Range-Free Localization and Its Impact on Large Scale Sensor Networks

TIAN HE, CHENGDU HUANG, BRIAN M. BLUM, JOHN A. STANKOVIC, and TAREK F. ABDELZAHER
University of Virginia

With the proliferation of location dependent applications in sensor networks, location awareness becomes an essential capability of sensor nodes. Because coarse accuracy is sufficient for most sensor network applications, solutions in range-free localization are being pursued as a cost-effective alternative to more expensive range-based approaches. In this paper, we present APIT, a novel localization algorithm that is range-free. We show that our APIT scheme performs best when an irregular radio pattern and random node placement are considered, and low communication overhead is desired. We compare our work, via extensive simulation, with three state-of-the-art range-free localization schemes to identify the preferable system configurations of each. In addition, we provide insight into the impact of localization accuracy on various location dependent applications and suggestions on improving their performance in the presence of such inaccuracy.

Categories and Subject Descriptors: C.2. [**Computer Communication Networks**]: Network Protocols

General Terms: Algorithms, Performance, Design

Additional Key Words and Phrases: Localization, positioning, location discovery, sensor network

1. INTRODUCTION

Sensor networks have been proposed for various applications, including search and rescue, disaster relief, target tracking, and smart environments. The inherent characteristics of these sensor networks make a node's location an important part of their state. For such networks, location is being used to identify the location at which sensor readings originate (for example, identifying a target's position during tracking, providing the location of an earthquake survivor

This work was supported in part by NSF grant CCR-0098269, the MURI award N00014-01-1-0576 from ONR, and the DAPRPA IXO office under the NEST project (grant number F336615-01-C-1905).

Authors' addresses: T. He, Department of Computer Science and Engineering, University of Minnesota; B. M. Blum and J. A. Stankovic, Department of Computer Science, University of Virginia; C. Huang and T. F. Abdelzaher, Department of Computer Science, UIUC; email: tianhe@cs.umn.edu.

buried underneath rubble). It is also used in communication protocols that route to geographical areas instead of IDs ([Hong et al. 2002; Karp and Kung 2000; Ko and Vaidya 1998; Xu et al. 2001]), and in other location-based services, such as sensing coverage [Yan et al. 2003] and location directory service [Li et al. 2000]. In addition to the applications and protocols mentioned, continued research in WSNs will serve to invent and identify many additional protocols and applications, many of which will likely depend on location-aware sensing devices.

Many localization algorithms for sensor networks have been proposed to provide per-node location information. With regard to the mechanisms used for estimating location, we divide these localization protocols into two categories: *range-based* and *range-free*. The former is defined by protocols that use absolute point-to-point distance estimates (range) or angle estimates for calculating location. The latter makes no assumption about the availability or validity of such information. Because of the hardware limitations of WSN devices, solutions in range-free localization are being pursued as a cost-effective alternative to more expensive range-based approaches.

This paper makes three major contributions to the localization problem in WSNs. First, we propose a novel range-free algorithm, called APIT, with enhanced performance under realistic system configurations. Second, although many different protocols [Bulusu et al. 2000; Nagpal 1999; Niculescu and Nath 2003b] have been proposed to solve the localization problem in a range-free context, no prior work has been done to compare them in realistic settings. This paper is the first to provide a realistic and detailed quantitative comparison of existing range-free algorithms to determine the system configurations under which each is optimized. We perform such a study to serve as a guide for future research. Third, no attempt has previously been made to broadly study the impact of location error on various location-dependent applications and protocols. This paper provides insight into the effect of localization accuracy on applications and suggestions on how to improve their performance in the presence of such inaccuracy.

The remainder of the paper is organized as follows: Section 2 discusses previous work in localization for sensor networks. Section 3 describes APIT. Section 4 gives brief descriptions of three other state-of-the-art range-free protocols to which we compare our work. Section 5 describes our simulation. Section 6 follows with a detailed performance comparison of the four range-free localization algorithms described. Section 7 further investigates the impact of localization error on various location-dependent applications and protocols, such as routing and target tracking. Finally, we discuss future work in Section 8 and conclude in Section 9.

2. STATE OF THE ART

Many existing systems and protocols attempt to solve the problem of determining a node's location within its environment. The approaches taken to solve this localization problem differ in the assumptions that they make about their respective network and device capabilities. These include assumptions about device hardware, signal-propagation models, timing and energy requirements, network makeup (homogeneous vs. heterogeneous), the nature of the environment (indoor vs. outdoor), node or beacon density, time synchronization of devices, communication costs, error requirements, and device mobility. In this section, we discuss prior work in localization with regard to these characteristics. We divide our discussion into two subsections where we present both range-based and range-free solutions.

2.1 Range-Based Localization Schemes

Time of Arrival (TOA) technology is commonly used as a means of obtaining range information via signal-propagation time. The most basic localization system to use TOA techniques is GPS [Wellenhoff et al. 1997]. GPS systems require expensive and energy-consuming electronics to precisely synchronize with a satellite's clock. With hardware limitations and the inherent energy constraints of sensor network devices, GPS and other TOA technology present a costly solution for localization in wireless sensor networks.

The Time Difference of Arrival (TDOA) technique for **ranging** (estimating the distance between two communicating nodes) has been widely proposed as a necessary ingredient in localization solutions for wireless sensor networks. While many infrastructure-based systems have been proposed that use TDOA [Bahl and Padmanabhan 2000; Harter et al. 1999; Priyantha et al. 2000], additional work such as AHLos [Savvides et al. 2001, 2002] has employed such technology in infrastructure-free sensor networks. Like TOA technology, TDOA also relies on extensive hardware that is expensive and energy consuming, making it less suitable for low-power sensor network devices. In addition, TDOA techniques using ultrasound require dense deployment (numerous anchors distributed uniformly) as ultrasound signals usually only propagate 20–30 feet.

To augment and complement TDOA and TOA technologies, an Angle of Arrival (AOA) technique has been proposed that allows nodes to estimate and map relative angles between neighbors [Niculescu and Nath 2003b]. Similar to TOA and TDOA, AOA estimates require additional hardware too expensive to be used in large-scale sensor networks.

Received Signal Strength Indicator (RSSI) technology such as RADAR [Bahl and Padmanabhan 2000] and SpotOn [Hightower and Boriello 2001] has been proposed for hardware-constrained systems. In RSSI techniques, either theoretical or empirical models are used to translate signal strength into distance estimates. For RF systems [Bahl and Padmanabhan 2000; Hightower and Boriello 2001], problems such as multipath fading, background interference, and irregular signal propagation characteristics (shown in an empirical study of this technology [Ganesan et al. 2002]) make range estimates inaccurate. Work to mitigate such errors, such as robust range estimation ([Girod and Estrin 2001]), two-phase refinement positioning ([Savarese et al. 2002; Savvides et al. 2002]), and parameter calibration ([Whitehouse and Culler 2002]) have been proposed to take advantage of averaging, smoothing, and alternate hybrid techniques to reduce error to within some acceptable limit. While solutions based on RSSI have demonstrated efficacy in simulation and in a controlled laboratory environment, the premise that distance can be determined based on signal strength, propagation patterns, and fading models remains questionable, creating a demand for alternate localization solutions that work independent of this assumption.

2.2 Range-Free Localization Schemes

In sensor networks and other distributed systems, errors can often be masked through fault tolerance, redundancy, aggregation, or by other means. Depending on the behavior and requirements of protocols using location information, varying granularities of error may be appropriate from system to system. Acknowledging that the cost of hardware required by range-based solutions may be inappropriate in relation to the required location precision, researchers have sought alternate range-free solutions to the localization problem in sensor networks. These range-free solutions use only regular radio modules as basics for localization; hence, they do not incur any additional hardware cost.

In Bulusu et al. [2000], a heterogeneous network containing powerful nodes with established location information is considered. In this work, anchors beacon their position to neighbors that keep an account of all received beacons. Using this proximity information, a simple centroid model is applied to estimate the listening nodes' location. We refer to this protocol as the *Centroid algorithm.*

An alternate solution, DV-HOP [Niculescu and Nath 2003a] assumes a heterogeneous network consisting of sensing nodes and anchors. Instead of single-hop broadcasts, anchors flood their location throughout the network maintaining a running hop count at each node along the way. Nodes calculate their position based on the received anchor locations, the hop count from the corresponding anchor, and the average distance per hop—a value obtained through anchor communication. Like DV-Hop, an *Amorphous Positioning* algorithm proposed in [Nagpal 1999] uses offline hop-distance estimations, improving location estimates through a neighbor-information exchange.

These range-free techniques are described in more depth in Section 4 and are used in our analysis for comparison with our work.

3. APIT LOCALIZATION SCHEME

In this section, we describe our novel area-based range-free localization scheme, which we call APIT. APIT requires a heterogeneous network of sensing devices where a small percentage of these devices (percentages vary depending on network and node density) are equipped with high-powered transmitters and location information obtained via GPS or some other mechanism. We refer to these location-equipped devices as **anchors**. Using beacons from these anchors, APIT employs a novel *area-based* approach to perform location estimation by isolating the environment into triangular regions between beaconing nodes (Figure 1). A node's presence inside or outside of these triangular regions allows a node to narrow down the area in which it can potentially reside. By utilizing combinations of anchor positions, the diameter of the estimated area in which a node resides can be reduced to provide a good location estimate.

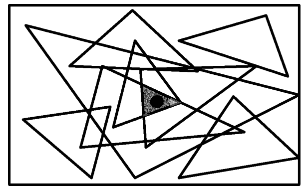


Fig. 1. Area-based APIT algorithm overview.

3.1 Main Algorithm

The theoretical method used to narrow down the possible area in which a target node resides is called the Point-In-Triangulation Test (PIT). In this test, a node chooses three anchors from all audible anchors (anchors from which a beacon was received) and tests whether it is inside the triangle formed by connecting these three anchors. APIT repeats this PIT test with different audible anchor combinations until all combinations are exhausted or the required accuracy is achieved. At this point, APIT calculates the center of gravity (COG) of the intersection of all of the triangles in which a node resides to determine its estimated position.

The APIT algorithm can be broken down into four steps: (1) beacon exchange, (2) PIT testing, (3) APIT aggregation, and (4) COG calculation. These steps are performed at individual nodes in a purely distributed fashion. Before providing a detailed description of each of these steps, we first present the basic pseudocode for our algorithm:

Receive location beacons (Xi,Yi) from N anchors.

InsideSet $= \Phi$ // *the set of triangles in which I reside*

For (*each triangle Ti* $\in \binom{N}{3}$ *triangles*) {

If (*Point-In-Triangle-Test* (*Ti*) == *TRUE*)

InsideSet = *InsideSet* $\cup$ {*Ti*}

If (*accuracy*(*InsideSet*) > *enough*) *break;*

},

/* *Center of gravity* (*COG*) *calculation* */

Estimated Position = *COG* ($\cap$*Ti* $\in$ *InsideSet*);

We note that the size of InsideSet grows cubically with the number of anchor beacons heard. For example, with 30 audible beacons in a sensor network of 1500 nodes, the radio region will be divided by 4060 triangles into small pieces. If the PIT tests render correct inside/outside decisions, each decision will narrow down the area in which a target node can possibly reside, making the final error small. In the next two sections, we describe the perfect PIT test and discuss the infeasibility of performing this test in a WSN. We then introduce a practical approximation to this perfect PIT test, applicable to our work.

3.2 Perfect PIT Test

In this section, we provide a perfect, albeit theoretical, solution to the following problem: For three given anchors: $A(a_x, a_y)$, $B(b_x, b_y)$, $C(c_x, c_y)$, determine whether a point M with an unknown position is inside triangle $\triangle$ABC or not.

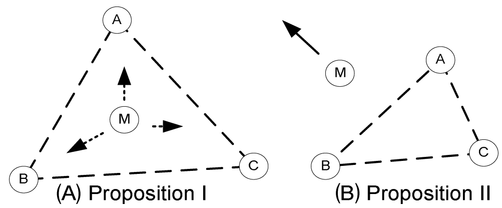


Fig. 2. Propositions I and II.

***Proposition I*:** *If M is inside triangle $\triangle ABC$, when M is shifted in any direction, the new position must be nearer to (further from) at least one anchor A, B or C* (Figure 2(A))

***Proposition II*:** *If M is outside triangle $\triangle ABC$, when M is shifted, there must exist a direction in which the position of M is further from or closer to all three anchors A, B, and C* (Figure 2(B)).

Propositions I and II are intuitively correct (the formal proofs are in He et al. [2003a]). Accordingly, the Perfect PIT test methodology derived from propositions I and II is as follows:

***Perfect PIT Test Theory* :** *If there exists a direction such that a point adjacent to M is further/closer to points A, B, and C simultaneously, then M is outside of $\triangle ABC$. Otherwise, M is inside $\triangle ABC$.*

The Perfect PIT test is guaranteed to be correct in deciding whether a point M is inside triangle $\triangle$ABC. However, there are two major issues when performing this in a WSN:

- How does a node recognize directions of departure from an anchor without moving?
- How to exhaustively test all possible directions in which node M might depart/approach vertexes A, B, and C simultaneously?

We address these issues in the next section.

3.3 Approximation of the Perfect PIT Test

The Perfect PIT test is infeasible in practice; however, we can still obtain a very high level of accuracy by an approximation method introduced in this section.

3.3.1 *Departure Test.* In previous work [Bahl and Padmanabhan 2000; Hightower et al. 2000], researchers have assumed a circular, or otherwise well-defined, mathematical or empirical model such as a log-normal attenuation model for radio propagation characteristics that describes the relationship between the signal strength degradation and the distance a radio signal travels. However, according to a recent empirical study by D. Ganesan at UCLA [Ganesan et al. 2002], this assumption does not hold well in practice. In our work, we make a much weaker assumption about radio propagation characteristics. We assume that in a certain propagation direction, defined to be within a narrow angle from the sending anchor (Figure 3), the received signal strength

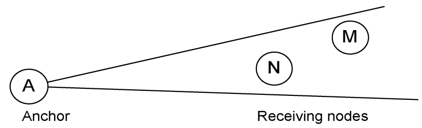


Fig. 3. Departure test.

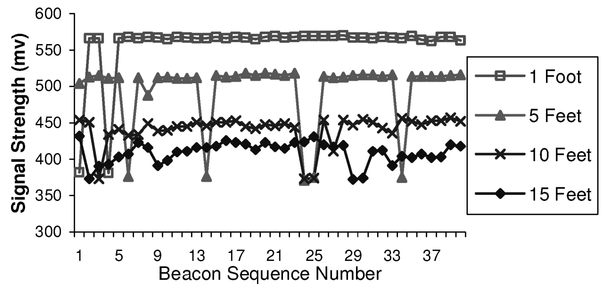


Fig. 4. Signal strength at different distances.

is monotonically decreasing in an environment without obstacles. This simply says that in a given direction, the further away a node is from the anchor, the weaker the received signal strength will be. Through signal strength comparisons between neighboring nodes, this assumption allows a node to determine whether a neighboring node is closer to a given anchor.

***Departure Test Definition*:** *Test whether M is further away from anchor A than N* (Figure 3).

In addition to gathering evidence drawn from prior empirical studies of WSNs [Ganesan et al. 2002], we checked the validity of our assumption on Berkeley's MICA mote [Crossbow] testbed in an obstruction-free laboratory environment. In this experiment, we incrementally increased the distance between sending (anchor) and receiving motes. Figure 4 shows the measured signal strength of 40 beacons from a single anchor at varying distances.

We conclude from Figure 4 that our assumption of monotonically decreasing signal strength in a given direction is usually valid. For example, the signal strength readings shown in Figure 4 are usually about 560 mv at 1-foot, and about 510 mv at 5-feet. However, we note that there are various points on the graph where this signal strength property is violated because of burst disturbance effects. Two approaches to minimize the effect of such disturbances include taking a running average of the signal strength over time and using our robust aggregation, a technique discussed further in Section 3.4.

It should be noted that our scheme does not make any assumptions about the correlation between *absolute* distance and signal strength; hence, we consider our scheme a range-free solution. More importantly, although we use radio signal comparisons throughout the paper, our scheme can actually work with any system, so long as it can support a form of the departure test, for example, by using the hop counts.

3.3.2 *Approximate PIT Test.* To perform PIT testing in sensor networks without requiring that nodes move, we define an Approximate PIT Test (APIT)

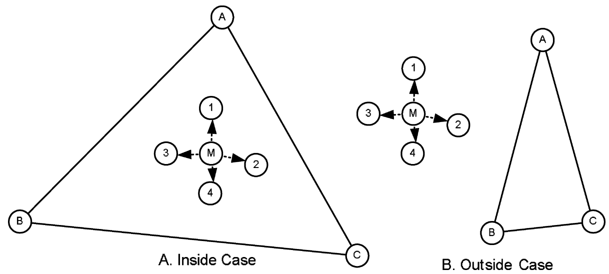


Fig. 5. Approximate PIT test.

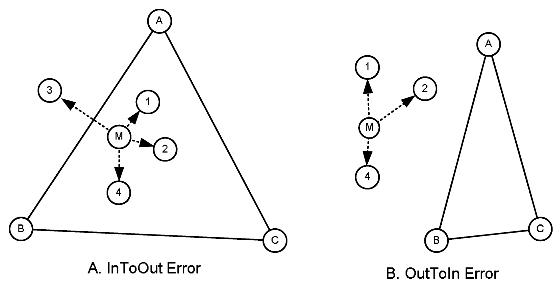


Fig. 6. Error scenarios for the APIT test.

that takes advantage of the relatively high node density of these networks (usually with connectivity above 6). The basic idea behind the APIT test is to use neighbor information, exchanged via beaconing, to emulate the node movement in the Perfect PIT test. The APIT test is formally described below.

Approximate PIT Test**:** *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* $\triangle ABC$. *Otherwise, M assumes it resides outside this triangle*.

We further explain the APIT test through an example. Figure 5(A) presents a scenario where none of M's neighbors, 1, 2, 3, or 4, is further from/closer to all three anchors A, B, and C simultaneously. In this scenario, M will assume that it is inside the triangle $\triangle$ ABC according to the definition. The other scenario is shown in Figure 5(B), where neighbor 3 will report to node M that it is further away from A, B, and C than M. This allows M to assume it resides outside of triangle $\triangle$ ABC.

Because APIT can only evaluate a finite number of directions (the number of neighbors), APIT can make an incorrect decision. The two scenarios where incorrect decisions are made are depicted in Figure 6. In Figure 6(A), we show what we deem *InToOut* error, where the node is inside the triangle, but concludes, based on the APIT test, that it is outside the triangle. This can happen when M is near the edge of the triangle, while some of M's neighbors (3 in this case) are outside the triangle and further from all points ABC, in relation to node M. As a result, M mistakenly thinks it is outside of triangle ABC due to this **edge effect**. On the other hand, the **irregular placement** of neighbors can result in *OutToIn* error. Figure 6(B) depicts a scenario where M is outside

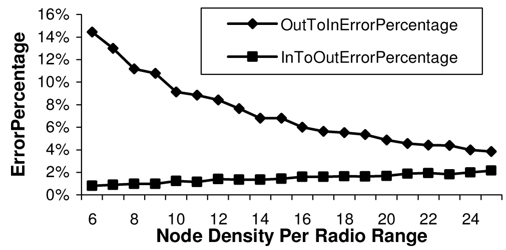


Fig. 7. APIT error under varying node densities.

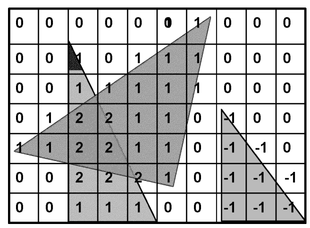


Fig. 8. SCAN approach.

of triangle ABC and none of its neighbors is further from/closer to all three anchors, A, B, and C, simultaneously. This makes M mistakenly assume it is inside triangle ABC.

Fortunately, from experimentation, we find that the percentage of APIT tests exhibiting such an error is relatively small (14% in the worst case). Figure 7 demonstrates this error percentage as a function of node density. When node density increases, APIT can evaluate more directions, considerably reducing *OutToInError* (Figure 6(B)). On the other hand, *InToOutError* will slightly increase due to the increased chance of **edge effects**.

3.4 APIT Aggregation

Once the individual APIT tests finish, APIT aggregates the results (inside/outside decisions among which some may be incorrect) through a grid SCAN algorithm (Figure 8). In this algorithm, a grid array is used to represent the maximum area in which a node will likely reside. In our experiments, the length of a grid side is set to 0.1R, to guarantee that estimation accuracy is not noticeably compromised.

For each APIT **inside decision** (a decision where the APIT test determines the node is inside a particular region) the values of the grid regions over which the corresponding triangle resides are incremented. For an **outside decision,** the grid area is similarly decremented. Once all triangular regions are computed, the resulting information is used to find the maximum overlapping area (e.g., the grid area with value 2 in Figure 8), which is then used to calculate the center of gravity for position estimation.

The pseudocode for APIT aggregation is as follows:

For (*each triangle* $T_i \in \binom{N}{3}$ *triangles*) {
  If (*APIT* (T_i) == *Out*) *AddNegativeTriangle*(T_i);

	(X,Y)		SS
A	20	20	1mv
B	45	31	2mv
C	23	56	3mv

Node M

	(X,Y)		SS
A	20	20	2mv
B	45	31	3mv
C	23	56	1mv

Node 1

Fig. 9. Table of heard anchors.

	(X,Y)		MySS	SS1	.....	SSn
A	20	20	1mv	2mv		6mv
B	45	31	2mv	3mv		7mv
C	23	56	3mv	1mv		7mv

Node M

Fig. 10. Combined table.

$If\,(APIT(T_i)\ == In)\, AddPositiveTriangle(T_i);$
}
Find the area with Max values;

APIT aggregation is a robust approach that can mask errors in individual APIT tests. As we know from Figure 7, the majority (more than 85%, in the worst case) of APIT tests are correct. With limited error, the correct decisions build up on the grid and the small number of errors only serves as a slight disturbance to the final estimation.

If the maximum range of an anchor node is known, we can filter out the grid points, which are out of range of any anchors heard by this node before we run SCAN algorithm. This leads to better localization accuracy and less memory requirement.

3.5 A Walk through the APIT Algorithm

In this section, we present an example to further explain our APIT algorithm.

1. Having received beacons from anchors A, B, and C, each node maintains a table (Anchor ID, Location, Signal Strength) for each anchor heard (Figure 9).
2. Each node beacons once to exchange anchor tables with its neighbors. These tables are merged at every node to maintain neighborhood state (Figure 10).
3. APIT runs on every column of the node's table to determine whether a neighboring node exists that has consistently larger/smaller signal strengths from the three anchors A, B, and C.[1] If such a neighbor is found, M assumes that it is outside triangle ABC. If no such neighbor is found, M assumes it is inside this region.
4. Each node repeats step 3 for varying combinations of three anchors. (Note: we only demonstrate one combination of three anchors in this example).

[1]No PIT test is performed when neighboring nodes do not share three common anchor points.

5. The algorithm described in Section 3.4 is then used to determine the area with maximum overlap.
6. Finally, the center of gravity of this area is used as the final location estimation.

3.6 APIT Performance Analysis

We consider a static senor network with N anchors and M nodes. Since APIT requires each anchor and node to broadcast once, the communication overhead of our APIT algorithm is $N+M$ under a collision-free situation. We have proved (see authors for proof) that if a target node can receive beacons from K anchors, the maximum number of polygons partitioned by these anchors can be achieved by placing all anchors on a convex curve. This anchor placement creates $(K-1)(K-2)/2+K(K-1)(K-2)(K-3)/24$ partitions. Assuming the nominal anchor radio range is R, the average size of each partition is then:

$$\frac{\pi R^2}{(K-1)(K-2)/2 + K(K-1)(K-2)(K-3)/24}$$

It should be noted that the above formula only indirectly reflects the upper bound performance of the Perfect PIT test. APIT has less accuracy due to approximation, as we will show in our evaluations.

By using our SCAN algorithm during APIT aggregation, we bound the computational complexity of the APIT algorithm by $O(L)$ (L is the number of APIT tests and each test only requires several comparisons). If we use a geometric algorithm to perform APIT aggregation precisely, the computational complexity will be $O(L^2)$. In order to perform SCAN algorithm, each node keeps bitmaps (Figure 8).

In a mobile sensor network, periodic beaconing is a straightforward solution to maintain the current anchor and node positions. A more sophisticated method to minimize localization cost under such a network is left as future work.

3.7 Key Observations

We note several key observations here to justify the use of our APIT algorithm in sensor networks.

- Redundancy and high node density are the key positive characteristics of sensor networks over traditional ad hoc networks. By exploiting this redundancy, aggregated decisions can provide good accuracy during location estimation, regardless of the fact that information obtained by an individual test is coarse and error prone.
- In order to obtain high redundancy without increasing deployment costs, we can use a single moving anchor that sends out beacons at different locations to localize all nodes inside a sensor network.

4. RANGE-FREE SCHEMES

In this section, we briefly describe the key features of three state-of-the-art range-free localization algorithms studied in our simulation. These algorithms are implemented in accordance with the published design and, with the exception of a few enhancements, made to ensure that our comparison is as fair as possible. The protocols discussed include:

- **Centroid Scheme** [Bulusu et al. 2000].
- **DV-Hop Scheme** [Niculescu and Nath 2003a].
- **Amorphous Scheme** [Nagpal 1999; Nagpal et al. 2003].

In addition to the aforementioned range-free algorithms, we implement an oracle version of APIT that uses the Perfect PIT Test defined in Section 6.2. For completeness, we provide brief descriptions of these algorithms. More details can be found in Bulusu et al. [2000], Nagpal [1999], and Nagpal et al. [2003].

4.1 Centroid Localization

N. Bulusu and J. Heidemann [Bulusu et al. 2000] proposed a range-free, proximity-based, coarse-grained localization algorithm, that uses anchor beacons, containing location information (X_i,Y_i), to estimate node position. After receiving these beacons, a node estimates its location using the following centroid formula:

$$(X_{est}, Y_{est}) = \left(\frac{X_1 + \cdots + X_N}{N}, \frac{Y_1 + \cdots + Y_N}{N}\right)$$

The distinguished advantage of this Centroid Localization scheme is its simplicity and ease of implementation. In a later publication [Bulusu et al. 2001], N. Bulusu augments her work by suggesting a novel density-adaptive algorithm (HEAP) for placing additional anchors to reduce estimation error. Because HEAP requires additional data dissemination and incremental beacon deployment, while other schemes under consideration only use ad hoc deployment, we do not include this later work in our simulations.

4.2 DV-Hop Localization

DV-Hop localization is proposed by D. Niculescu and B. Nath in the Navigate project [Niculescu and Nath 2001]. DV-Hop Localization uses a mechanism that is similar to classical distance vector routing. In this work, one anchor broadcasts a beacon to be flooded throughout the network containing the anchors location with a hop count parameter initialized to one. Each receiving node maintains the minimum counter value per anchor of all beacons it receives and ignores those beacons with higher hop count values. Beacons are flooded outward with hop count values incremented at every intermediate hop. Through this mechanism, all nodes in the network (including other anchors) get the shortest distance, in hops, to every anchor. The hop count for a single anchor A, generated by simulation, is shown in Figure 11.

In order to convert hop count into physical distance, the system estimates the average distance per hop without range-based techniques. Anchors perform this task by obtaining location and hop-count information for all other anchors inside the network. The average single-hop distance is then estimated by anchor

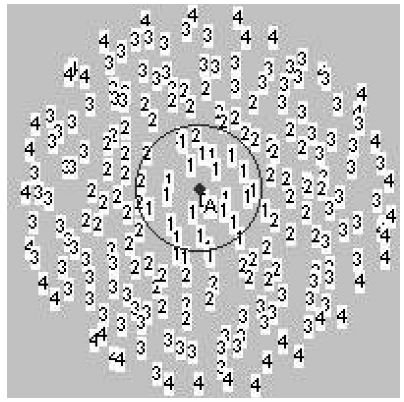


Fig. 11. Anchor beacon propagation phase.

i using the following formula:

$$HopSize_i = \frac{\sum \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum h_j}$$

In this formula, (x_j, y_j) is the location of anchor j, and h_j is the distance, in hops, from anchor j to anchor i. Once calculated, anchors propagate the estimated HopSize information out to the nearby nodes.

Once a node can calculate the distance estimation to more than 3 anchors in the plane, it uses triangulation (multilateration) to estimate its location. Theoretically, if errors exist in the distance estimation, the more anchors a node can hear the more precise localization will be.

4.3 Amorphous Localization

The Amorphous Localization algorithm [Nagpal 1999; Nagpal et al. 2003], proposed independently from DV-Hop, uses a similar algorithm for estimating position. First, like DV-Hop, each node obtains the hop distance to distributed anchors through beacon propagation.

Once anchor estimates are collected, the hop–distance estimation is obtained through local averaging. Each node collects neighboring nodes' hop distance estimates and computes an average of all its neighbors' values. One-half of the radio range is then deducted from this average to compensate for error caused by low resolution.

The Amorphous Localization algorithm takes a different approach from the DV-Hop algorithm to estimate the average distance of a single-hop. This work assumes that the density of the network, n_{local}, is known *a priori,* so that it can calculate HopSize offline in accordance with the Kleinrock and Silvester formula [Kleinrock and Silvester 1978]:

$$HopSize = r\left(1 + e^{-n_{local}} - \int_{-1}^{1} e^{-\frac{n_{local}}{\pi}\left(\arccos\ t - t\sqrt{1-r^2}\right)} dt\right)$$

Finally, after obtaining the estimated distances to three anchors, triangulation is used to estimate a node's location.

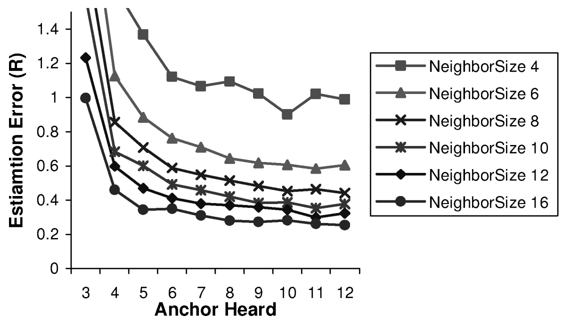


Fig. 12. Phase transition in the DV-based algorithm.

4.3.1 *Amorphous Localization Enhancement*[2]. By using only three anchors, Nagpal suggests in [1999] a critical minimum average neighborhood size of 15, imposed to obtain good accuracy. As shown in the APIT algorithm, increasing estimation redundancy reduces estimation error. We, therefore, argue that the same design philosophy can be applied to Nagpal [1999]. By increasing the number of anchors used in their estimation, we can effectively reduce the critical minimum average neighborhood requirement from 15 nodes per communication area, to 6, under uniform node placement (Figure 12) without reducing estimation accuracy (this number would be 8 for random node placement).

This enhancement uses work done by Jan Beutel [1999] in the Picoradio Project at UC Berkeley. A minimum mean square error (MMSE) algorithm triangulates node positions, based on the locations of multiple anchors (in this case more than 3), and associates distances between each anchor and the target node.

Using this enhancement, we show that the Amorphous algorithm can actually work in a sparsely connected network. Increasing the number of anchors participating in multilateration can dramatically reduce the required level of network connectivity. In Figure 12, we see that when three anchors are used, the estimation error (normalized to units of node radio range R) is large, regardless of the level of connectivity. By increasing the number of anchors to 5, we obtain better precision than that with three anchors, when the levels of connectivity as low as 6.

More importantly, Figure 12 shows two kinds of phase transitions that occur. First, when the neighbor size exceeds 8, increasing the number of anchors participating in multilateration brings down the estimation error below one-half of the radio range, a bound tolerated by the applications we studied in Section 7. Second, the estimation accuracy increases dramatically as the number of anchors heard increases to 6. However, after that, continuing to increase the number of anchors heard only slightly increases precision. In accordance with Figure 12, for DV-based algorithms, in order to confine the average estimation error to reside within one-half of the radio range, we suggest that both the

[2]A recent publication [Nagpal et al. 2003] in ISPN'03 makes a similar enhancement to the one we propose here.

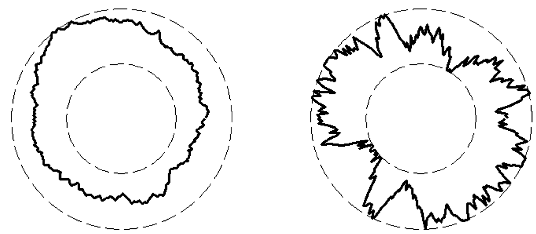


Fig. 13. Irregular radio pattern.

neighborhood size and the number of anchors used in multilateration, remain about 8 to 10. We argue that it is not quite cost-effective to further increase node density or the number of anchors used in multilateration for better accuracy after these phase transition points.

4.4 Perfect PIT Algorithm

As previously mentioned, the precision of our APIT algorithm is highly dependent on the correctness of the APIT Test. To obtain boundary conditions for a best-estimate in our localization scheme, we simulate a perfect PIT algorithm that utilizes an oracle. This oracle can guarantee correctness when determining whether a node resides within the triangular region created by the three anchors. We use this as a precise bound on our APIT algorithm.

5. SIMULATION SETTINGS

This section describes the simulation settings we use in our evaluation.

5.1 Radio Model

Some previous work in localization assumes that a perfect circular radio model exists. As stated before, empirical studies [Ganesan et al. 2002] on real testbeds have shown that this assumption is invalid for WSNs. To ensure that our evaluation is as true to reality as possible, we use a more general radio model in our evaluation. Specifically, we assume a model with an upper and lower bound on signal propagation (Figure 13). Beyond the upper bound, all nodes are out of communication range; within the lower bound, every node is guaranteed to be within communication range. If the distance between a pair of nodes is between these two boundaries, three scenarios are possible: (1) symmetric communication. (2) unidirectional asymmetric communication, and (3) no communication.

The parameter DOI is used to denote the irregularity of the radio pattern. It is defined as the maximum radio range variation per unit degree change in the direction of radio propagation. When the DOI is set to zero, there is no range variation, resulting in a perfectly circular radio model. To get a better idea of how this DOI parameter affects signal-propagation characteristics, Figure 13 shows the radio patterns generated in simulation with DOI values set to 0.05 and 0.2, respectively. To investigate how well our model resembles the reality in sensor motes, we measure the communication range of a MICA mote as

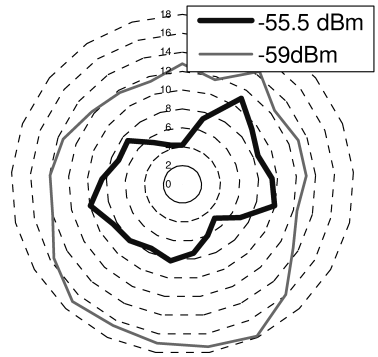


Fig. 14. Radio pattern from MICA2.

the receiver direction varies from 0 to 360 degrees. The two communication ranges are determined when received signal strength threshold is set to −55.5 and −59 dBm, respectively. The radio patterns are shown in Figure 14. These patterns give us the measured DOI values of 0.12 and 0.09 for two received signal thresholds, respectively.

5.2 Placement Model

In our simulations, nodes and anchors are distributed in a rectangular terrain in accordance with predefined densities. Two common placement strategies are investigated, namely random and uniform.

- Random placement: it distributes all nodes and anchors randomly throughout the terrain.
- Uniform placement: the terrain is partitioned into grids and nodes and anchors are evenly divided among these grids (random distribution inside each grid).

5.3 System Parameters

In our experiments, we study several system-wide parameters that we feel directly affect estimation error in range-free localization algorithms. A description of these parameters follows:

- Node density (ND): Average number of nodes per node radio area.
- Anchors heard (AH): Average number of anchors heard by a node and used during estimation.
- Anchor to node range ratio (ANR): The average distance an anchor beacon travels divided by the average distance traveled by a regular node signal. When this value equals one, the anchor and nodes have the same average radio range. The larger this value, the fewer anchors required to maintain a desired AH value.
- Anchor percentage (AP): The number of anchors divided by the total number of nodes (1000–3000 nodes). This value can be derived from the three parameters described above using the formula: $AP = AH/(AH + ND{*}ANR^2)$.

- Degree of irregularity (DOI): DOI is defined in Section 5.1 as an indicator of radio pattern irregularity.
- GPS error: In reality, GPS-equipped anchors will render imprecise readings. In our evaluation, this parameter is defined as the maximum possible distance from the real anchor position to the GPS estimated anchor position in units of node radio range (R).
- Placement: Random and uniform node/anchor placements are investigated in the evaluation.

In the evaluation, all distances including error estimation, are normalized to units of node radio range (R) to ensure generally applicable results.

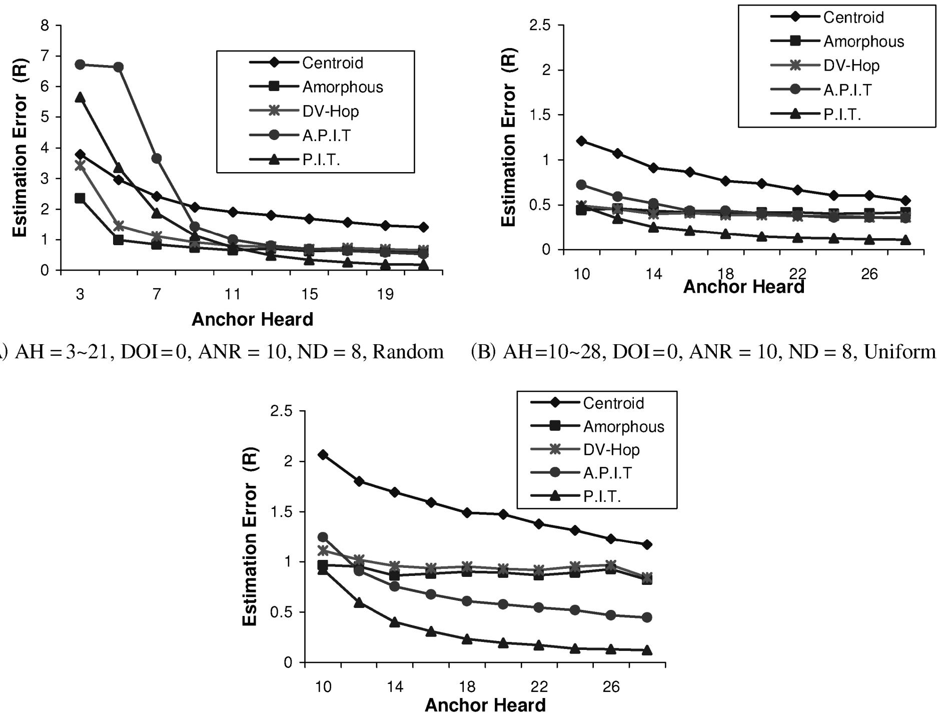
5.4 A Note about Comparisons

The range-free localization algorithms studied in this paper share a common set of system parameters and most of them are defined in a consistent way across the algorithms we analyze. However, due to different anchor beacon propagation methods utilized in different algorithms, the anchor to node range ratio (ANR) parameter varies between algorithms. In the Centroid and APIT algorithms, direct communication between anchors and **target nodes** (nodes attempting to determine their location) is used. In this case, ANR is set to the physical radio range ratio between anchor and target nodes. In the Amorphous and DV-Hop algorithms studied, the physical radio range of anchors is the same as that of target nodes and the ANR is set to the distance an anchor beacon can propagate in units of node radio range (R). In our evaluation, we indicate any performance implications that result from this implementation difference.

6. EVALUATION

This section provides a detailed quantitative analysis comparing the performance of the range-free localization algorithms described in Sections 3 and 4. The obvious metric for comparison when evaluating localization schemes is location estimation error. We have conducted a variety of experiments to cover a wide range of system configurations including varying (1) anchor density, (2) target node density, (3) radio range ratio (ANR), (4) radio propagation patterns, and (5) GPS error. Because communication can have a significant impact on sensor network systems with low bandwidth, we also use communication overhead, in terms of number of beacons exchanged, as a telling secondary metric to evaluate the cost and performance of the localization schemes studied.

Outside of studying the effect of certain parameters on localization error, we use default values of $AH = 16$, $ND = 8$, and $ANR = 10$ (anchor percentage = 2%) in most of our experiments. These settings are in line with our expectation of future sensor network technology and facilitate comparisons between figures. In all of our graphs, each data point represents the average value of 600 trials with different random seeds and the 90% confidence intervals for the data are within 5–10% of the mean shown. We note that for legibility reasons, we do not plot these confidence intervals in this paper. Full experimental data can be obtained from the authors upon request.



(A) AH = 3~21, DOI = 0, ANR = 10, ND = 8, Random (B) AH = 10~28, DOI = 0, ANR = 10, ND = 8, Uniform

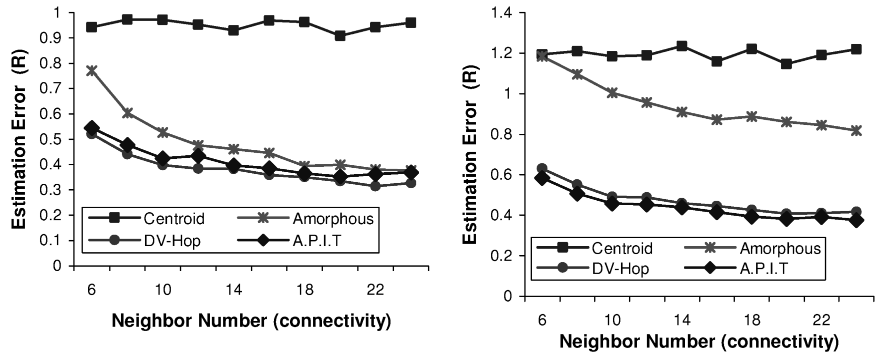
(C) AH = 10~28, DOI = 0, ANR = 10, ND = 8, Random

Fig. 15. Error varying AH.

6.1 Localization Error when Varying AH

In this experiment, we analyze the effect of varying the number of anchors heard (AH) at a node to determine its effect on localization error.

Figure 15(A) shows that the overall estimation error decreases as the number of anchors heard increases. However, it is important to note that different algorithms transition at different points in the graph. For example, the Amorphous and DV-Hop schemes improve rapidly when AH is below 7 and are nearly insensitive to the addition of anchors above 7. In contrast, the precision of APIT and the Centroid Localization scheme constantly improve as AH is increased (Figure 15(B) and 15(C)). Our APIT algorithm performs worse than the Centroid algorithm when AH is below 8, because of the fact that the diameter of the divided area is not sufficiently small. This effect is significantly reduced by increasing AH values. For larger AH values, APIT consistently outperforms the Centroid scheme. Figure 15(B) extends AH to higher values in order to show estimation error below 0.6 R. We note that our APIT algorithm requires only 12 anchors to reach the 0.6R level, while the Centroid scheme requires 24. Finally, Figure 15(C) presents the same experimental results for random node placement. By comparing graphs B (uniform placement) and C (random placement), we show that the DV-based algorithm is more sensitive to irregular node placement than both APIT and the Centroid scheme. This is mainly due



(A) DOI = 0.1, ANR = 10, AH = 16, Uniform (B)DOI = 0.2, ANR = 10, AH = 16, Uniform

Fig. 16. Error varying ND.

to the fact that *HopSize* estimation in the DV-Hop and Amorphous schemes, is less precise in nonisotropic deployment.

6.2 Localization Error when Varying ND

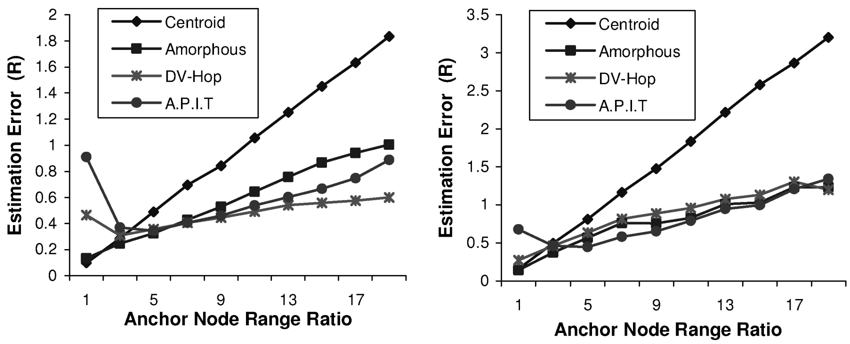
Figure 16 explores the effect of node density (ND) on the localization estimation accuracy. For all but the Centroid algorithm, localization error decreases as the number of neighbors increases. Since there is no interaction between nodes in the Centroid algorithm, we see nearly constant results while varying ND. However, due to its relatively simple design, the Centroid localization scheme does not perform as well as the others.

Because the offline estimation of *HopSize* in the Amorphous algorithm has a large error when the node density is small, the estimation error is large when the node density is below 10. APIT and DV-Hop, however, are robust to varying ND and produce good results as long as the neighbor density remains above 6. By comparing Figure 16A (DOI = 0.1) and Figure 16(B) (DOI = 0.2), we show that the DV-Based algorithms, especially the Amorphous algorithm, are more sensitive to irregular radio patterns than the APIT scheme. This is mainly due to the fact that *HopSize* estimation in the previous schemes is less precise in the presence of irregular radio patterns. However, it should be noted that DV-Hop abates this error by online estimation.

6.3 Localization Error when Varying ANR

Section 6.1 demonstrated that a large number of anchors are desired for good estimation results. The cost of having such a large percentage of anchors can be ameliorated by increasing the anchor radio range to which beacons travel. This happens because larger beacon propagation distances mean less anchors are required to achieve the same AH value. For example, if an algorithm requires AH equal to the neighborhood node density (ND), we need 50% of the nodes to be anchors when the ANR equals one. By increasing the ANR by a factor of 10, we can reduce the required anchor percentage to only 1%.

The implication of this solution, as shown in Figure 17, is that estimation error increases as ANR increases. This occurs because larger beacon propagation



(A) ND = 8, AD = 16, DOI = 0.1, Uniform (B) ND = 8, AD = 16, DOI = 0.1, Random

Fig. 17. Error under different ANR.

distances result in larger accumulated error. We note from Figure 17 that while all algorithms possess this relationship, the estimation error of the Centroid algorithm increases more significantly with increased ANR, in comparison to the other three algorithms. However, we also note that when the ANR is smaller than 3, APIT has a large InToOutErrorRatio due to the edge effect (described in Section 3.3.2). In this system configuration, a Centroid algorithm has its advantages.

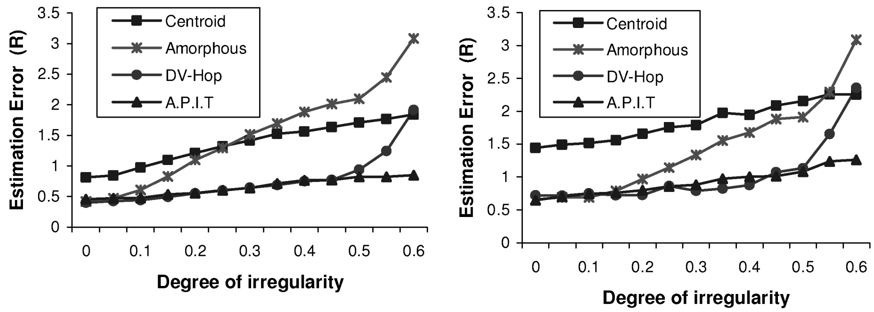
From an alternate perspective, we show that we can increase accuracy by using a smaller ANR. For example, the estimation error, shown in previous sections, can be reduced by about 30 to 50% when we use an ANR value of 5 instead of 10. However, this will increase the anchor percentage (AP) from 2 to 8%, requiring that more anchors be deployed.

6.4 Localization Error when Varying DOI

In this experiment, we investigate the impact of irregular radio patterns on the precision of localization estimation. It is intuitive that irregular radio patterns can affect the network topologies resulting in irregular hop count distributions in the Amorphous and DV-Hop algorithms. The HopSize formula, used in the Amorphous algorithm, assumes that radio patterns are perfectly circular. We can see, in Figure 18, how this inaccurate estimate directly contributes to localization error as the DOI increases. In contrast, the DV-Hop scheme estimates HopSize using online information-exchanged between anchors. This results in much better performance than the Amorphous algorithm, even though they are both DV-based algorithms. Because the Centroid and APIT algorithms do not depend on hop count and HopSize estimations and because the effect of DOI is abated by the aggregation of beaconed information, these algorithms are more robust than the Amorphous algorithm.

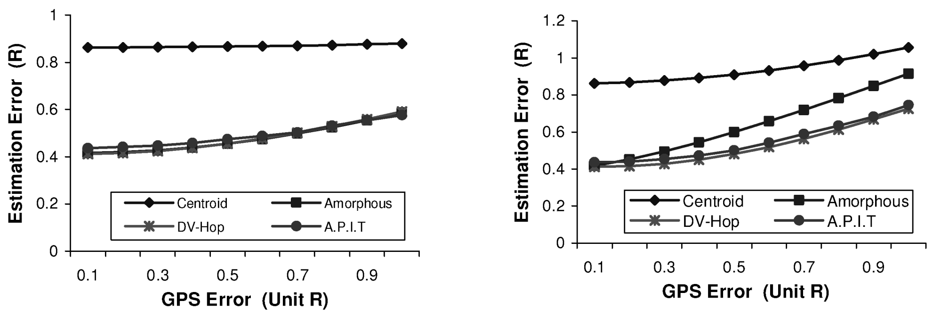
6.5 Localization Error When Varying GPS Error

In other experiments, we consider the distinct possibility that the GPS or an alternative system, which provides anchor nodes with location information, is error prone. Figure 19(A) and (B) demonstrate how initial location error at



(A) ANR = 10, ND = 8, AH =16, Uniform (B) ANR = 10, ND = 8, AH=16, Random

Fig. 18. Error under varying DOI.



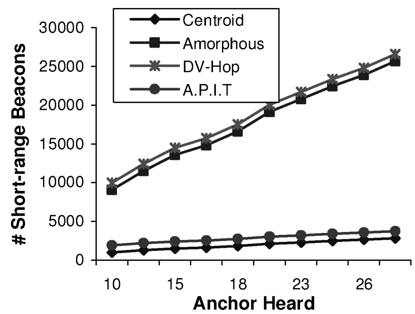
(A) ANR = 10, ND = 8, AH=16, Uniform, Random Error (B) ANR = 10, ND = 8, AH =16, Uniform, Bias Error

Fig. 19. Error under different GPS error.

anchors directly affects the error of the range-free localization protocols studied. In general, in all four schemes, GPS error is abated considerably by utilizing location information from multiple anchors. In the random error case (Figure 19A), we assume GPS error is isotropic, that is, the estimation error can occur in any direction. In this situation, the error impact of GPS is very small. We also see (Figure 19(B)) that when GPS error is biased (skewed in a particular direction) due to nonrandom factors, the estimation error of all schemes increases at a much slower rate than GPS error due to aggregation.

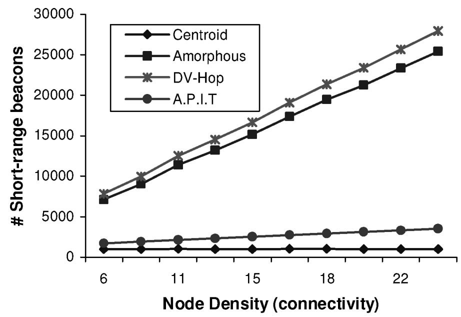
6.6 Communication Overhead for Varied AH

Figure 20 shows the results of experiments that test the communication overhead with regard to AH. It is important to note that the Centroid and APIT schemes use *long-range* anchor beacons, while the Amorphous and DV-hop algorithms use *short-range* beacons. Considering that energy consumption quadratically increases with increased beacon range, in Figure 20, we equate one long-range beacon to ANR^2 short-range beacons. This means that one long-range beacon sent out by APIT is counted as 100 short-range beacons when ANR = 10. Figure 20, shows that without flood-based beacon propagation, the Centroid and APIT algorithms use much fewer beacons than DV-based algorithms. For example, the APIT algorithm uses only about 10% of the beacons that the DV-Hop scheme uses when AH is set to 16.



ANR = 10, ND = 8, DOI = 0.1, Uniform

Fig. 20. Communication overhead for varied AH.



ANR = 10, AH = 16, DOI = 0.1, Uniform

Fig. 21. Overhead for varied node density.

Figure 20 also shows that APIT requires more beacons than the Centroid algorithm because of the neighborhood information-exchange. In addition, DV-Hop requires more beacons than the Amorphous algorithm because of additional online HopSize estimation requirements.

It should be noted that the evaluation of communication overhead here assumes a collision-free environment. If taking the collision into account, we expect that Amorphous and DV-hop algorithms introduce even more control overhead because of the flooding required by those two schemes.

6.7 Communication Overhead for Varied ND

Figure 21 demonstrates the effect of neighborhood density on required communication for localization. We can see from this graph that because there is no interaction between nodes in the Centroid scheme, the overhead stays constant. Communication overhead in our APIT scheme does increases with increased node density. However, it does so at a much lower rate than the DV-based schemes.

Drawing conclusions from Figures 20 and 21, we argue that as far as the communication overhead is concerned, the DV-Hop and Amorphous schemes are less suitable solutions for sensor networks with limited bandwidth when compared to the APIT and Centroid schemes. This is due to the large number of beacons required in these schemes.

Table I. Performance and Requirements Summary

	Centroid	DVHop	Amorp.	APIT
Accuracy	Fair	Good	Good	Good
NodeDensity	>0	>8	>8	>6
AnchorHeard	>10	>8	>8	>10
ANR	>0	>0	>0	>3
DOI	Good	Good	Fair	Good
GPSError	Good	Good	Fair	Good
Overhead	Smallest	Largest	Large	Small

6.8 Computational Overhead

The predominant concerns about sensor network protocols are the communication and power consumption overhead. However, it is desirable to evaluate the computational overhead of each algorithm. The major complexity of APIT algorithm is from the intersection of overlapped triangles. This has been discussed in Section 3.6. DV-Hop and Amorphous localization use multilateration to estimate nodes' locations. Their overheads are relatively smaller. Centroid algorithm only uses a simple averaging function and, thus, it has the smallest computation overhead.

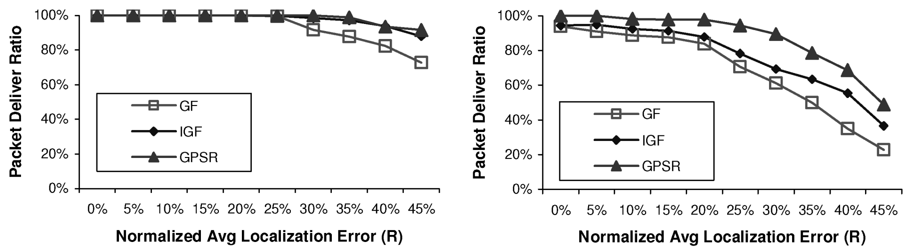
6.9 Evaluation Summary

In addition to the experiments previously discussed, we have conducted a variety of experiments to cover a varying range of system configurations. These experiments help us better understand the situations where the different localization schemes considered are more or less appropriate than one another.

Table I provides an overview of our results and can be used as a design guide for applying range-free schemes in WSN systems. This table shows that no single algorithm works best under all scenarios and that each localization algorithm has preferable system configurations. Although the Centroid scheme has the largest estimation error, its performance remains independent of node density and it boasts the smallest communication overhead and simplicity of implementation. Although DV-Hop requires more communication beacons to perform online estimation, it is notably more robust than the Amorphous algorithm in HopSize estimation. Finally, our APIT algorithm trumps the other algorithms when an irregular radio pattern and random node placement are considered and low communication overhead is desired. However, we acknowledge that APIT has more demanding requirements for both ANR values and the number of anchors used in localization.

7. LOCALIZATION ERROR IMPACT

In localization for WSNs, achieving better results (usually with regard to location accuracy) requires increasing the relative cost of the localization scheme via additional hardware, communication overhead, or the imposition of constraints and system requirements. Although more accurate location information is preferable, the desired level of granularity should depend on a cost/benefit analysis of the protocols that utilize this information. In this section, we investigate four types of location-dependent applications, namely, (1) location-based



(A) High-density scenario (22 node/radio range) (B) Low-density scenario (8 node/ radio range)

Fig. 22. Delivery ratio with varied localization errors.

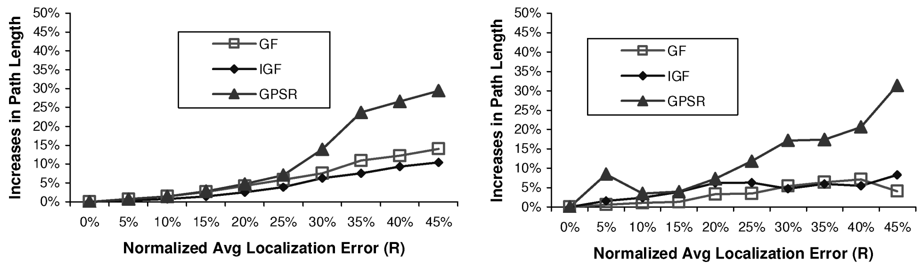
routing, (2) target estimation, (3) target tracking, and (4) sensing coverage. Based on the results, we conclude that except the routing in sparse networks, range-free localization schemes are able to support these sensor network applications sufficiently with only slight performance degradation.

7.1 Routing Performance

A localization service is critical for location-based routing protocols such as GF [Navas and Imielinski 1997], GPSR [Karp and Kung 2000], IGF [He et al. 2003b], LAR [Ko and Vaidya 1998], and GAF [Xu et al. 2001]. In these protocols, individual nodes make routing decisions based on knowledge of their geographic location. While most work in location-based routing assumes perfect location information, the fact is that erroneous location estimates are virtually impossible to avoid. Problems arise as error in the location service can influence location-based routing to choose the best next hop (the neighbor closest to the destination), or can make a node inadvertently think that the packet could not be routed because no neighbors are closer to the final destination.

To investigate the impact of localization error on routing, we studied three routing protocols GF [Navas and Imielinski 1997], GPSR [Karp and Kung 2000], and IGF [He et al. 2003b] under the low-traffic network conditions so that network congestion does not influence our results. In the experiments, localization errors are uniformly distributed in [0, 2 $\times$ Avg Localization Error], and the localization errors are normalized to units of node radio range (R) to ensure generally applicable results. We investigate both high (22 nodes per radio range) and low-density scenarios (8 nodes per radio range).

In the experiment, we increase the average localization error from 0 to 50% of the radio range in steps of 5% to measure the end-to-end delivery ratios and the percentage of increase in path length due to the localization errors. Two observations can be made about the impact of localization error to routing algorithms. First, when node density is high, the localization impact is relatively small. For example, all algorithms achieve 100% delivery ratio with moderate localization errors (<25%) (Figure 22). GPSR and IGF achieves 95% delivery ratio under about one-half radio range error. However, when node density is small, location-based routings suffer a lot. For example, although GPSR can deal with void, however, it can only delivery 50% packets when the localization error is about one-half radio range. This phenomenon suggests that ID-based



(A) High-density scenario (22 node/radio range) (B) Low-density scenario (8 node/ radio range)

Fig. 23. Increase in path length overhead with different localization errors.

routing should be used in sparse sensor network, if we cannot obtain very precise localization results.

Second, the path length increases moderately with the increase of the localization errors. The localization impact to the path length of GPSR is higher than other protocols, because GPSR uses the perimeter forwarding if the greedy forwarding fail due to the localization errors. We also note that the path length overhead is not affected much by the network node density, a fact that was not true for packet delivery ratio metric (Figure 23).

7.2 Target Estimation Performance

Many of the most frequently proposed applications for WSNs utilize target position estimations for tracking, search and rescue, or other means. In these proposed applications, when a target is identified, some combination of the nodes that sensed that target report their location to a centralized node (leader or base station). This node then performs aggregation on the received data to estimate the actual location of the target. Because target information could be used for locating survivors during a disaster, or identifying an enemy's position for strategic planning, the accuracy of this estimation is crucial to the application that uses it. Note that nodes normally have different sensing ranges with different sensing devices, and sensing ranges are normally different from communication radio range. For general applicability in the following experiments, we use sensing density (the number of nodes per sensing range) as one of system parameters.

Intuitively an increase in localization error will directly lead to target estimation error. To better understand the degree to which this error will propagate to other protocols, we investigate average estimation error under different node densities for varying degrees of location error. For these experiments, we implement a target estimation algorithm described in Blum et al. [2003]: the average $x$and y coordinates of all reporting nodes[3] are taken as the target location estimation. The results of various experiments are depicted in Figure 24. This graph shows that target estimation error due to location error is dampened during the aggregation process. Aside from showing varying degrees of estimation error with respect to node location error, Figure 24 also shows that the

[3]Nodes report when they sense the event of interest in the environment.

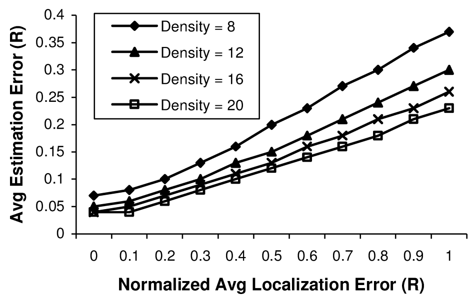


Fig. 24. Target estimation error with different localization errors under varying node density.

absolute target estimation error decreases with increased node density. For example, when localization error is equal to 0.5R and sensing density reaches 16 nodes per radio range, the estimation error is only about 65% as large as when the node density is 8. Based on Figure 24, we suggest that localization error impact can be reduced with a higher degree of aggregation.

7.3 Object-Tracking Performance

We evaluate the performance of tracking application that uses estimation in context. In this experiment, a mobile evader randomly walks around the specified terrain while a pursuer attempts to catch it. Initially the evader and pursuer are placed at the left top and left bottom corner, respectively. The evader chooses a direction to go across the terrain at a constant speed. After simulation starts, the pursuer is informed of the current location of the evader periodically via sensing nodes in the terrain that detect the evader, coordinate to estimate the targets position with regard to their own positions, and periodically report this result to the mobile pursuer. When receiving a report, the pursuer readjusts its direction in an attempt to intercept the evader. When the pursuer comes within the sensing radius of the evader, the evader is considered caught and the simulation ends. For this experiment, we compare the average tracking time (the time from pursuer takeoff to when the evader is caught) under different localization errors, to the tracking time in the case of no localization error. Figure 25 shows normalized tracking time in relation to the localization error for various pursuer speeds.

Figure 25 shows that tracking time increases slowly as the localization error increases. For example, when the average localization error is as large as 0.8R and the pursuer speed is 6 m/s, the pursuer requires only 20% more time in comparison to the ideal situation in which no localization error exists. This overhead can be further reduced to 10%, by increasing the pursuer's speed to 10 m/s.

7.4 Sensing-Coverage Performance

Energy efficient sensing coverage is critical for many important WSNs applications, such as military surveillance. A recent work, sensing-coverage [Yan et al. 2003] proposed a scheme that attempts to minimize energy consumption of sensing-coverage services. The basic idea of this scheme is when multiple

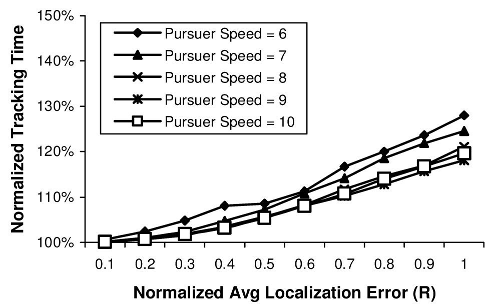


Fig. 25. Normalized tracking time with different pursuer speeds varying localization error. Terrain size, 1400m × 1400m; radio range, 100m; sensing range, 50m; density, 4 nodes per circle; evader speed, 5 m/s.

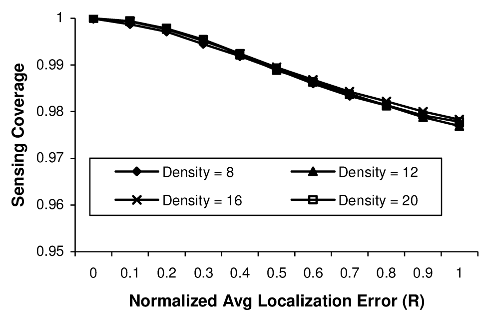


Fig. 26. Local error impact on sensing coverage.

sensor nodes cover a geographic location, ideally only one of the nodes is turned on at any time so that energy consumption is minimized. For each geographic location in the sensing area, all the sensors that can cover the location jointly determine their schedules of being turned on and off. Hence, energy consumption is minimized while sensing coverage is not compromised. In the absence of localization error, this scheme can guarantee 100% sensing coverage when no sensing void (location that can not be covered by any node) exists.

To investigate the impact of localization errors on the performance of this scheme, we implemented aforementioned algorithm and studied its sensing coverage ratio and energy consumption under different sensing densities and localization errors. Localization errors conceivably have negative impact on sensing coverage ratio because sensor nodes can falsely claim being able to cover a location based on inaccurate information of their locations and, hence, make the location left uncovered at some time. It is the case, however, that simulation results in Figure 26 indicate that such an effect is very small. Sensing coverage decreases slowly with increased localization errors across all sensing density values in our experiments. This mainly is because with uniform distributed localization error, the effects of under cover and over cover counteract each other, and, hence, reduce the chance of uncovered points.

In terms of energy consumption, the impact of localization errors is also small. Figure 27 plots the energy consumption under various localization errors

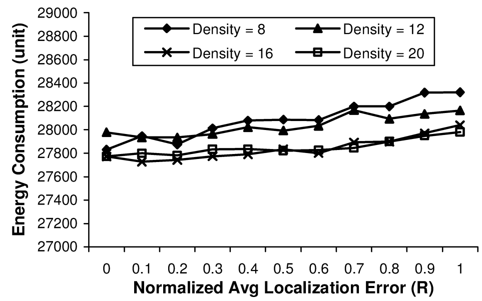


Fig. 27. Local error impact on energy consumption.

and sensing densities. We can see that average energy consumption increases only 1.7% when localization error is as large as one radio range (statistically insignificant), compared with the no error case. This is because all nodes are scheduled according to error locations and the length of resulting schedules are irrelevant to the actual locations of the nodes.

8. FUTURE WORK

It is well known that range-free localizations are subject to the effect of irregular radio patterns. We have done extensive evaluation of range-free protocols in simulation under a realistic radio model. However, we acknowledge that such a model can only serve as an approximation to real situation. Due to the lack of long-range anchor nodes and other constraints, we are not be able to evaluation all aforementioned protocols in a running system. We leave this to future work.

9. CONCLUSION

Given the inherent constraints of the sensor devices envisioned and the estimation accuracy desired by location-dependent applications, range-free localization schemes are regarded as a cost-effective and sufficient solution for localization in sensor networks. From our comparison study, we identify preferable system configurations of four different recently proposed range-free localization schemes as a design guideline for further research. In particular, an APIT scheme, proposed in this paper, performs best when irregular radio patterns and random node placement are considered and low communication overhead is desired. Moreover, we provide insight on how localization error affects a variety of location-dependent applications. These results show that the accuracy provided by the range-free schemes considered is sufficient to support various applications in sensor networks with only slight performance degradation.

REFERENCES

Bahl, P. and Padmanabhan, V. N. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proceedings of the IEEE INFOCOM '00*.

Beutel, J. 1999. Geolocation in a picoRadio environment, M.S. Thesis, ETH Zurich, Electronics Laboratory.

Blum, B., Nagaraddi, P., Wood, A., Abdelzaher, T. F., Son, S., and Stankovic, J. A. 2003. In *Proceedings of Mobisys 2003*, San Francisco, CA.

BULUSU, N., HEIDEMANN, J., AND ESTRIN, D. 2000. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine 7*, 5, 28–34.

BULUSU, N., HEIDEMANN, J., AND ESTRIN, D. 2001. Adaptive beacon placement. In *IEEE ICDCS '01*, Phoenix, AZ.

BULUSU, N., HEIDEMANN, J., ESTRIN, D., AND TRAN, T. 2003. Self-configuring localization systems: Design and experimental evaluation. In *TECS Special Issue on Networked Embedded Computing*.

CAFFERY, J. JR. 2000. A new approach to the geometry of TOA location. In *IEEE Vehicular Technology Conference (VTC)*, Boston, MA.

CAPKUN, S., HAMDI, M., AND HUBAUX, J. P. 2001. GPS-free positioning in mobile ad-hoc networks. In *Proceedings of HICCSS '01*, Maui, Hawaii.

CROSSBOW. MICA Sensor Board Information, http://www.xbow.com

DOHERTY, L., GHAOUI, L. E., AND PISTER, K. S. J. 2001. Convex position estimation in wireless sensor networks. In *Proceedings of the IEEE INFOCOM '01*, Anchorage, AK.

ESTRIN, D., GOVINDAN, R., HEIDEMANN, J., AND KUMAR, S. 1999. Next century challenges: Scalable coordination in sensor networks. In *Proceedings of MobiCom '99*, Seattle, Washington.

GANESAN, D., KRISHNAMACHARI, B., WOO, A., CULLER, D., ESTRIN, D., AND WICKER, S. 2002. Complex behavior at scale: An experimental study of low-power wireless sensor vetworks. *Technical Report UCLA/CSD-TR 02-0013*.

GIROD, L. AND ESTRIN, D. 2001. Robust range estimation using acoustic and multimodal sensing. In *Proceedings of IROS '01*, Maui, Hawaii.

HARTER, A., HOPPER, A., STEGGLES, P., WARD, A., AND WEBSTER, P. 1999. The anatomy of a context-aware application. In *Proceedings of MobiCom '99*, Seattle, Washington.

HE, T., HUANG, C., BLUM, B. M., STANKOVIC, J. A., AND ABDELZAHER, T. F. 2003a. Range-free localization schemes in large scale sensor networks. In *Proceedings of MobiCom 2003*.

HE, T., BLUM, B. M., STANKOVIC, J. A., AND ABDELZAHER, T. F. 2003b. A lazy-binding communication protocol for highly dynamic wireless sensor networks. Still in submission.

HIGHTOWER, J. AND BORIELLO, G. 2001. Location systems for ubiquitous computing. *IEEE Computer*, *34*, 8, 57–66.

HIGHTOWER, J., BORIELLO, G., AND WANT, R. 2000. SpotON: An indoor 3D location sensing technology based on RF signal strength. University of Washington CSE Report #2000-02-02.

HONG, X., XU, K., AND GERLA, M. 2002. Scalable routing protocols for mobile ad hoc networks. *IEEE Network Magazine*, *16*, 4.

KARP, B. AND KUNG, H. T. 2000. GPSR: Greedy perimeter stateless routing for wireless networks. In *Proceedings of MobiCom '00*, New York.

KLEINROCK, L. AND SILVESTER, J. 1978. Optimum transmission radii for packet radio networks or why six is a magic number. In *Proceedings of National Telecomm Conference*, 4.3.1–4.3.5.

KO, Y. B. AND VAIDYA, N. H. 1998. Location-aided routing (LAR) in mobile ad hoc networks. In *Proceedings of MobiCom '98*, Dallas, TX.

LI, J., JANNOTTI, J., DE COUTO, D. S. J., KARGER, D., AND MORRIS, R. 2000. A scalable location service for geographic ad-hoc routing. In *Proceedings of MobiCom '00*, New York.

NAGPAL, R. 1999. Organizing a global coordinate system from local information on an amorphous computer. A.I. Memo, MIT A.I. Laboratory.

NAGPAL, R., SHROBE, H., AND BACHRACH, J. 2003. Organizing a global coordinate system from local information on an ad hoc sensor network. In *IPSN '03*, Palo Alto.

NAVAS, J. C. AND IMIELINSKI, T. 1997. Geographic addressing and routing. In *Proceedings of MobiCom '97*, Budapest, Hungary.

NICULESCU, D. AND NATH. B. 2001. Ad-hoc positioning systems. In *Proceedings of IEEE GLOBECOM '01*.

NICULESCU, D. AND NATH, B. 2003a. DV based positioning in ad hoc networks. *Journal of Telecommunication Systems 22*, 1, 267–280.

NICULESCU, D. AND NATH, B. 2003b. Ad hoc positioning system (APS) using AoA. In *Proceedings of the IEEE INFOCOM' 03*, San Francisco, CA.

PRIYANTHA, N. B., CHAKRABORTY, A., AND BALAKRISHNAN, H. 2000. The cricket location-support system. In *Proceedings of MobiCom '00*, New York.

SAVARESE, C., RABAY, J., AND LANGENDOEN, K. 2002. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *USENIX Technical Annual Conference*, Monterey, CA.

SAVVIDES, A., HAN, C. C., AND SRIVASTAVA, M. B. 2001. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of MobiCom'01*, 2001, Rome, Italy (July).

SAVVIDES, A., PARK, H., AND SRIVASTAVA, M. B. 2002. The bits and flops of the N-hop multilateration primitive for node localization problems. In *WSNA'02*, Atlanta, GA.

WANT, R., HOPPER, A., FALCAO, V., AND GIBBONS, J. 1992. The active badge location system. *ACM Transactions on Information Systems*.

WELLENHOFF, B. H., LICHTENEGGER, H., AND COLLINS, J. 1997. *Global Positions System: Theory and Practice*, 4th Ed. Springer Verlag, NY.

WHITEHOUSE, K. AND CULLER, D. 2002. Calibration as parameter estimation in sensor networks. In *First ACM International Workshop on Wireless Sensor Networks and Application*, Atlanta GA.

XU, Y., HEIDEMANN, J., AND ESTRIN, D. 2001. Geography-informed energy conservation for ad hoc routing. In *Proceedings of MobiCom '01*, Rome, Italy.

YAN, T., HE, T., AND STANKOVIC, J. A. 2003. Differentiated surveillance service for sensor networks. In *SenSys 2003*, Los Angeles, CA.

ZHOU, G., HE, T., KRISHNAMURTHY, S., AND STANKOVIC, J. A. 2004. Impact of radio irregularity on wireless sensor networks. In *Proceedings of MobiSys'04*, Boston, MA.

Received December 2003; revised April 2004; accepted April 2004