landmark, namely SCAN, DOUBLE SCAN, and HILBERT. We show that any deterministic trajectory that covers the whole area offers significant benefits compared to a random movement of the landmark. When the mobile landmark traverses the network area at a fine resolution, SCAN has the lowest localization error among the three trajectories, followed closely by HILBERT. But when the resolution of the trajectory is larger than the communication range, the HILBERT space-filling curve offers significantly better accuracy than the other two trajectories. We further study the tradeoffs between the trajectory resolution and the localization accuracy in the presence of 2-hop localization, in which sensors that have already obtained an estimate of their positions help to localize other sensors. We show that under moderate sensor mobility, 2-hop localization along with a good trajectory reduces the average localization error over time by about 40%.

Keywords: Wireless sensor networks; Localization; Mobile robots; Mobile landmarks; Path planning

1. Introduction

Many applications of wireless sensor networks require the sensor nodes to obtain their locations. For example, a deployed sensor network to monitor fires should ideally be able to pinpoint the location of the fire with some accuracy to enable firefighters to respond. Thus, sensor data in many applications is associated with the location where the data was sensed. In addition, many routing and data dissemination schemes rely on locations of sensor nodes being known [37,11]. For these reasons, localization of sensor networks has been an actively researched problem (e.g., [20,8,16,5,30,27,2,33,14]) .

A trivial method for sensor localization is for the sensors to be equipped with GPS [36]. However, there are several arguments against this architecture. GPS units increase the costs of sensor devices especially for large deployments. Additionally, the power consumption of the GPS devices reduces the lifetime of sensor networks. Finally, GPS on sensor devices reduces their deployability due to the increased form factor.

To mitigate such issues with GPS deployment on sensors, several distributed localization schemes have been proposed that do not require GPS on all sensor nodes [20,8,31,16,5,12,30]. In this case, only a fraction of the sensors have GPS units and these nodes transmit their

* Corresponding author. Tel.: +1 765 494 9143.
E-mail addresses: dkoutson@purdue.edu (D. Koutsonikolas), smdas@purdue.edu (S.M. Das), ychu@purdue.edu (Y.C. Hu).

0140-3664/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2007.05.048

coordinates to the rest of the sensors, in order to help them localize themselves. A promising method to localize sensor networks is to use one mobile landmark, e.g., a mobile robot [24,10,33]. Such mobile landmarks are equipped with GPS units and move throughout the sensor network area providing sensor nodes with their locations. Such an architecture offers significant practical benefits. The size of a robot is much larger than the size of a sensor and thus it is much easier to install a GPS unit on it. Moreover, a robot is not as energy constrained as a sensor. Since the localization accuracy can always be improved by increasing the resolution of the movement trajectory if the mobile landmark can move arbitrarily faster, a fundamental research issue when using mobile landmarks is the planning of the movement trajectory of the mobile landmark in order to maximize the localization accuracy, for a given velocity of the mobile landmark.

The problem of finding a good mobile landmark trajectory was discussed in [33] in which the authors made two important observations. First, a node is best localized when the mobile landmark passes close to it, because in that case the Received Signal Strength (RSS) is the largest. Second, many collinear beacons (beacons transmitted by the mobile landmark when it moves on a straight line) do not help localization, since the sensor still cannot determine on which side of the line it is, hence at least one non-collinear beacon is necessary. In spite of these observations, in their study they do not consider any specific trajectory for the mobile landmark.

In this paper, we study the design of mobile landmark trajectories to maximize the localization accuracy for sensor networks. We first show that a carefully selected deterministic trajectory can guarantee that all the sensors receive beacons and obtain an estimate for their positions, and it significantly reduces the average localization error, compared to random movement. We examine in detail three different deterministic trajectories, namely SCAN, DOUBLE SCAN, and HILBERT. Our results show that among the three trajectories, SCAN offers the best performance when the trajectory has a fine resolution and hence the average distance between the sensors and the trajectory is small. But for trajectories with a coarse resolution, HILBERT is the best choice. To our knowledge, this is the first study of mobile landmark trajectories for sensor network localization.

We further study the tradeoffs between the resolution of trajectories and the localization accuracy in the presence of 2-hop localization, in scenarios with moderate sensor mobility. The location errors for a set of mobile sensors come from two sources. One source of error is the localization algorithm itself, i.e., when a sensor receives beacons from the mobile landmark and estimates its position using the localization algorithm. The other source of error is the sensor's own movement before it can perform the localization operation again, i.e., when the mobile landmark finishes a complete round traversing the network area and the beacons can reach the sensor again. While the localization error can be reduced by having the mobile landmark traveling a more refined trajectory, doing so also elongates the duration between consecutive localization operations (fixing the velocity of the mobile landmark). Furthermore, traveling along a coarse trajectory may cause certain sensors not to be localized as they are far away from the trajectory and do not receive any beacons. Such sensors have to resort to 2-hop localization, i.e., be localized using beacons emitted from other sensors that have been localized using beacons directly sent from the mobile landmark. 2-hop localization, however, can introduce accumulative error. Despite this, our results show that with a moderate sensor mobility and using the HILBERT trajectory, 2-hop localization reduces the localization error by about 40% compared to 1-hop localization.

The rest of the paper is organized as follows. Section 2 surveys various schemes for sensor network localization. Section 3 gives a background on the single-landmark localization algorithm we use to evaluate the performance of the three trajectories. Section 4 describes the three different trajectories, namely SCAN, DOUBLE SCAN, and HILBERT. Section 5 describes the 2-hop localization scheme we use to study the tradeoffs between the trajectory resolution and the localization accuracy. Section 6 describes the experimental setup and Section 7 presents the simulation results. Finally, Section 8 concludes the paper.

2. Related work

There has been a large body of research on localization for wireless sensor networks over the last few years. They share the same main idea that nodes with unknown coordinates are helped by one or more nodes with known coordinates (e.g., GPS-equipped nodes) in order to estimate their positions. The nodes with known coordinates are called *landmarks*, *anchors*, or *seeds*. The various localization schemes can be classified based on the mobility state of nodes and landmarks. Schemes within each category can be further classified as range-free or range-based. Range-free techniques only use connectivity information between sensors and landmarks, while range-based techniques use distance or angle estimates in their locations estimations. Range-based techniques have used Received Signal Strength (RSS), Time Difference of Arrival (TDoA), or Angle of Arrival (AoA). Such a taxonomy is shown in Fig. 1. In the rest of this section, we briefly survey representative schemes in each category.

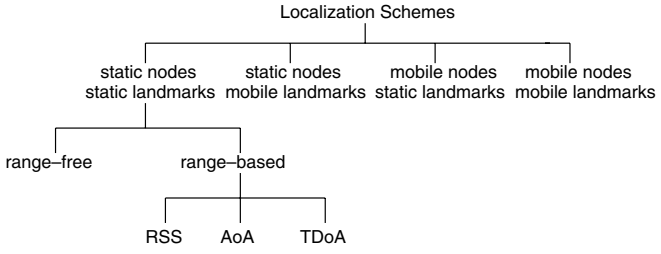

Fig. 1. A taxonomy of localization schemes.

2.1. Static networks – static nodes and landmarks

Range-based schemes in this category include [20–22,8,30,25]. In Ad Hoc Positioning System (APS) [20–22], information about landmarks is propagated in a hop-by-hop mode to the whole network. This information can be range measurements (DV-Distance, Euclidean), angle measurements, or a combination of them. Time Positioning System (TPS) [8] is based on Time Difference of Arrival (TDoA) of RF signals. As opposed to APS, TPS uses only three landmarks with high transmission power (three base stations), placed in appropriate positions, such as the whole monitored area to be enclosed within the angle formed by them. AHLoS [30], uses an approach similar to APS, but also employs Time Difference of Arrival (TDoA), instead of RF ranging. AFL [25] is another interesting approach in which localization is performed without the existence of any GPS-equipped node.

As opposed to range-based methods, range-free ones only use the content of messages received from neighbor nodes, in order to estimate their locations. DV-Hop [20] is a range-free version of APS in which nodes propagate hop count distance to landmarks. The idea of DV-Hop has been used by several other schemes, such as the Amorphous localization algorithm [19] and the Probability Grid [35]. Two more range-free schemes are the Centroid method [5], in which each node estimates its location by calculating the center of the locations of the landmarks it hears, and the APIT method [12], which divides the deployment area in triangular regions between landmarks, and uses a grid algorithm to estimate the largest area in which a node is likely to be found.

Similar to APS, there are some more schemes which can be either range-based or range-free. One such example is Multi-Dimensional Scaling (MDS) [32,31,16], which has its origins in psychometrics/psychophysics. MDS views similarities between data as distances and finds a placement of points in a low-dimensional space, where distances between points approximate original similarities.

2.2. Mobile nodes, static landmarks

The schemes in this category are indoor location systems, which are used to locate mobile users in buildings. These schemes usually consist of two phases: an offline phase, in which a database of signal strengths at various points with respect to known landmarks is built, and the online localization phase, during which a mobile user compares the signal strength values it received from different landmarks, with the stored values in the database, and the best fit gives an estimate of its location. Most known systems in this category are RADAR [2] and LEASE [17] which use the Received Signal Strength Indicator RSSI, Cricket [27] which uses a combination of radio and acoustic ranging, and VOR Base Stations [23] which use a combination of distance and angle measurements.

2.3. Static nodes, mobile landmarks

The need to reduce the number of expensive GPS-equipped nodes has led to a new approach. In this approach, only one mobile landmark (e.g., a robot, a man, or a vehicle) is used in order to localize a set of static sensors. The landmark traverses the deployment area and either periodically transmits beacons with its coordinates to help sensors to estimate their positions, or receives beacons transmitted by unknown nodes and estimates their positions, applying some signal processing technique.

In [24], a robot moves in the monitored area and localizes all nodes based on the RSSI of beacons it receives by them. The authors use a Robust Extended Kalman Filter (REKF) as a state estimator in predicting sensor locations. In [10] and [9], nodes localize themselves based on beacons they receive from the robot and on their neighbors connectivity. They impose geometrical constraints on their location estimations, in order to reduce the uncertainty of their positions. A third scheme is proposed in [33]. Here, again sensor nodes localize themselves based on the RSS and the coordinates of a mobile landmark but instead of simply imposing geometrical constraints, they use Bayesian inference to process the received information and compute their positions. The common drawback in all these approaches is that they consider random movement for the mobile landmark. Only the authors in [33] discuss the problem of finding a good mobile landmark trajectory, but they do not propose any specific solution. To our best knowledge, we are the first to study specific mobile landmark trajectories for sensor network localization.

Different from the previous schemes, in [26] the authors propose Mobile-Assisted Localization (MAL), an algorithm that guides the robot in order to collect the necessary pairwise distances from nodes in order to perform localization. This scheme is different from our approach, because it does not assume location information on the robot. Moreover, MAL does not scale well for large network sizes, because the robot has to discover each node, one by one, and move around it. In our approach, we do not send the robot explicitly to each node, but instead we select a trajectory which will guarantee that all nodes will be able to receive beacons from the robot.

2.4. Mobile networks

All the approaches discussed above consider static networks. TPS in [8] and the REKF approach in [24] are also designed to work with mobile sensor nodes, but the computational load will be heavily increased. APS can also tolerate mobility for some fraction of nodes, but communication overhead will significantly increase. In contrast with [24], the other two schemes of the previous class cannot work with high mobility, because localization is performed at each node. Hence, when the robot is away, nodes cannot update their positions, and the localization error increases over time.

The Monte Carlo localization (MCL) method [14] is the only method which can be used in mobile sensor networks (where both nodes and landmarks can move) to exploit mobility and increase accuracy of location estimation. The key idea of MCL is to represent the a posteriori distribution of possible locations using a set of N weighted samples and to update them recursively in time using the importance sampling method. It gives satisfactory results, but it requires a very high density of mobile landmarks (1 landmark per transmission range is required for an accuracy of 40% of the radio range). In Section 5, we examine how a *2-hop localization* scheme, based on *Iterative Multilateration* proposed in [30], along with a carefully selected trajectory, can mitigate the mobility problem by using only one mobile landmark.

3. Background – localization algorithm

In studying the effectiveness of different mobile landmark trajectories, we use the localization algorithm proposed by Sichitiu et al. [33] which uses the Received Signal Strength Indicator (RSSI) for ranging and Bayesian inference to estimate the positions of the unknown nodes. However, the specific localization algorithm is orthogonal to our study and hence we expect our findings to remain valid when other RF-signal-based schemes are used (e.g., [9,10]). In the remainder of this section, we give a brief description of the algorithm.

Before running the algorithm, an offline calibration phase is necessary, which is described in the next section. This phase constructs the Probability Distribution Function (PDF) Table, which is stored at each node and maps every RSSI value to a PDF. According to the algorithm, the mobile landmark periodically broadcasts beacon packets as it traverses the deployment area. These packets contain the coordinates of the mobile landmark (x_B, y_B), which can be obtained by GPS. When a node receives a beacon packet, it performs a lookup at the PDF Table and obtains the probability distribution function of the distance corresponding to the RSSI of the beacon packet. Using this function, the sensor imposes the following constraint on its position estimation:

$$\begin{aligned} Constraint(x,y) &= \mathrm{PDF}_{\mathrm{RSSI}}(d((x,y),(x_B,y_B))) \\ &\forall (x,y) \in [(x_{\min},x_{\max}) \times (y_{\min},y_{\max})] \end{aligned} \tag{1}$$

where $\mathrm{PDF}_{\mathrm{RSSI}}$ is the probability distribution function, $d((x,y), (x_B,y_B))$ is the Euclidean distance between the points with coordinates (x,y) and (x_B,y_B), and $x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$ are the bounding coordinates of the sensor deployment area.

For the largest RSSI value stored in the PDF Table, which corresponds to the minimum distance, a Gaussian constraint would be a pessimistic choice, and hence a circular constraint is used instead, as follows:

$$\begin{aligned} Constraint(x,y) &= \begin{cases} \frac{1}{\pi \times d_{min}^2} & \text{if } d((x,y),(x_B,y_B)) \leqslant d_{\min} \\ 0 & \text{otherwise} \end{cases} \\ &\forall (x,y) \in [(x_{\min},x_{\max}) \times (y_{\min},y_{\max})] \end{aligned} \tag{2}$$

where $d_{\min}$ is the distance corresponding to the largest RSSI value. Bayesian inference is then applied and the new position estimate *NewPosEst* is computed for each node, based on the old position estimate *OldPosEst* and the new constraint *Constraint*:

$$\begin{aligned} NewPosEst(x,y) &= \frac{OldPosEst(x,y) \times Constraint(x,y)}{\int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} OldPosEst(x,y) \times Constraint(x,y)} \\ &\forall (x,y) \in [(x_{\min},x_{\max}) \times (y_{\min},y_{\max})] \end{aligned} \tag{3}$$

The initial position estimate for each sensor is initialized to a constant value, since in the beginning, a node is equally likely to be in any position in the deployment area.

This process is repeated for each received beacon packet. Finally, when the node stops receiving any more beacon packets, either because the mobile landmark has moved away, or because a maximum number of beacons has been received, the node uses the last position estimate *PosEst* to compute its best position coordinates $(\hat{x}, \hat{y})$ as follows:

$$\begin{aligned} \hat{x} &= \int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} x \times PosEst(x,y)\ \mathrm{d}x\mathrm{d}y \\ \hat{y} &= \int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} y \times PosEst(x,y)\ \mathrm{d}x\,\mathrm{d}y \end{aligned} \tag{4}$$

4. Mobile landmark trajectories

Finding the optimal trajectory of the mobile landmark for sensor network localization is a very challenging problem. Essentially, path planning for this particular application has two goals: (a) to offer network coverage and (b) to provide good quality beacons. Robot coverage has been well studied in robotics (e.g., [1,6,18,38]), where the goal is to ensure that the robot will travel over *all* points in a region (for tasks such as lawn mowing, spray painting, vacuuming, etc.). In our case, this goal becomes less strict and we do not require the robot to travel over all points of the deployment area. Instead, we simply want to ensure that all the sensors can receive some beacons from the mobile landmark. This goal can be achieved by any deterministic trajectory, as we show in Section 7, with properly selected parameters. The second goal of path planning, which is unique in the sensor network localization problem, is much more challenging. A set of beacons is considered "of good quality" for a sensor, if they are non-collinear, and their signal strengths can be accurately mapped to their distances from the sensor. However, the locations of the nodes are unknown, hence it is not possible to determine in advance a trajectory that will ensure good quality beacons for the sensors. Moreover, sensors may move, which makes the problem even more complicated. Finally, in a realistic

environment, multipath fading due to phenomena such as reflections, scatterings, etc. cause large *random* signal variations. Hence, the quality of a beacon does not depend only on its position with respect to the sensor, but also on the quality of the signal that particular moment.

The above discussion implies that analytical determination of the optimal trajectory of the mobile landmark in the context of sensor network localization is not feasible. Instead, in this paper, we compare three well-known trajectories, that offer general desirable characteristics, and identify, through detailed simulations, which of them offers higher localization accuracy. We assume that the network operator knows a priori the size of the target area and has an estimate about the sensors' velocities. The best trajectory is then selected offline and stored at the robot before its deployment. In the rest of this section, we describe these three different trajectories, namely SCAN, DOUBLE SCAN, and HILBERT. For each trajectory, we describe its basic characteristics, followed by a brief qualitative discussion on their advantages and disadvantages.

4.1. SCAN

SCAN is a simple and easily implemented trajectory. The mobile landmark traverses the network area along one dimension, as shown in Fig. 2(a). In this figure, the mobile landmark travels along the y axis, and the distance between two successive segments of the trajectory, parallel to the y axis, defines the resolution of the trajectory. If the communication range of the sensors is R, the resolution should be at most $2R$, to make sure that all the sensors will be able to receive beacons. If we denote by L the dimension of a square deployment area, we can estimate the total distance traveled by the mobile landmark when it covers the whole area as follows. Since its trajectory consists of $\frac{L}{R}+1$ segments of length L, parallel to y axis and $\frac{L}{R}$ segments of length R, parallel to x axis, the total distance D is given by the formula:

$$D = \left(\frac{L}{R}+1\right)\times L + \left(\frac{L}{R}\right)\times R = \left(\frac{L}{R}+2\right)\times L \tag{5}$$

SCAN has the advantage of offering uniform coverage to the whole network, and it ensures that all nodes will be able to receive beacons from the mobile landmark under a properly selected resolution. Moreover, uniformity keeps the maximum error low, as we will show in Section 7. However, SCAN has one important drawback – collinearity of beacons. For large resolution, many nodes will receive beacons only from one line segment and one direction, which will create uncertainty and prevent them from obtaining a good estimate along the x axis. To avoid this problem, the trajectory has to be dense enough for the sensors to be able to hear the mobile landmark when it moves on two successive segments along the y axis. In this case, the intersections of the imposed constraints eliminate the uncertainty and offer high accuracy.

4.2. DOUBLE SCAN

Another straightforward way to overcome the collinearity problem of SCAN is to scan the network along both directions, as shown in Fig. 2(b). In this case, the mobile landmark first traverses the whole network, scanning along the y axis, as in the previous case, and all the nodes obtain a good estimate for their y coordinate. Then the mobile landmark performs a second scanning along the x axis, giving the nodes the possibility to eliminate the uncertainty for their x coordinates. The problem with this method is that it requires the mobile landmark to travel doubled distance, compared to the simple scan, for the same resolution. In Fig. 2(b), we selected to keep the distance traveled by the mobile landmark similar for all trajectories, hence DOUBLE SCAN is performed with a double resolution compared to SCAN. According to this figure, if we denote the dimension of the deployment area by L, and the resolution of DOUBLE SCAN by $2R$, the total distance traveled by the mobile landmark is

$$\begin{aligned} D &= 2\left[\left(\frac{L-R}{2R}+1\right)\times L + \left(\frac{L-R}{2R}\right)\times 2R\right] \\ &= 2\left[\left(\frac{L-R}{2R}+2\right)\times L - R\right] \end{aligned} \tag{6}$$

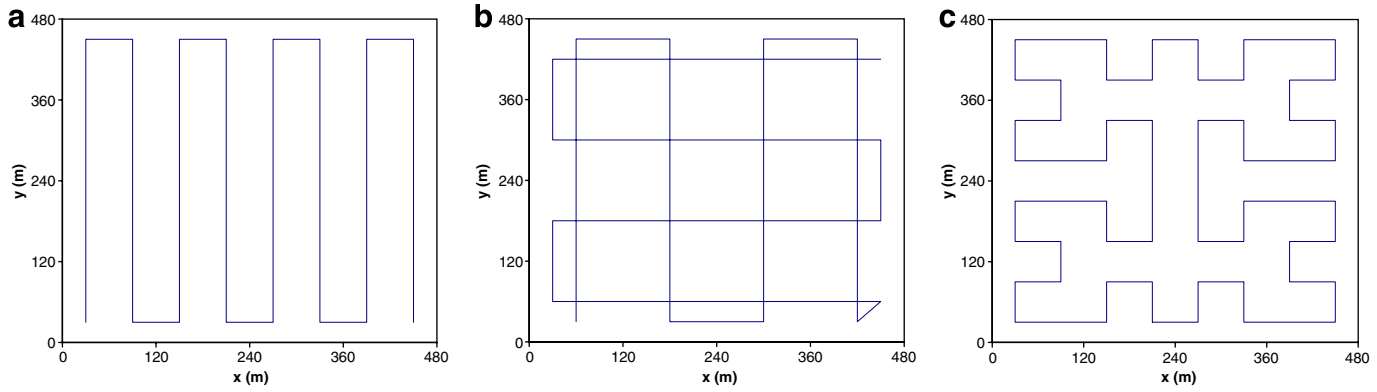

Fig. 2. The mobile landmark trajectories studied in the paper. (a) SCAN, deployment area 420 m × 420 m, resolution 60 m. (b) DOUBLE SCAN, deployment area 420 m × 420 m, resolution 120 m. (c) HILBERT, deployment area 420 m × 420 m, extended deployment area 480 m × 480 m, resolution 60 m.

4.3. HILBERT

A HILBERT space-filling curve [13] creates a linear ordering of points in a higher-dimensional space that preserves the physical adjacency of the points. Algorithms based on bit manipulation [7], finite-state diagrams [3], and recursive construction [15] exist to generate HILBERT curves. A level-n HILBERT curve divides the 2-dimensional space into 4^n square cells and connects the centers of those cells using 4^n line segments, each of length equal to the length of the side of a square cell. We define the resolution of the HILBERT curve as the length of each line segment, as shown in Fig. 2(c). A deployment area of dimension L and a resolution equal to R obviously can be divided into $\frac{L}{R} \times \frac{L}{R}$ squares of side length R, i.e., $4^n = \frac{L}{R} \times \frac{L}{R}$. In this case, the total distance traveled by the mobile landmark is given by

$$D = 4^n \times R = \left(\frac{L}{R}\right)^2 \times R = \frac{L^2}{R} \tag{7}$$

The key reason we study HILBERT curves in this paper is that such curves make many turns, compared to SCAN or DOUBLE SCAN. This implies that if the mobile landmark moves on a HILBERT curve, the sensors to be localized will have the chance to receive *non-collinear* beacons and obtain a good estimate for their positions. A HILBERT curve has also a potential drawback compared to SCAN or DOUBLE SCAN. Since this curve always connects the centers of two successive square cells, the mobile landmark will never move on the border of the deployment area. Thus sensors near the border will possibly receive beacons only from one direction and their estimates will not be accurate. To solve this problem, we virtually extend the dimensions of the deployment area by $\frac{R}{2}$ on each side. If L_r is the dimension of the real deployment area, then the area used in order to construct the HILBERT curve is $L_e = L_r + R$.

By replacing $L = L_e$ in Eq. (7) and $L = L_r$ in Eqs. (5) and (6), the total distances traveled by the mobile landmark with HILBERT, SCAN, and DOUBLE SCAN are given by:

$$D_{Hilbert} = 4^n \times R \tag{8}$$

$$D_{Scan} = (4^n - 1) \times R \tag{9}$$

$$D_{DoubleScan} = (4^n + 2^n - 4) \times R \tag{10}$$

Eqs. (8) and (9) show that the total distance for the two trajectories, HILBERT and SCAN, differs only by R.

Figs. 2(a–c), show the three trajectories in an area of 420 m × 420 m and resolution of 60 m for SCAN and HILBERT, and 120 m for DOUBLE SCAN. The HILBERT curve is a level-*3* curve and the extended area has dimensions 480 m × 480 m. The total distance traveled by the mobile landmark in these three figures, according to Eqs. (8)–(10), is equal to 3840 m for HILBERT, 3780 m for SSCAN, and 4080 m for DOUBLE SCAN.

5. Multihop localization

When sensors are static, the selection of a good trajectory is enough to ensure localization with high accuracy. In those cases, the mobile landmark has to traverse the network only once, and the position estimates obtained by the sensors will remain the same over time. But it is not always realistic to assume that sensors are static. When sensors are deployed in outdoor environments, their positions may randomly change because of many factors, such as wind, currents, and animals.

In a mobile scenario, the location error of sensors comes from two sources. One source is the localization error from the localization algorithm itself (due to the inherent noise in the translation of signal strength values to distances through the PDF Table), when a sensor receives beacons from the mobile landmark and executes the localization algorithm. This source also exists in static scenarios. The other, which appears only in mobile scenarios, is the sensor's movement between two consecutive localization epochs by the mobile landmark. We define a *localization epoch* as the duration in which the mobile landmark traverses the whole network area once.

The localization error can be reduced by having the mobile landmark travel along a more refined trajectory, since in that case sensors will be able to receive more beacons and from closer distances. However, a very fine-grained trajectory elongates the duration between two consecutive localization epochs. On the other hand, when the mobile landmark travels along a very coarse trajectory, certain sensors may not receive any beacons, because they are far away from the trajectory, remaining unlocalized.

For these sensors, *multihop localization* could be used to perform localization. The idea of a multihop localization scheme was first proposed by Savvides et al. in [30] under the name *Iterative Multilateration*, as well as by Sichitiu et al. in [34], but in both works it was only evaluated for static networks (static sensors/static landmarks). In multihop localization, nodes that receive beacons directly from the mobile landmark and obtain an estimate about their locations, broadcast beacons with their own position estimates. In this way, nodes that are far away from the mobile landmark can now localize themselves by using beacons emitted by other nodes. Hence, multihop localization allows the mobile landmark to travel along a coarse-grained trajectory in order to reduce the duration of each localization epoch, while still giving all the nodes the chance to receive some beacons and estimate their positions.

However, multihop localization does not remove the first source of error – the localization error caused when a node receives beacons from large distances. Furthermore, it can cause accumulative error, since many nodes are localized using location estimates from other nodes, which

already contain some error. In other words, with a coarse trajectory and multihop localization, we trade off instantaneous localization error with the error from node movement within each localization epoch, and the benefits depend on which of the two sources of error is larger. For static or very low mobility scenarios, multihop localization is not expected to help. But when the velocity of the sensors increases, it can potentially reduce the time-average error in location estimates.

To limit the error accumulation, we incorporated multihop localization into our localization algorithm as follows. Every time a sensor receives a beacon from the mobile landmark, it invokes the localization algorithm as before. It also sends out a beacon of its own which contains its position estimate if the following two conditions are met. First, it should have received at least three beacons from the mobile landmark or other sensors. Second, a large percentage of the beacons it has received should be from the mobile landmark. Since only nodes that hear the mobile landmark can meet these two conditions, multihop localization is effectively reduced to *2-hop localization*.

There are two important implementation details in this approach. First, as opposed to the mobile landmark, sensors do not transmit beacons periodically, but only when they themselves receive a new beacon. If two sensors transmit at the same moment (despite the random backoff at the link layer), a collision may happen, which will increase the probability for some other sensors not to receive this beacon and possibly remain unlocalized. To increase the possibility for its neighbors to receive a beacon it transmits, a sensor sends three back-to-back beacons with the same position estimate, instead of only one. The same sequence number is used in all the three beacons so that the neighbors can reject duplicates.

Second, each time the mobile landmark completes a whole traversal of the network and comes back to the vicinity of a sensor, that sensor resets its position estimate, since it is now stale, and starts localization again based on the new beacons it receives from the mobile landmark. To support this, the mobile landmark includes a *localization epoch number* in each beacon. A sensor compares the localization epoch number of each beacon it receives with the localization epoch number of the previous beacons. When the new localization epoch number is larger than the stored one, it resets its position estimate and starts a fresh localization using the new beacons. It also includes the new localization epoch number in the beacons it transmits.

6. Experimental methodology

6.1. System calibration

Before running the localization algorithm, a system calibration phase is necessary in order to construct the PDF Table, which is used by the algorithm. Following the method proposed in [33,34,29], we used two nodes in our simulator (one sender and one receiver), placing them in different distances between 2.5 m and 50 m. The communication range of the two nodes was set to 40 m. To make the simulation realistic, we used a Rician fading model, with a Rician k-factor = 5. The Rician model is used to describe environments where the source and the receiver are in LOS of each other and the direct signal component is much stronger than the components caused by reflections and scattering [28]. It gives a good approximation of the reality when sensors are deployed in an open area. For each distance, we took 1600 measurements of the signal strength. For each signal strength value, we computed the mean distance and the standard deviation, and stored this information in the PDF Table. The localization algorithm assumes that, for each signal strength value, the probability distribution function of this value versus distance is Gaussian, and this assumption was verified by our simulations, as well as by real world experiments in [33]. One example of this function is shown in Fig. 3(a) for RSSI = −52 dbm. However, it is interesting to see that there is a threshold in the signal strength, below which the Gaussian model is not valid anymore. In our simulator, we found this threshold to be equal to −80 dbm, which corresponds to physical distances of up to 40 m. Beyond this distance the noise in the signal strength measurements fluctuates due to phenomena such as reflection, scattering, and multipath propagation, and the probability distribution function of the signal strength versus distance can no longer be approximated by a Gaussian, as shown in Fig. 3(b) for RSSI = −86 dbm. Hence, in our PDF Table we only included RSS values larger than −80 dbm.

6.2. Experimental setup

Our simulations are performed using the Glomosim [39] simulator. Glomosim is a widely used mobile wireless network simulator with a detailed and accurate physical signal transmission model. We performed two different sets of simulations. In the first set, we evaluate the performance of three different mobile landmark trajectories, SCAN, DOUBLE SCAN, and HILBERT. We consider three different resolutions for each trajectory: 30 m, 45 m, and 60 m. The deployment area dimensions are set to be equal to $15 \times R$ in each case, where R is the resolution used. This gives areas of size 450×450 m, 675×675 m and 900×900 m, respectively. For HILBERT, the three extended areas are 480×480 m, 720×720 m, and 960×960 m, respectively. We note that the localization error does not depend on the node density; it only depends on the robot's trajectory resolution which determines the amount and nature of the beacons the sensors receive. Hence, the same node density is used in the three areas, and the numbers of sensor nodes are 660, 1485, and 2640, respectively. We assume a random sensor deployment and we evaluate our selected trajectories under this assumption. For different network topologies, different trajectories might have to be considered if the exact topology is known to the network operator a priori.

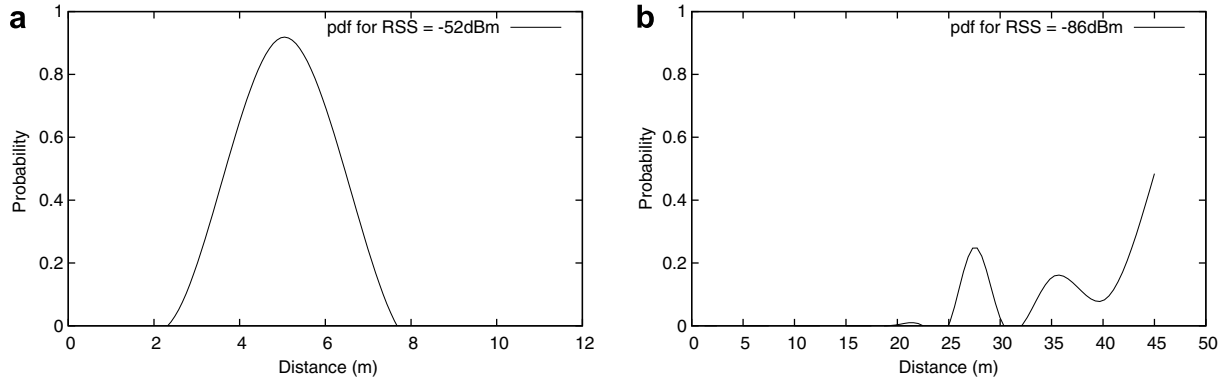

Fig. 3. Probability distribution functions (PDFs) for two different Received Signal Strength values. (a) PDF for RSS = −52 dbm. (b) PDF for RSS = −86 dbm.

However, we assume that typically the exact topology is not known since the sensors may be sprayed in large quantities, and hence a random distribution is the best way to model such a deployment. Note in this case, the operator still can easily get a rough estimate of the region after the sensors have been deployed. In all scenarios, sensors are static and a mobile landmark (robot) moves around them. The speed of the robot is constant and equal to 2 m/s. The simulation results are averaged over 10 runs.

To measure the localization accuracy under different trajectories, we measure the distance between the actual and the estimated position of a sensor. We consider the average localization error over all sensors. We also show the error along the x and the y axis separately.

In the second set of simulations, we keep the best of the three trajectories and we use it to study the tradeoffs between the trajectory resolution and localization accuracy in the presence of 2-hop localization. We use two different resolutions for the robot trajectory: 60 m and 120 m. With the first resolution, all nodes can receive beacons from the robot, so we do not use *2-hop localization*. The second resolution, however, is very large (3 times the transmission range), and *2-hop localization* is used so that all the sensors can obtain some estimate about their position.

In the second set of simulations, we consider only the largest of the three areas used in the first set, with dimensions 900 m × 900 m. The 2640 sensors are again placed randomly in the area, but this time they move. We use a modified version of the random waypoint model [4] to describe the movement of the sensors. The pause time is set to 0, and all the sensors move with the same velocity. We consider three different velocities for sensors: 1 m/ h, which simulates a very low mobility scenario, 30 m/h and 60 m/ h. The speed of the robot is kept to 2 m/s. Note that we are interested in practical scenarios, where the velocity of the sensors is orders of magnitude smaller than the velocity of the robot. To limit the error propagation due to beacons sent out by localized sensors, such beacons are sent out only after at least 80% of the beacons a sensor has received are from the robot. The simulation is repeated for many scenarios with different initial sensor distributions, but the results were very similar and we show them for only one scenario.

In both sets of simulations we use a wireless radio with 2 Mbps bit rate and 40 m transmission range. The beacon transmission interval is 2.5 s. This interval was selected to ensure that the robot will always transmit beacons at the points where it changes direction. This is very important in order to reduce the average localization error, as we will explain in Section 7.1.3, especially for the HILBERT trajectory, which makes many turns.

7. Experimental results

In this section, we first evaluate the impact of different trajectories on the localization accuracy. We then evaluate the impact of the trajectory resolution on the localization accuracy in the presence of multihop localization.

7.1. Performance comparison of different trajectories

We evaluate the performance of the three different trajectories under different resolutions.

7.1.1. Resolution 30 m

First, we simulate the trajectories with a fine resolution, in an area of 450 × 450 m. The localization errors are shown in Fig. 4(a). We observe that the average total localization error remains very small, lower than 1 m, for all three trajectories. We also observe that SCAN and DOUBLE SCAN have almost the same localization error (the difference is only 0.02%) and they both slightly outperform HILBERT by 3.5%. The reason for the high accuracy is that the resolution of 30 m is small compared to the 40m transmission range, and each node receives high signal strength beacons, which correspond to the entries with low standard deviations in the PDF Table.

The fine granularity of the trajectory is also the reason for which SCAN outperforms HILBERT in this case, although HILBERT has many more turns, offering the sensors the chance to receive more non-collinear beacons. Due to this granularity, any node is always able to receive beacons

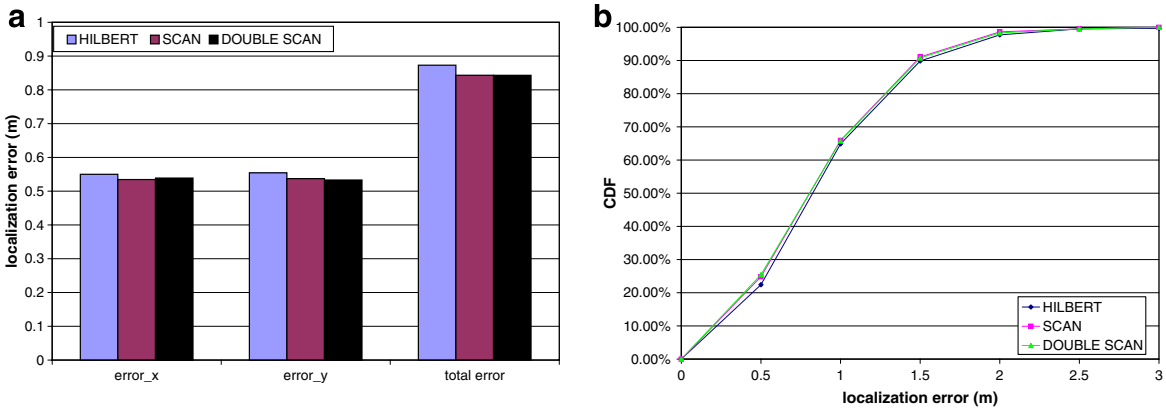

Fig. 4. Localization error and its CDF for resolution 30 m. (a) Localization error. (b) CDF of localization error.

from at least two different line segments, even with SCAN trajectory, and the intersection of the constraints imposed by the two beacons eliminates ambiguity. Hence the localization error with such a small resolution is only affected by the distances from which sensors receive beacons with each trajectory.

Fig. 5(a) shows that with SCAN, the worst position for a node is to be in the middle of the area defined by two consecutive line segments, parallel to *y*-axis. In that case, the node will always receive two beacons from a 15 m distance. These two beacons are enough for it to localize itself accurately, placing itself on the intersection of the two circles.

Fig. 5(b) shows the worst position for a node with HILBERT trajectory. In that case, the closest distance from which nodes like A can receive beacons, is 21.2 m, larger than 15 m. Hence, on average, nodes in case of HILBERT trajectory receive beacons from larger distances, compared to SCAN trajectory, and this causes the average localization error to be larger with the former trajectory. Note that in Fig. 5(a) as well as in Figs. 5(b–f) for simplicity we used

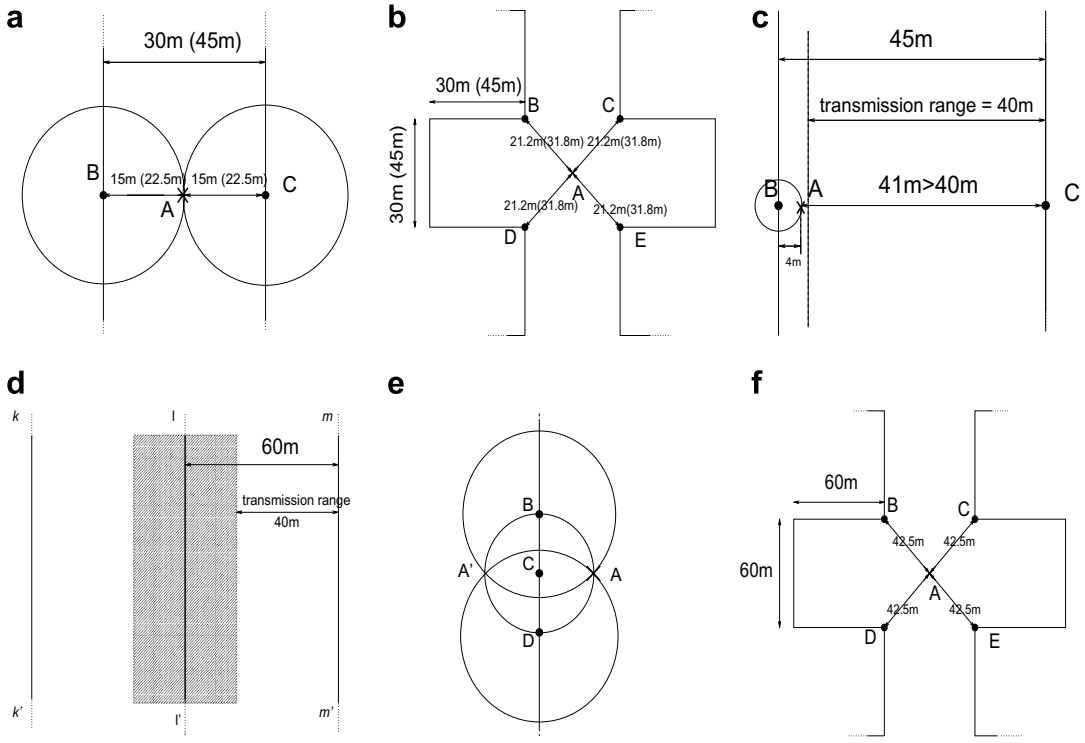

Fig. 5. Geometrical explanation of the trajectories performance. (a) SCAN with resolution 30 m (45 m) – in the worst case a node can receive 2 beacons from 15 m (22.5 m) distance. (b) HILBERT with resolution 30 m (45 m) – in the worst case a node can receive 4 beacons from 21.2 m (31.8 m) distance. (c) SCAN with resolution 45 m - node A can receive only collinear beacons, but from a very close distance, hence the localization error is still low. (d) SCAN with resolution 60 m - nodes in the shaded rectangular area can receive only collinear beacons. (e) A node that receives only collinear beacons cannot accurately determine its location. There are always two possible locations, on the left and on the right side of the robot trajectory. (f) A pathological case with HILBERT trajectory and large resolution: the distance between node A and the robot is always larger than the transmission range.

circular constraints instead of the Gaussian ones, described by Eqs. (1) and (3).

Fig. 4(a) shows that the error for SCAN along the x axis is smaller than the error along the y axis, as opposed to what we expected. Thus, with such a small resolution, scanning along both axes is not necessary. Note that the error with DOUBLE SCAN is the same to the error with SCAN, although it scans along both directions, because DOUBLE SCAN is performed with a double resolution. In this case, the advantage we get from the 2D scanning is counterbalanced by the fact that many nodes receive beacons with low signal strength and the distances corresponding to them have large standard deviations.

In Fig. 4(b) we observe that the Cumulative Distribution Function (CDF) for the localization error is almost the same for all the three trajectories. Moreover, more than 99% of the nodes have an error smaller than 2.5 m and the maximum error is 2.9 m. This error is same as the error observed by Sichitiu et al. in [33] with a random robot trajectory. But the authors in [33] tested their algorithm in a much smaller area than ours, in which each sensor could hear the robot at any moment. If the area is so large that the robot cannot always be in transmission range with all sensors, a random trajectory cannot guarantee that all sensors will be localized. To prove this, we repeated the simulation with the robot randomly moving in the deployment area for a total period of time equal to the time required for HILBERT to complete a whole traversal of the area. We used again a modified version of the random waypoint model to describe the robot's movement. The pause time was set to 0, and the robot's velocity was kept constant and equal to 2 m/s. In this scenario, 9% of the nodes could not receive any beacons, remaining unlocalized, and the average error for the rest of the nodes was 4.5 m, much larger than the error for any of the three trajectories we examine.

The conclusion is that when the robot can traverse the deployment area with a very fine resolution, the trajectory itself is not important. The high accuracy of the signal strength measurements ensures that the localization error will remain very small and the only requirement for the trajectory is to cover the whole area, thus any deterministic trajectory that can guarantee this will give similar performance, as opposed to a random movement. The type of the trajectory becomes critical, when we cannot afford to send the robot to cover the area with a very small resolution due to time or energy constraints.

7.1.2. Resolution 45 m

Fig. 6(a) shows that the increased resolution affects both the absolute values for the localization errors and the relative performance of the three trajectories. As we can see, the average localization error is now increased by 33%, 26%, and 123% for HILBERT, SCAN, and DOUBLE SCAN, respectively. The robot now does not pass as close to the nodes as in the previous case, so many beacons received by sensor nodes have lower signal strengths and thus larger standard deviations, which increase the error.

Surprisingly, SCAN still gives the lowest localization error, outperforming HILBERT and DOUBLE SCAN by 9% and 77%, respectively. In case of SCAN, most nodes are still able to receive beacons from two directions (left and right) and resolve the ambiguity in their location estimation. The only case a node receives beacons from only one segment is shown in Fig. 5(c). For this to happen, a node has to be very close to one segment of the robot trajectory, closer than 5 m. But in this case, all signal strength–distance mappings from beacons of that segment are very accurate, since the signal strength is quite large, and the localization error still remains small. Thus, again the distances from which nodes receive beacons define the larger localization error. For sensors in the worst possible positions, the closest distance from which they can receive beacons is 22.5 m for SCAN, and 38.1 m for HILBERT. Thus, again SCAN outperforms HILBERT.

The CDFs of the localization error for the three trajectories show an interesting behavior. In Fig. 6(b) we observe that the CDF for the error with DOUBLE SCAN initially is better than the CDFs for the other two trajectories. But there is a threshold of about 5 m, above which it becomes worse than the other two. The intuition behind this behavior is as follows: DOUBLE SCAN can localize most nodes with high accuracy, higher than SCAN and HILBERT, although it is performed with a double resolution, because it still gets advantage of the 2D scanning. But there is a small percentage of nodes (about 2% or 30 nodes) which cannot receive beacons from both scannings (along x- and y-axis), and the error for them is larger than 10 m. Even worse, we found out that the maximum error for DOUBLE SCAN can go up to 45 m, while it remains around 5 m for the other two trajectories. Fig. 6(b) also shows that the CDFs for SCAN and HILBERT are almost identical, with SCAN having a slightly better curve, which justifies its better performance.

We repeated again the simulation with random movement for the robot. The result was about 17% unlocalized nodes, and average localization error of about 6.6 m.

7.1.3. Resolution 60 m

The results for the localization error are shown in Fig. 7(a). The resolution here is much larger than the transmission range and this increases the error more than 100% for all the three trajectories. Compared to the 45 m resolution, the total localization error becomes almost double for HILBERT, 3.5 times larger for SCAN, and 4.5 larger times for DOUBLE SCAN.

An important observation here is that, with SCAN and the 60m resolution, about 66% of the nodes are able to receive beacons from only one line segment (one direction), as opposed to the previous two resolutions. The explanation for this is shown in Fig. 5(d). In this figure, we show three segments of the SCAN trajectory, namely kk', ll', and mm', and the distance between kk' and ll', as well as between ll' and mm' is 60 m, equal to the selected resolution. Any node located in the shaded area, around ll', is able to receive beacons only from points of segment ll',

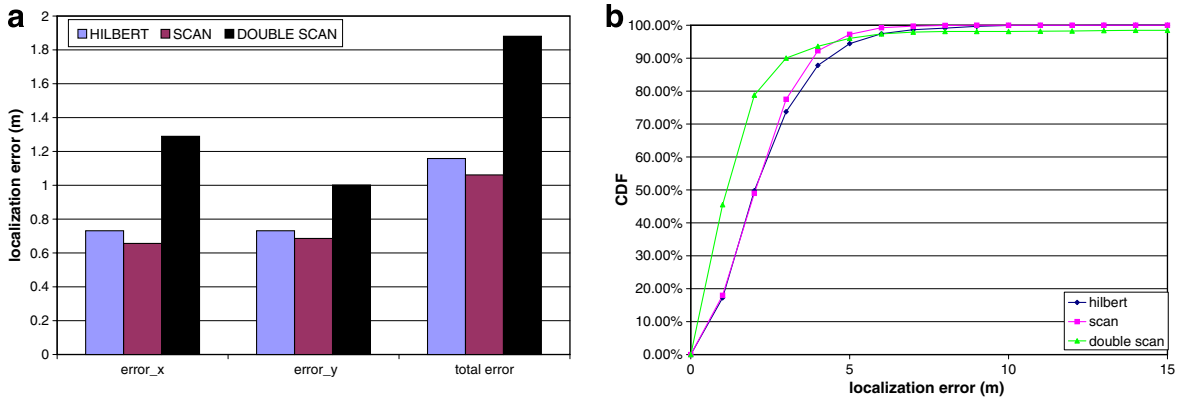

Fig. 6. Localization error and its CDF for resolution 45 m. (a) Localization error. (b) CDF of localization error.

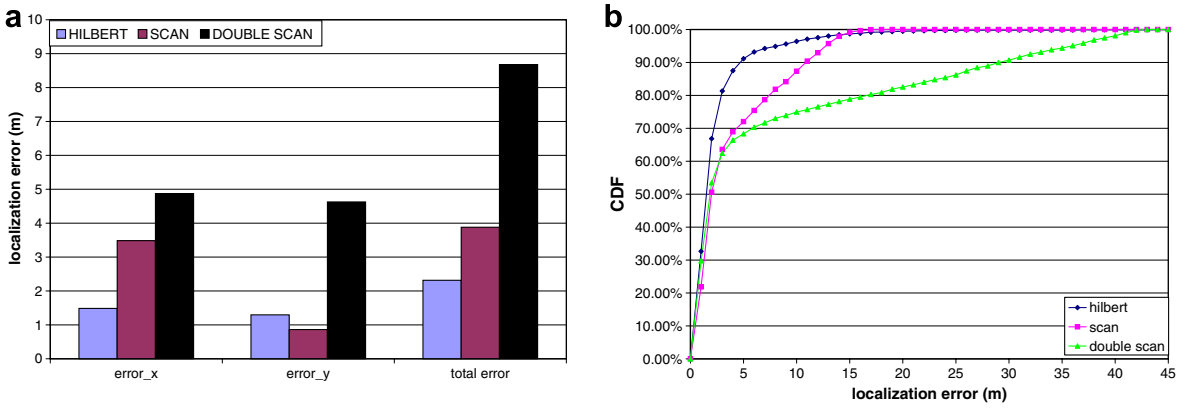

Fig. 7. Localization error and its CDF for resolution 60 m. (a) Localization error. (b) CDF of localization error.

because its distance from the other two segments is larger than the transmission range of 40 m. To generalize, for each of the trajectory segments parallel to the y axis, nodes located in a zone of 40 m around this zone (20 m on each side) only receive beacons from that segment, since their distance from the next segment is larger than 40 m. The total area covered by these zones is $15 \times 40 \times 900$ m $=$ 540,000 m, which gives a percentage of 66% if it is divided by the total area of 900 m $\times$ 900 m.

The result for nodes located in those zones is an increase in the error along x axis, since they cannot decide on which side of the trajectory they are, as shown in Fig. 5(e). In this figure, a node located at point A, receives three collinear beacons from locations B, C, and D. The intersection of the three circles with centers at B, C, and D (the three constraints imposed on node A's position estimate) gives two possible locations for node A, namely A and A′. There is no way for node A to reject one of the two locations, thus its estimated location as the average of these two locations is only accurate for the direction along y-axis, while it contains a large error for the direction along x-axis.

This disparity between the errors in the x and y axis can be seen in Fig. 7(a), in which we observe that the error along x axis for SCAN is 4 times larger than the error along y axis, while for the other two trajectories the error is similar along the two axes. With HILBERT trajectory, nodes can always receive beacons from at least two different line segments. With DOUBLE SCAN, which is performed with doubled resolution, only a percentage of the nodes can receive from at least two segments, but the rest of the nodes are equally likely to be close to a segment parallel to either x- or y-axis, hence on average, the error is similar along both axes.

Due to the large percentage of nodes receiving only collinear beacons in case of SCAN, the main factor that affects the localization error has been shifted from the distances between the sensors and the mobile landmark, to the collinearity of received beacons. As a result, the total localization error for SCAN is now about 68% larger than for HILBERT, and the latter has the best performance among the three trajectories. The distances between the mobile landmark and the sensors are important only in case of DOUBLE SCAN, which is performed with a very large resolution, three times larger than the transmission range. For this reason, the average error is quite larger for DOUBLE SCAN, compared to the other two trajectories. Due to the same reason, about 6% of the nodes (140–150 nodes) cannot receive any beacon.

The CDFs for the localization error with HILBERT and SCAN, shown in Fig. 7(b), show a drawback for the HILBERT

trajectory. In this figure, we observe that the CDF is better for HILBERT in its largest part, but it becomes worse for the largest error values. More specifically, there is a threshold (15 m error), in which the two CDFs are intersecting. This is because the maximum error with the HILBERT trajectory is about 40 m, similar to the maximum error for DOUBLE SCAN, while it is only 17 m for SCAN.

The reason for which HILBERT has much larger maximum error compared to SCAN, is that the latter offers more uniformity in coverage, compared to the former. With HILBERT trajectory and this large resolution, there are some pathological cases, for nodes located in the "holes" created by the HILBERT Curve. One such case is shown in Fig. 5(f). In this figure, node A is located at the center of a square with an edge of 60 m, and the four points of the robot trajectory closest to it are the vertices B, C, D, and E of the square. Although we selected the beacon transmission rate in order to make sure that the robot will always transmit when located at points such as B, C, D, and E (see Section 6.2), the distance of node A to each of these four points is 42.5 m, larger than the transmission range. Under a perfect channel, node A would not be able to receive any beacon. Because of randomness in the signal propagation caused by fading, reflections, scatterings etc., nodes like A actually can receive some of those four beacons, but the large distance implies large errors in the PDF Table mapping, and hence large localization error.

However, only a very small percentage of sensors have very large errors (only 2% or 50 nodes out of 2640 have an error larger than 15 m), while the average error is much lower for HILBERT trajectory, as we explained above, because most nodes can receive beacons from at least two directions. Hence, on average HILBERT offers the highest accuracy under a coarse trajectory.

Fig. 8 shows the real positions of the sensors, the position estimates and the robot HILBERT trajectory for one of the scenarios. For the sake of clarity we do not show the whole 900 m × 900 m area but only a part of it. In this figure, we observe that most of the position estimates are very close to the real positions of the sensors, but we can also see two of the special cases we described above, where the localization error increases. First, in the area adjacent to the point with coordinates (200, 225) there are three sensors which can only receive collinear beacons, when the robot moves on the line segment on their left side. As we explained in Fig. 5(e), these sensors cannot determine on which side of the line they really lie and they erroneously place themselves on this line. Second, the four points with coordinates (300, 180), (300, 240), (360, 240), and (360, 180) define a "hole", as described in Fig. 5(f). The sensor node with coordinates (325, 220) is within this hole (although not exactly at the center) and the localization error is large, because it can only receive a few beacons when the robot travels near the upper left corner of the hole.

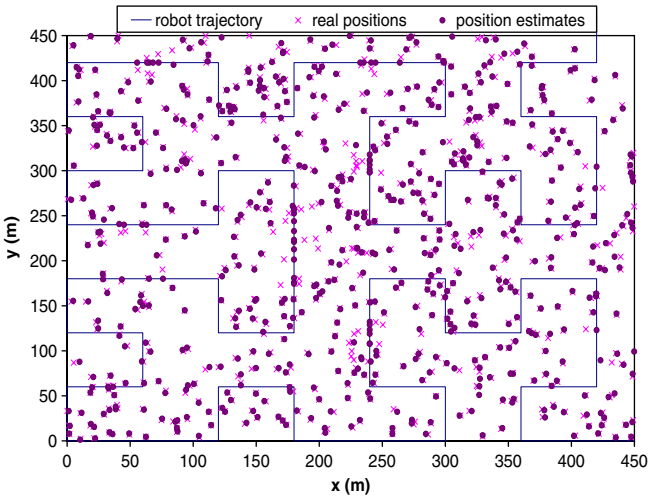

Fig. 8. The real and estimated positions of the sensors in the 900 × 900 m area, and the trajectory of the robot (HILBERT). For clarity, only a quarter of the area is shown.

In case of random robot movement, 31% of the nodes were unlocalized and the average error for the rest of them was 7.3 m, which is smaller than the error for DOUBLE SCAN, but much larger than the error for HILBERT and SCAN.

7.1.4. Summary and findings

In summary, a carefully selected deterministic trajectory can significantly decrease the localization error, compared to a random one, and also guarantee that all sensors will obtain position estimates. The comparison between three deterministic trajectories shows that when the resolution of the trajectory is smaller than or similar to the transmission range, SCAN gives slightly smaller error compared to HILBERT. But when the resolution is much larger than the transmission range, HILBERT clearly outperforms SCAN. Hence HILBERT is useful in cases where we want to traverse the deployment area with a very coarse granularity, due to energy/time limitations or surface properties (e.g., many obstacles may not allow a very dense trajectory). DOUBLE SCAN can give the best performance among the three trajectories, but with the cost of double total traveled distance. Hence, it is the best choice only for static networks, when the mobile landmark is sent only once and there are no time or energy constraints.

7.2. Multihop localization

As mentioned in Section 5, it is not always realistic to assume that sensors are static. Sensor nodes may have moderate mobility due to wind, animals etc. For such scenarios, *2-hop localization* along with a coarser resolution could be beneficial, depending on which of the two sources of error introduced in Section 5 most affects the total localization error.

In this section we examine the tradeoffs of *2-hop localization* in scenarios with mobile sensors. For this, we vary the sensor velocity and the resolution of the mobile landmark trajectory. In this way, we change the amount by

which each of the two sources of error contributes to the total localization error. We compare *2-hop localization* with *1-hop localization*. The former is the case where sensors which have received at least 80% of their total beacons from the robot, also transmit beacons. The latter is the basic case, where all the sensors use only beacons sent by the mobile landmark to estimate their positions. Since we are interested in traversing the network with a coarse resolution, we use the HILBERT space filling curve as the robot trajectory, which has been shown in Section 7.1 to give the smallest localization error when the resolution coarsens.

Figs. 9(a)–(c) show the average localization error over time when the sensor velocity is 1 m/ h, 30 m/ h, and 60 m/ h, respectively, for *1-hop* and *2-hop localization*, in a $900\text{ m} \times 900\text{ m}$ area with 2640 mobile sensors. The selected resolution is 60 m for *1-hop localization* and 120 m for *2-hop localization*. The latter is three times larger than the communication range, and many sensors cannot receive any beacons from the mobile landmark. It takes 7680 s for the mobile landmark to traverse the whole network once with *1-hop localization* and 3840 s with *2-hop localization*. Thus, the three figures show the average localization error over time for a time interval equal to three or six localization epochs, respectively. Note also that we do not plot the error during the first localization epoch of *1-hop localization* (0–7680 s), since in this period there are nodes which have not yet obtained an estimate for their positions.

7.2.1. Sensor velocity 1 m/ h

In this case, the sensors have very low mobility. In one epoch, a sensor may travel a maximum distance of 2.1 m on the same direction, which implies that the localization error for a sensor in one epoch cannot increase more than 2.1 m, because of its movement. Hence, the main source of error with this velocity is the localization error incurred when the sensors execute the localization algorithm, rather than the movement of the sensors. Fig. 9(a) shows that *1-hop localization* significantly outperforms *2-hop localization*. The average localization error for *1-hop localization* has increased from 2.3 m in the static scenario of Fig. 7(a) to 3.5 m, but it remains lower than 5 m and, most important, almost constant over time. On the other hand, *2-hop localization* does not offer any benefit in this case, but it increases the error, which oscillates between 31 m and 38 m.

7.2.2. Sensor velocity 30 m/ h

In this case, the velocity of the sensors is quite large, and the two sources of error contribute almost equally to the total error. In each localization epoch, there are periods of time during which the mobile landmark localizes many sensors and the average localization error decreases, as well

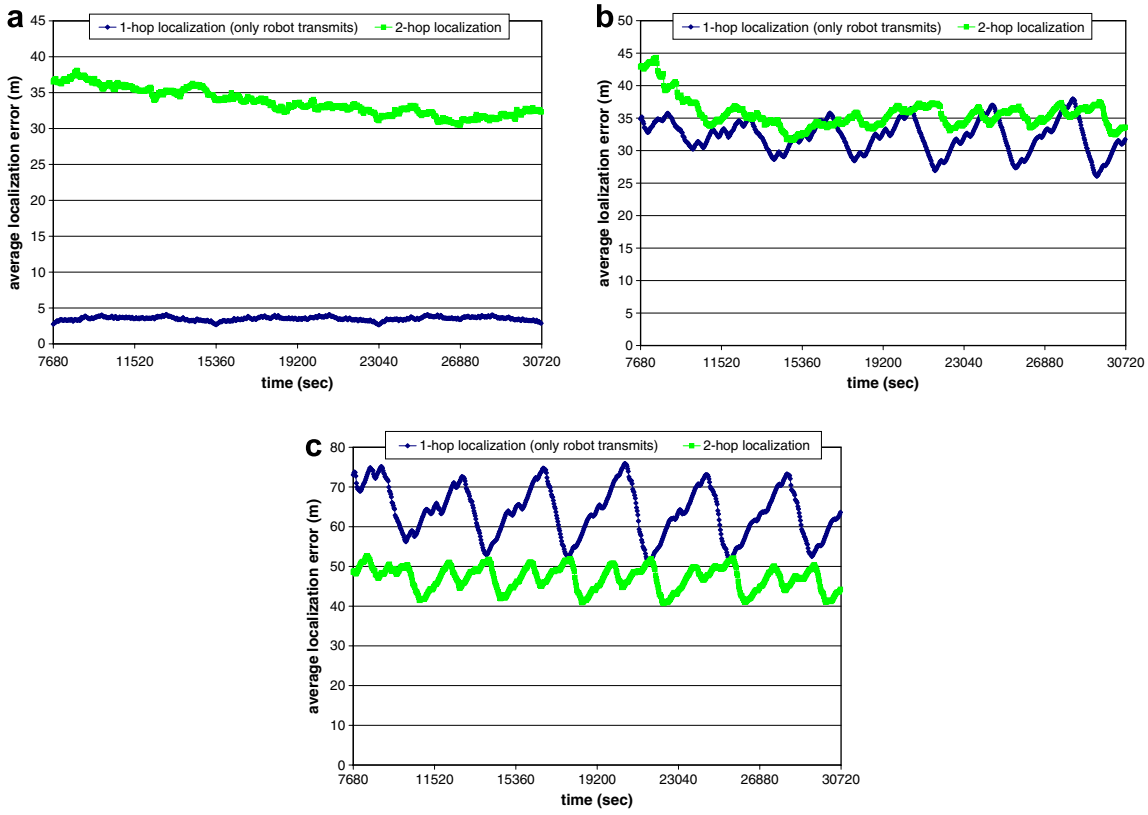

Fig. 9. Average localization error over time for three different sensor velocities. (a) Sensor velocity 1 m/h. (b) Sensor velocity 30 m/h. (c) Sensor velocity 60 m/h.

as periods of times during which the position estimates of many sensors become stale and the average localization error increases. Since the mobile landmark repeats the same trajectory periodically, the average localization error over time also has some periodicity, as shown in Fig. 9(b). In this figure, we observe that the average localization error oscillates between 27 m and 36 m over time for *1-hop localization* and between 33 m and 37 m for *2-hop localization*. Although the average error over time is still smaller for *1-hop localization*, the difference now is very small and there are also periods of time where *2-hop localization* outperforms *1-hop localization*. With *2-hop localization*, sensors never get the chance to obtain a very accurate estimate, since the mobile landmark does not pass close to most of them, but their estimates never become too stale. This is reflected in the variation of the average localization error over time for the two curves of Fig. 9(b), which is about 10 m for *1-hop localization*, while it is only 4 m for *2-hop localization*.

7.2.3. Sensor velocity 60 m/ h

In this case, the sensors mobility is the main source of error, rather than the error caused by the localization algorithm. Fig. 9(c) shows that the average error for *2-hop localization* oscillates between 40 m and 52 m, while it oscillates between 52 m and 76 m for *1-hop localization*. Hence, *2-hop localization* clearly outperforms *1-hop localization*. On average *2-hop localization* reduces the localization error by about 40%. Also, the variation of the error with *2-hop localization* is only about 12 m, half of the variation observed with *1-hop localization*. Note that although an error of 40 m or 50 m seems very large compared to the error of 2 m or 3 m as seen in Section 7.1, this is the best we can achieve when sensors move. There are applications in which even this accuracy is useful (e.g., for geographic routing, animal monitoring.) Moreover, this accuracy is still achieved by using only one mobile robot, which is very efficient in terms of cost. The only work other than ours, that considers sensor mobility [14], achieves an error of about 25 m by using 1 mobile landmark per transmission range, which is translated to 160 mobile landmarks in an area of $900\text{ m} \times 900\text{ m}$ and transmission range of 40 m. This approach, although more accurate, requires many more mobile landmarks compared to ours.

8. Conclusions

In this paper, we have studied the problem of path planning for mobile landmarks to reduce localization error as well as the time spent on localizing the sensor network. Although several works have proposed various localization schemes, none of these works has studied the trajectory of the mobile landmark. In this paper, we studied three different deterministic trajectories for use by a mobile landmark in sensor network localization. We showed that by carefully selecting the parameters of a deterministic trajectory, we can reduce the error compared to the random movement trajectory and also guarantee that all the sensors will obtain a position estimate. Our performance results show that among the three trajectories, SCAN offers the best performance when the trajectory has fine resolution i.e., the average distance between the sensors and the trajectory is small. However, for resolutions that are larger than the transmission range, the HILBERT space-filling curve outperforms SCAN by about 68%. DOUBLE SCAN can obtain the lowest localization error, but at the cost of doubling the distance traveled by the mobile landmark compared to the other two trajectories.

We also studied the path planning problem in scenarios where sensors have moderate mobility. In such scenarios, we showed that the average localization error can be significantly reduced over time, by using a large trajectory resolution combined with *2-hop localization*, in which nodes that obtain a good estimate about their positions, help other nodes to localize themselves.

There are several avenues for further research in the area of path planning of mobile landmarks. (1) We are interested in determining how transmission power control can be used by the mobile landmark to increase the distance in which the Gaussian model for RSS values remains valid. This will allow the mobile landmark to travel along coarser trajectories while maintaining the same localization error. It is interesting to investigate the noise distributions of RF beacons when operating over special hardware that supports power control. (2) In this paper we evaluated the various trajectories using simulations. A testbed evaluation requires a large number of sensors, which are not currently available in our testbed. This is a focus of our future work. However, we believe that our conclusions will still be valid in a real world deployment since we used a quite realistic simulation environment. Moreover, our trajectories are practical and can be easily implemented in mobile robot platforms (e.g., [40]). (3) Most mobile robot control software can detect obstacles that arise in the planned path (e.g., using sonar), and dynamically adjust the robot's movement to travel around them. Subsequently, the robot can re-align with the planned path. Thus, we expect a close correlation between our simulation results and an evaluation on a real world testbed. However, we plan to study whether better trajectories can be designed given an obstacle map. (4) We plan to study the tradeoffs in using few cooperative robots as mobile landmarks instead of just one.

Acknowledgements

This work was supported in part by NSF Grants ANI-0338856 and IIS-0329061.

References

[1] E. Acar, H. Choset, Complete sensor-based coverage with extended-range detectors: a hierarchical decomposition in terms of critical points and voronoi diagrams, in: Proceedings of the 2001 IEEE/RSJ

International Conference on Intelligent Robots and Systems (IROS '01), 2001.
[2] P. Bahl, V.N. Padmanabhan, RADAR: an in-building RF-based user location and tracking system, in: Proceedings of IEEE INFOCOM, March 2000.
[3] T. Bially, Space–filling curves: their generation and their application to bandwidth reduction, IEEE Trans. Inform. Theory IT-15 (1969) 658–664.
[4] J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, J. Jetcheva,. A performance comparison of multi-hop wireless ad hoc network routing protocols, in: Proceedings of ACM MobiCom, October 1998.
[5] N. Bulusu, J. Heidemann, D. Estrin, GPS-less low cost outdoor localization for very small devices, IEEE Personal Commun. Mag. 7 (5) (2000) 28–34.
[6] Z. Butler, A. Rizzi, R. Hollis, Contact sensor-based coverage of rectilinear environments, in: Proceedings of IEEE International Symposium on Intelligent Control, 1999.
[7] A.R. Butz, Convergence with Hilbert's space filling curve, J. Comput. Syst. Sci. 3 (1969) 128–146.
[8] X. Cheng, D.C. Andrew Thaeler, Guoliang Xue, TPS: a time-based positioning scheme for outdoor wireless sensor networks, in: Proceedings of IEEE INFOCOM, March 2004.
[9] P. Corke, R. Peterson, D. Rus, Networked robots: flying robot navigation using a sensor net, in Proceedings of ISRR, October 2003.
[10] A. Galstyan, B. Krishnamachari, K. Lerman, S. Pattem, Distributed online localization in sensor networks using a moving target, in: Proceedings of IPSN, April 2004.
[11] T. He et al., Speed: A stateless protocol for real-time communication in sensor networks, in: Proceedings of ICDCS, May 2003.
[12] T. He, C. Huang, B.M. Blum, J.A. Stankovic, T. Abdelzaher, Range-free localization schemes for large scale sensor networks, in: Proceedings of MobiCom, September 2003.
[13] D. Hilbert, Über die stetige Abbildung einer Linie auf Flächenstück, Math. Ann.,1891, pp. 459–460.
[14] L. Hu, D. Evans, Localization for mobile sensor networks, in: Proceedings of ACM MobiCom, September 2004.
[15] Y.C. Hu, S.L. Johnsson, Data parallel performance optimizations using array aliasing, in: Algorithms for Parallel Processing, 1999.
[16] X. Ji, H. Zha, Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling, in: Proceedings of IEEE INFOCOM, March 2004.
[17] P. Krishnan, A.S. Krishnakumar, W.-H. Ju, C. Mallows, S. Ganu, A system for LEASE: system for location estimation assisted by stationary emitters for indoor RF wireless networks, in: IEEE INFOCOM, March 2004.
[18] C. Luo, S.X. Yang, Complete coverage path planning with designated starting and planning locations using a neural network paradigm, in: Proceedings of Eighth International Conference on Neural Information Processing, 2001.
[19] R. Nagpal, H. Shrobe, J. Bachrach, Organizing a global coordinate system from local information on an ad hoc sensor network, in: Proceedings of IPSN, April 2003.
[20] D. Niculescu, B. Nath, Ad hoc positioning system (APS), in: Proceedings of Globecom, November 2001.
[21] D. Niculescu, B. Nath, Ad hoc positioning system (APS) using AOA, in: Proceedings of IEEE INFOCOM, March 2003.
[22] D. Niculescu, B. Nath, Error characteristics of ad hoc positioning systems (APS), in: Proceedings of MobiHoc, May 2004.
[23] D. Niculescu, B. Nath, VOR base stations for indoor 802.11 positioning, in: Proceedings of MobiCom, September 2004.
[24] P.N. Pathirana, N. Bulusu, A.V. Savkin, S. Jha, Node localization using mobile robots in delay-tolerant sensor networks, IEEE Trans. Mobile Comput. 4 (2004) 285–296.
[25] N.B. Priyantha, H. Balakrishnan, E. Demaine, S. Teller, Anchor-free distributed localization in sensor networks, in: Proceedings of SenSys, November 2003.
[26] N.B. Priyantha, H. Balakrishnan, E. Demaine, S. Teller, Mobile-assisted localization in wireless sensor networks, in: Proceedings of IEEE INFOCOM, March 2005.
[27] N.B. Priyantha, A. Chakraborty, H. Balakrishnan, The cricket location-support system, in: Proceedings of MobiCom, September 2000.
[28] J. Proakis, Digital Communications, fourth ed., McGraw Hill, 2000.
[29] V. Ramadurai, M.L. Sichitiu, Localization in wireless sensor networks: a probabilistic approach, in: Proceedings of ICWN, June 2003.
[30] A. Savvides, C.-C. Han, M.B. Strivastava, Dynamic fine-grained localization in ad-hoc networks of sensors, in: Proceedings of ACM MobiCom, July 2001.
[31] Y. Shang, W. Ruml, Improved MDS-based localization, in: Proceedings of IEEE INFOCOM, March 2004.
[32] Y. Shang, W. Ruml, Y. Zhang, M. Fromherz, Localization from mere connectivity, in ACM Mobihoc, June 2003.
[33] M.L. Sichitiu, V. Ramadurai, Localization of wireless sensor networks with a mobile beacon, in: Proceedings of MASS, September 2004.
[34] M.L. Sichitiu, V. Ramadurai, P. Peddabachagari, Simple algorithm for outdoor localization of wireless sensor networks with inaccurate range measurements, in: Proceedings of ICWN, June 2003.
[35] R. Stoleru, J.A. Stankovic, Probability grid: a location estimation scheme for wireless sensor networks, in: IEEE SECON, 2004.
[36] USCG Navigation Conter GPS page. http://www.navcen.uscg.nil/gps/default.html, January 2000.
[37] F. Ye, H. Luo, J. Cheng, S. Lu, L. Zhang, A two-tier data dissemination model for large-scale wireless sensor networks, in: Proceedings of ACM MobiCom, September 2002.
[38] A. Zelinsky, R. Jarvis, J. Byrne, S. Yuta, Planning paths of complete coverage of an unstructured environment by a mobile robot, in: Proceedings of International Conference on Advanced Robotics, 1993.
[39] X. Zeng, R. Bagrodia, M. Gerla. Glomosim: a library for parallel simulation of large-scale wireless networks, in: Proceedings of PADS Workshop, May 1998.
[40] Activmedia robotics. http://www.activrobots.com/.

Dimitrios Koutsonikolas is currently a Ph.D. student in the School of Electrical and Computer Engineering at Purdue University, USA. Previously, he received a B.Engg. degree from the National Technical University of Athens (NTUA), Greece. His research interests include wireless ad hoc, mesh and sensor networks. He is a recipient of the Ross Fellowship, Purdue University, 2004, and a Tellabs Fellowship, Center for Wireless Systems and Applications (CWSA), Purdue University, 2006.

Saumitra M. Das is currently a Ph.D. candidate in the School of Electrical and Computer Engineering at Purdue University, USA. Previously, he received a MS degree from Carnegie Mellon University, USA and a B.Engg. degree from the University of Bombay, India. His research interests include cross-layer system design for multi-hop wireless networks, scalable routing strategies in wireless ad hoc networks, and mobile robotics.

Y. Charlie Hu is an Associate Professor of Electrical and Computer Engineering and Computer Science at Purdue University. He received his M.S. and M.Phil. degrees from Yale University in 1992 and his Ph.D. degree in Computer Science from Harvard University in 1997. From 1997 to 2001, he was a research scientist at Rice University. Dr. Hu's research interests include operating systems, distributed systems, Internet routing and measurement, and wireless networking. He has published over 100 papers in these areas. Dr. Hu received the NSF CAREER Award in 2003. He served as a TPC Vice Chair for ICDCS 2007 and the 2004 International Conference on Parallel Processing and a co-founder and TPC co-chair for the International Workshop on Mobile Peer-to-Peer Computing. Dr. Hu is a member of USENIX and ACM and a senior member of IEEE.