This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2008 proceedings.

Towards a Theory of Robust Localization Against Malicious Beacon Nodes

Sheng Zhong Murtuza Jadliwala Shambhu Upadhyaya Chunming Qiao
Department of Computer Science and Engineering,
State University of New York at Buffalo,
Buffalo, NY 14260, U. S. A.
Email: {szhong,msj3,shambhu,qiao}@cse.buffalo.edu

***Abstract*—Localization in the presence of malicious beacon nodes is an important problem in wireless networks. Although significant progress has been made on this problem, some fundamental theoretical questions still remain unanswered: in the presence of malicious beacon nodes, what are the necessary and sufficient conditions to guarantee a bounded error during 2-dimensional location estimation? Under these necessary and sufficient conditions, what class of localization algorithms can provide that error bound? In this paper, we try to answer these questions. Specifically, we show that, when the number of malicious beacons is greater than or equal to some threshold, there is no localization algorithm that can have a bounded error. Furthermore, when the number of malicious beacons is below that threshold, we identify a class of localization algorithms that can ensure that the localization error is bounded. We also outline two algorithms in this class, one of which is guaranteed to finish in polynomial time (in the number of beacons providing information) in the worst case, while the other is based on a heuristic and is practically efficient. For completeness, we also extend the above results to the 3-dimensional case. Experimental results demonstrate that our solution has very good localization accuracy and computational efficiency.**

I. Introduction

Localization or *location discovery* in distributed wireless networks is the problem where every node in the network needs to efficiently and accurately determine its own location w.r.t some local or global coordinate system. In this work, we focus on beacon-based algorithms that use distance information to compute locations ([1], [2], [6], [7], [14], [15], [19], [20]). Such algorithms require the presence of special nodes, called beacon or anchor nodes, that are placed at strategic positions in the network and they know their own locations. Then, the remaining nodes estimate their location by computing distance estimates to a set of beacon nodes. Beacon-based methods perform well when all the beacon nodes are honest, but their accuracy suffers considerably due to the presence of malicious beacon nodes. Malicious beacons can cheat by broadcasting incorrect self locations or by manipulating the transmit power levels, thus altering the distance computation and eventually the estimated final location of the target node.

Previous research efforts in this direction have focused only on either removing this (over)dependence on beacon nodes ([4], [8], [21]) or on minimizing the effects of malicious beacons ([10], [11]) during localization. But, even before delving into the possible solutions for this problem, we feel that there are a plethora of questions that have been left unanswered by previous research efforts: Under what condition(s) do there exist algorithms that can overcome the cheating effect of the malicious beacons? When such algorithms exist, how can we find them out? What kind of guarantee on the solution quality (in terms of bounds on the error in localization) can such algorithms provide? None of the previous research works have attempted to answer these questions. Specifically, there has been no systematic study on the hardness and feasibility of the localization problem in hostile environments. In this paper, we attempt to fill this gap by first establishing the necessary (and sufficient) condition for distributed distance-based localization in the presence of malicious nodes. After such an initial feasibility study, we identify a class of algorithms that provides a guarantee on the localization accuracy, even in the presence of cheating beacon nodes.

Specifically, we make the following contributions in this paper. First, we prove that if the number of malicious nodes is greater than or equal to $\frac{n-2}{2}$, where n is the number of beacons providing information, then no algorithm can provide any degree of localization accuracy. Next, we show that there exist algorithms that provide a guaranteed degree of localization accuracy, if the number of malicious beacons is less than or equal to $\frac{n-3}{2}$. To prove this result, we identify a class of algorithms such that each algorithm in this class determines the location of a node with bounded localization error. Later, we present two illustrative examples of algorithms in this class. The first algorithm has a worst-case computational complexity polynomial in n, while the second algorithm has much better efficiency in practice. In addition to the above theorems and algorithms, we extend our work to the 3-dimensional case, where the location of every node is represented by points in the three-dimensional coordinate system. Finally, we verify the localization accuracy and computational efficiency of the proposed algorithms through simulation experiments.

The rest of the paper is organized as follows. We discuss the background and related work in Section II and present our network model in Section III. In Section IV, we prove the necessary condition for existence of localization algorithm with guaranteed degree of accuracy; in Section V, we give the definition of the algorithm class with guaranteed degree of accuracy. Two example algorithms in the class are given in Section VI, while the extension to 3-dimensional localization

978-1-4244-2026-1/08/$25.00

is given in Section VII. Experimental evaluations are in Section VIII. We conclude in Section IX.

II. Background and Related Work

In the past, researchers have followed two approaches towards overcoming the problem of malicious nodes in localization algorithms. The first approach is to detect malicious nodes by observing the inconsistencies in the communication from such nodes and efficiently eliminating them (from consideration) before localization. Sastry et al. [17] proposed a location verification technique to verify the relative distance between a verifying node and a beacon node while Pires et al. [9] gives protocols to detect malicious nodes in range-based localization approaches by detecting malicious message transmissions. Liu et al. [12] also proposed methods to detect malicious beacon nodes in beacon-based localization approaches by deploying special detector nodes that capture malicious message transmissions by the beacon nodes.

Another approach towards robust localization is to efficiently perform localization in the presence of errors in distance measurements. These errors can be a result of external factors like random noise, measurement errors etc. or due to malicious nodes. Moore et al. [13] formulated the localization problem as a two-dimensional graph realization problem and described a beaconless (anchor-free), distributed, linear-time algorithm for localizing nodes in the presence of range measurement noise. Liu et al. [11] proposed two methods for robust localization in the presence of malicious beacon nodes. The first method filters out malicious beacon signals on the basis of inconsistency among multiple beacon signals, while the second method tolerates malicious beacon signals by adopting an iteratively refined voting scheme. As we will discuss later, this voting-based scheme proposed by Liu et al. is an algorithm in the class of algorithms that can provide bounded localization accuracy. Li et al. [10] develop robust statistical methods to make localization attack-tolerant. They propose an adaptive least squares and least median squares position estimator for beacon-based localization using triangulation. Similarly, Doherty et al. [3] use connectivity constraints and convex optimization to minimize errors in beacon-based localization techniques. Others have approached this problem by eliminating the need for beacon nodes during localization [4], [8], [15], [18]. Recently, researchers have also applied ideas from other domains like coding theory to achieve robustness in localization algorithms [16], [21].

III. Network Model

In this section, we describe the network model for the problem of distance-based localization (using beacon nodes) of a mobile device M in hostile environments. In other words, M wants to compute its own location using distance estimates to beacon nodes that know their own location and these beacon nodes may or may not cheat. Suppose that there are n beacons available for localization, denoted as $B_1, \ldots, B_n$. Among these n beacons, some are malicious. Let k be the number of malicious beacons. It is important to note that k is not necessarily known to the mobile device or to any honest beacons. However, the value of k clearly has a great influence on whether we can achieve a bounded localization error. In Section IV, we will establish the condition for having a bounded localization error based on the value of k.

Regardless of being honest or dishonest, each beacon B_i provides M with a measurement $\tilde{d}_i$ of the distance between B_i and M. (In practice, each beacon B_i actually provides M with some information from which the distance $\tilde{d}_i$ can be computed efficiently by M. We simplify this by letting B_i provide the measurement directly, which should not affect the results.) The precise distance between B_i and M is the Euclidean distance between the position coordinates of B_i and M and is denoted by $dst(B_i, M)$. Let the set of honest beacons be denoted by H. Then, for each beacon $B_i \in H$, $\tilde{d}_i$ is assumed to be a random variable that follows some fixed probability distribution, denoted as $msr(dst(B_i, M))$, such that

$$E[\tilde{d}_i] = dst(B_i, M),$$

i.e., the expected (mean) value of the estimated distance $\tilde{d}_i$ for each beacon B_i in H, is the precise distance between the beacon B_i and the node M. Also, in the case when B_i is honest, the difference between the estimated and the true distance is assumed to be very small, i.e.,

$$|\tilde{d}_i - dst(B_i, M)| < \epsilon,$$

where ϵ is a small constant. Ideally, this difference should be zero when the beacon is honest, but such discrepancies in distance estimates can occur due to factors like *measurement errors* either at the source or target. For each beacon $B_i \notin H$, $\tilde{d}_i$ is a value selected arbitrarily by the adversary. Note that we implicitly allow colluding attack here: In our model, we consider a single adversary who controls all malicious beacon nodes and decides $\tilde{d}_i$ for all $B_i \notin H$. This is a very strong adversary model that covers all possibility of collusion among malicious beacon nodes.

Since we assume a distance-based localization strategy, the output O of a localization algorithm can be defined by a function F of the measured distances $(\tilde{d}_i)$ from the device M to every beacon node in the network as shown below.

$$O = F(\tilde{d}_1, \ldots, \tilde{d}_n).$$

The error e of the localization algorithm is defined as the Euclidean distance between the actual position of the mobile device and the one output by the algorithm.

$$e = E[dst(M, O)].$$

Our next aim is, given the above model, to derive the necessary and sufficient condition for the existence of an algorithm that can do distance-based localization with a bounded localization error in the presence of malicious beacon nodes.

IV. Necessary Condition for Bounded Localization Error

In this section, we give a threshold of k such that if the number of malicious beacons is greater than or equal to this

threshold then no algorithm would be able to guarantee a bounded localization accuracy just based on the distances to the beacon nodes. Consequently, having the number of malicious beacons below this threshold is a necessary condition for getting a bounded localization error out of any distance-based localization algorithm.

Theorem 1: Suppose that $k \geq \frac{n-2}{2}$. Then, for any distance-based localization algorithm, for any locations of the beacons, there exists a scenario in which e is unbounded.

Proof: Without loss of generality, we assume that $k = \frac{n-2}{2}$ (because more malicious beacons clearly can launch any attack that $\frac{n-2}{2}$ malicious beacons can launch). We give the proof for the above theorem by a contradiction argument. Suppose that, in all scenarios, the output error $e < a$, where a is a constant. We show that this supposition leads to a contradiction. We first prove that for a fixed set of beacon nodes and beacon locations, if the above threshold holds (and if the exact identities of the malicious nodes are not known) then there exists at least two distinct scenarios having the same distribution of distances from the target node to the beacon nodes. This makes it impossible for any algorithm to differentiate between the two scenarios. Since the target location in the two scenarios have a significant difference, any algorithm must fail in one of the two scenarios.

Consider the two scenarios S_1 and S_2, as shown in Figure 1. The locations of all the beacons are same in both the scenarios, but the set of honest beacon nodes and the position of the target node M is assumed to be different in each scenario. Select an arbitrary point P in the line segment B_1B_2 and draw a line L through P such that L is perpendicular to B_1B_2. Choose an arbitrary number $a' > a$. Then there are two points P_1 and P_2 on the line L such that

$$dst(P_1, P) = dst(P_2, P) = \frac{1}{2}dst(P_1, P_2) = a' \geq a.$$

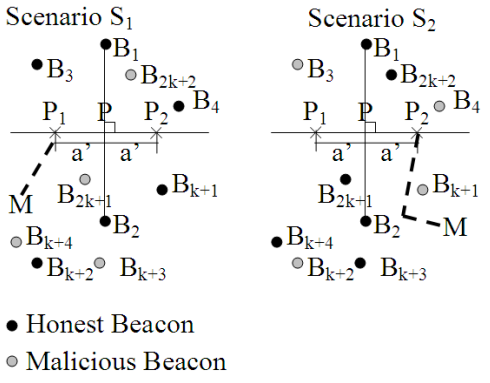

Fig. 1. Two Scenarios for Lower Bound Theorem

In scenario S_1, M is at location P_1 and the set of honest beacons is $H_1 = \{B_1, B_2, B_3, \ldots, B_{k+2}\}$. Denote by $\tilde{d}_{i,1}$ the measurement $\tilde{d}_i$ in scenario S_1. So, for each $B_i \in H_1$,

$$\tilde{d}_{i,1} \sim msr(dst(B_i, P_1)).$$

In scenario S_2, M is at location P_2 and the set of honest beacons is $H_2 = \{B_1, B_2, B_{k+3}, \ldots, B_{2k+2}\}$. Denote by $\tilde{d}_{i,2}$ the measurement $\tilde{d}_i$ in scenario S_2. So, for each $B_i \in H_2$,

$$\tilde{d}_{i,2} \sim msr(dst(B_i, P_2)).$$

Assume that in scenario S_1, the adversary chooses $\tilde{d}_{k+3,1}, \ldots, \tilde{d}_{2k+2,1}$ such that

$$\forall i \in \{k+3, \ldots, 2k+2\}, \tilde{d}_{i,1} \sim msr(dst(B_i, P_2)).$$

Similarly, assume that in scenario S_2, the adversary chooses $\tilde{d}_{3,2}, \ldots, \tilde{d}_{k+2,2}$ such that

$$\forall i \in \{3, \ldots, k+2\}, \tilde{d}_{i,2} \sim msr(dst(B_i, P_1)).$$

Since B_1 and B_2 are on the perpendicular bisector of line segment P_1P_2, we have

$$dst(B_1, P_1) = dst(B_1, P_2);$$

$$dst(B_2, P_1) = dst(B_2, P_2).$$

Therefore, we have two pairs of identical distributions:

$$msr(dst(B_1, P_1)) \cong msr(dst(B_1, P_2));$$

$$msr(dst(B_2, P_1)) \cong msr(dst(B_2, P_2)).$$

Now, it is easy to see that $(\tilde{d}_{1,1}, \tilde{d}_{2,1}, \tilde{d}_{3,1}, \ldots, \tilde{d}_{2k+2,1})$ and $(\tilde{d}_{1,2}, \tilde{d}_{2,2}, \tilde{d}_{3,2}, \ldots, \tilde{d}_{2k+2,2})$ are identically distributed. Consequently, the two outputs

$$O_1 = F(\tilde{d}_{1,1}, \tilde{d}_{2,1}, \tilde{d}_{3,1}, \ldots, \tilde{d}_{2k+2,1})$$

and

$$O_2 = F(\tilde{d}_{1,2}, \tilde{d}_{2,2}, \tilde{d}_{3,2}, \ldots, \tilde{d}_{2k+2,2})$$

are also identically distributed. This implies that

$$E[dst(P_2, O_1)] = E[dst(P_2, O_2)].$$

On the other hand, by our assumption, the output errors in both scenarios are less than a:

$$e_1 = E[dst(P_1, O_1)] < a,$$

$$e_2 = E[dst(P_2, O_2)] < a.$$

Consequently,

$$\begin{aligned} dst(P_1, P_2) &= E[dst(P_1, P_2)] \\ &\leq E[dst(P_1, O_1)] + E[dst(P_2, O_1)] \\ &= E[dst(P_1, O_1)] + E[dst(P_2, O_2)] \\ &< a + a \\ &= 2a. \end{aligned}$$

This is contradictory to the fact that $dst(P_1, P_2) = 2a' \geq 2a$. ■

This brings us to our next result in which we prove that, given the network model as explained in Section III, with no more than $\frac{n-3}{2}$ malicious beacons we can definitely compute the location of M with an error bound proportional to ϵ.

V. Algorithm Class and Error Analysis

In the previous section, we have shown that having $\frac{n-2}{2}$ or more malicious beacons makes it impossible to compute the location of M with a bounded error. In this section, we show that having $\frac{n-3}{2}$ or fewer malicious beacons makes it possible to compute the location of M with a bounded error. In particular, we identify a class of algorithms that can compute the location under this condition and present a formal analysis of the maximum localization error of such algorithms.

A. Class of Algorithms for Robust Localization

Before defining this algorithm class, we describe some terminology that we will be using during its definition. For each beacon B_i, define a ring R_i using the following inequality:

$$\tilde{d}_i - \epsilon < dst(B_i, X) < \tilde{d}_i + \epsilon.$$

As mentioned in Section III, ϵ is a small constant signifying some small measurement error. Clearly, there are altogether n rings. The boundary of these n rings consists of $2n$ circles—we call these circles the *boundary circles*. In particular, the inner circle of a ring is called an *inner boundary circle*, while the outer circle of a ring is called an *outer boundary circle*.

Definition 1: We say a point is a *critical point* if it is the intersection of at least two boundary circles. We say an arc is a *continuous arc* if it satisfies the following three conditions:

- The arc is part of a boundary cirle.
- If the arc is not a complete circle, then its two ends are both critical points.
- There is no other critical point in the arc.

We say an area is a *continous region* if it satisfies the following two conditions:

- The boundary of this area is one or more continous arcs.
- There is no other continuous arc inside the area.

For each beacon B_i, define a ring R_i using the inequality: $\tilde{d}_i - \epsilon < dst(B_i, X) < \tilde{d}_i + \epsilon$. We give the definition of our algorithm class based on these rings:

Definition 2: A localization algorithm is in the class of robust localization algorithms if its output is a point in a continuous region r such that r is contained in the intersection of at least $k + 3$ rings.

Note that, in the definition above, we have defined a *non-empty* class of algorithms. To see this, we show that, as long as $k \leq \frac{n-3}{2}$, we can always find a non-empty continuous region r satisfying the above requirement.

Theorem 2: For $k \leq \frac{n-3}{2}$, there exists a non-empty continuous region r in the intersection of at least $k + 3$ rings.

Proof: Consider the real location of mobile device M. Clearly, for each honest beacon B_i, M must be in the ring R_i:

$$\tilde{d}_i - \epsilon < dst(B_i, M) < \tilde{d}_i + \epsilon.$$

Since $k \leq \frac{n-3}{2}$, i.e., $n \geq 2k+3$, there are at least $k+3$ honest beacons. So M must be in the intersection of at least $k + 3$ rings. Define r as the continuous region in the intersection of these rings that contains the real location of M. Since M is in r, r must be non-empty. (Figure 2 gives an illustration.) ■

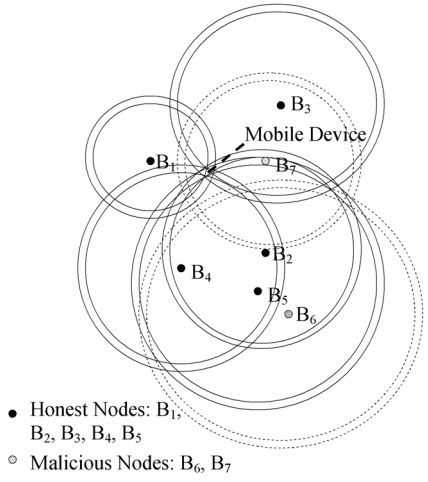

Fig. 2. Existence of Intersection of Rings ($k = 2$)

In fact, an example algorithm that belongs to this class is the voting-based localization scheme proposed by Liu et al. [11]. In Liu et al.'s scheme, they compute the intersection region (as discussed above) by dividing the entire localization area into a square grid and then taking a vote for each candidate location on the grid. The candidate locations with the maximum votes belong to the intersection area. Although very simple, the voting-based algorithm is computationally expensive, as it has to store the states of all the points on the grid and does an exhaustive search for the point with the maximum votes. In Section VI, we propose two other algorithms in this class that are much more efficient, one having a low worst-case complexity, the other running very fast in practice.

B. Error Bound Analysis

To analyze the error bound of algorithms in this class, we need to establish a couple of new definitions.

Definition 3: The *beacon distance ratio (γ)* is defined as the minimum distance between a pair of beacons divided by the maximum distance between a beacon and the mobile device:

$$\gamma = \frac{\min_{B_i, B_j} dst(B_i, B_j)}{\max_{B_i} dst(B_i, M)}.$$

Definition 4: Consider the lines going through pairs of beacons. Denote by $ang(B_iB_j, B_{i'}B_{j'})$ the angle between lines B_iB_j and $B_{i'}B_{j'}$—to avoid ambiguity, we require that $0° \leq ang(B_iB_j, B_{i'}B_{j'}) \leq 90°$. The *minimum beacon angle (α)* is defined as the minimum of such angles:

$$\alpha = \min_{B_i, B_j, B_{i'}, B_{j'}} ang(B_iB_j, B_{i'}B_{j'}).$$

The following theorem bounds the maximum localization error possible in our robust localization framework.

Theorem 3: For $k \leq \frac{n-3}{2}$, if $\epsilon \ll \min_{B_i} dst(B_i, M)$ and there are no three beacons in the same line, then the output error of any algorithm in the class of algorithms for robust

localization, as defined in Definition 2, is

$$e < \frac{2\epsilon}{\min\left\{\sin\frac{\arcsin(\gamma\sin(\alpha/2))}{2}, \cos\frac{\arcsin(\gamma\sin(\alpha/2))}{2}\right\}}.$$

Proof: Consider the continuous region r. It is in the intersection of at least $k+3$ rings. Since there are at most k dishonest beacons, at least 3 of these rings belong to honest beacons. Suppose that R_{i_1}, R_{i_2}, and R_{i_3} are three rings belonging to honest beacons among the at least $k+3$ rings. Let r' be the continuous region in the intersection of R_{i_1}, R_{i_2}, and R_{i_3} that contains r. Since O is in r, clearly O is also in r'. Next, we show that M is also in r'. Since M is also in the intersection of R_{i_1}, R_{i_2}, and R_{i_3}, we only need to prove the following lemma.

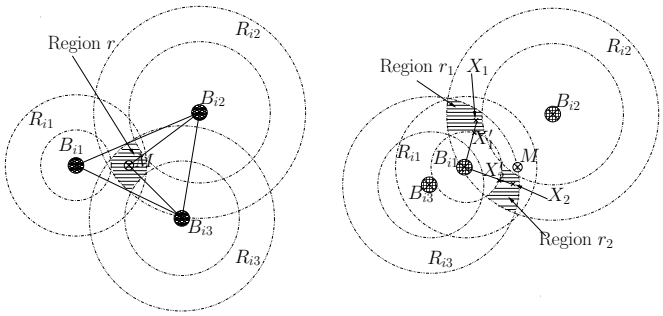

Fig. 3. Intersection of Rings

Lemma 1: If $\epsilon \ll \min_{B_i} dst(B_i, M)$ and there are no three beacons in the same line, then the intersection of R_{i_1}, R_{i_2}, and R_{i_3} has only one continuous region.

Proof: We prove by contradiction, as illustrated in Figure 3. Suppose that the intersection of R_{i_1}, R_{i_2}, and R_{i_3} has two continuous regions r_1 and r_2. Choose arbitrary points X_1 from r_1 and X_2 from r_2. Denote by X_1' (resp., X_2') the intersection of the line segment $B_{i_1}X_1$ (resp., $B_{i_1}X_2$) and the circle

$$dst(X, B_{i_1}) = \tilde{d}_{i_1} - \epsilon.$$

Similarly, denote by X_1'' (resp., X_2'') the intersection of the line segment $B_{i_3}X_1$ (resp., $B_{i_3}X_2$) and the circle

$$dst(X, B_{i_3}) = \tilde{d}_{i_3} - \epsilon.$$

Then clearly,

$$0 \le dst(X_1, X_1'), dst(X_1, X_1''), dst(X_2, X_2'), dst(X_2, X_2'') \le 2\epsilon. \quad (1)$$

We can see that,

$$\begin{aligned} ang(B_{i_1}B_{i_3}, B_{i_1}X_1) &= \arccos(dst(B_{i_1}, X_1)^2 \\ &\quad + dst(B_{i_1}, B_{i_3})^2 - dst(X_1, B_{i_3})^2) \\ &= \arccos((dst(B_{i_1}, X_1') \\ &\quad + dst(X_1, X_1'))^2 + dst(B_{i_1}, B_{i_3})^2 \\ &\quad - (dst(X_1'', B_{i_3}) + dst(X_1, X_1''))^2) \\ &= \arccos((\tilde{d}_{i_1} - \epsilon + dst(X_1, X_1'))^2 \\ &\quad + dst(B_{i_1}, B_{i_3})^2 \\ &\quad - (\tilde{d}_{i_3} - \epsilon + dst(X_1, X_1''))^2). \end{aligned}$$

We note that $\tilde{d}_{i_1} > dst(B_{i_1}, M) - \epsilon \gg \epsilon$. Similarly, $\tilde{d}_{i_3} \gg \epsilon$. Combining these facts with (1), we have

$$\begin{aligned} ang(B_{i_1}B_{i_3}, B_{i_1}X_1) &= \arccos((\tilde{d}_{i_1} - \epsilon + dst(X_1, X_1'))^2 \\ &\quad + dst(B_{i_1}, B_{i_3})^2 \\ &\quad - (\tilde{d}_{i_3} - \epsilon + dst(X_1, X_1''))^2) \\ &\approx \arccos((\tilde{d}_{i_1})^2 + dst(B_{i_1}, B_{i_3})^2 \\ &\quad - (\tilde{d}_{i_3})^2) \\ &\approx \arccos((\tilde{d}_{i_1} - \epsilon + dst(X_2, X_2'))^2 \\ &\quad + dst(B_{i_1}, B_{i_3})^2 \\ &\quad - (\tilde{d}_{i_3} - \epsilon + dst(X_2, X_2''))^2) \\ &= \arccos((dst(B_{i_1}, X_2') + \\ &\quad dst(X_2, X_2'))^2 + dst(B_{i_1}, B_{i_3})^2 \\ &\quad - (dst(X_2'', B_{i_3}) + dst(X_2, X_2''))^2) \\ &= \arccos(dst(B_{i_1}, X_2)^2 \\ &\quad + dst(B_{i_1}, B_{i_3})^2 - dst(X_2, B_{i_3})^2) \\ &= ang(B_{i_1}B_{i_3}, B_{i_1}X_2). \end{aligned}$$

Similarly, we can show that

$$ang(B_{i_1}B_{i_2}, B_{i_1}X_1) \approx ang(B_{i_1}B_{i_2}, B_{i_1}X_2).$$

However, when we put the above two equations together, we can get a contradiction. Without loss of generality, we assume that

$$ang(B_{i_1}B_{i_2}, B_{i_1}X_1) < ang(B_{i_1}B_{i_3}, B_{i_1}X_1),$$

since otherwise we can switch the indices i_2 and i_3. It is easy to see

$$\begin{aligned} ang(B_{i_1}B_{i_2}, B_{i_1}X_1) &= ang(B_{i_1}B_{i_3}, B_{i_1}X_1) \\ &\quad - ang(B_{i_1}B_{i_2}, B_{i_1}B_{i_3}) \\ &\le ang(B_{i_1}B_{i_3}, B_{i_1}X_1) - \alpha \\ &\approx ang(B_{i_1}B_{i_3}, B_{i_1}X_2) - \alpha \\ &= ang(B_{i_1}B_{i_2}, B_{i_1}X_2) \\ &\quad - ang(B_{i_1}B_{i_2}, B_{i_1}B_{i_3}) - \alpha \\ &\le ang(B_{i_1}B_{i_2}, B_{i_1}X_2) - 2\alpha \\ &\approx ang(B_{i_1}B_{i_2}, B_{i_1}X_1) - 2\alpha, \end{aligned}$$

which is a contradiction. ■

Now we know that both M and O are in r'. We will use this fact to show that

$$e < \frac{2\epsilon}{\min\left\{\sin\frac{\arcsin(\gamma\sin(\alpha/2))}{2}, \cos\frac{\arcsin(\gamma\sin(\alpha/2))}{2}\right\}}.$$

But before we can prove this result, we need another lemma:

Lemma 2: If there are no three beacons in the same line, then either

$$ang(B_{i_1}M, B_{i_2}M) \ge \arcsin(\gamma\sin(\alpha/2)),$$

or

$$ang(B_{i_1}M, B_{i_3}M) \ge \arcsin(\gamma\sin(\alpha/2)).$$

Proof: Since $ang(B_{i_1}B_{i_2}, B_{i_1}B_{i_3}) \geq \alpha$, we have either $ang(B_{i_1}B_{i_2}, B_{i_1}M) \geq \alpha/2$ or $ang(B_{i_1}B_{i_3}, B_{i_1}M) \geq \alpha/2$. Below we show that, if $ang(B_{i_1}B_{i_2}, B_{i_1}M) \geq \alpha/2$, then

$$ang(B_{i_1}M, B_{i_2}M) \leq \frac{\arcsin(\gamma \sin(\alpha/2))}{2}.$$

Similarly, we can show that, if $ang(B_{i_1}B_{i_3}, B_{i_1}M) \geq \alpha/2$, then

$$ang(B_{i_1}M, B_{i_3}M) \leq \frac{\arcsin(\gamma \sin(\alpha/2))}{2}.$$

Denote by D the distance from B_{i_2} to the line $B_{i_1}M$. Then

$$\begin{aligned} ang(B_{i_1}M, B_{i_2}M) &= \arcsin\left(\frac{D}{dst(B_{i_2}, M)}\right) \\ &= \arcsin\left(\frac{dst(B_{i_1}, B_{i_2})\sin(ang(B_{i_1}B_{i_2}, B_{i_1}M))}{dst(B_{i_2}, M)}\right) \\ &\geq \arcsin\left(\frac{dst(B_{i_1}, B_{i_2})\sin(\alpha/2)}{dst(B_{i_2}, M)}\right) \\ &\geq \arcsin(\gamma \sin(\alpha/2)). \end{aligned}$$

■

Using the above lemma, we know that, without loss of generality, we can assume that

$$ang(B_{i_1}M, B_{i_2}M) \geq \arcsin(\gamma \sin(\alpha/2)).$$

Denote by r'' the continuous region in the intersection of R_{i_1} and R_{i_2} that contains r'. Since both M and O are in r', they should also be in r''.

Each of the two rings involved has a pair of circles. Consider the four intersection points of these two pairs of circles. Without loss of generality, we suppose that the four intersection points are V_1, V_2, V_3, and V_4, ordered in the clockwise direction, and that $\angle V_2V_1V_4$ is acute. Since $\epsilon \ll \min_{B_i} dst(B_i, M)$, we can approximate r'' using the quadrangle $V_1V_2V_3V_4$. It is easy to show that

$$ang(V_1V_2, B_{i_1}M) \approx 90^\circ \approx ang(V_3V_4, B_{i_1}M);$$

thus we know that the line V_1V_2 is parallel to the line V_3V_4. Similarly, we can get that the line V_1V_4 is parallel to the line V_2V_3. Therefore, $V_1V_2V_3V_4$ is a parallelogram. Furthermore, we observe that

$$\begin{aligned} \angle V_2V_1V_3 &= \arcsin\left(\frac{2\epsilon}{dst(V_1, V_3)}\right) \\ &= \angle V_3V_1V_4. \end{aligned}$$

Therefore, $V_1V_2V_3V_4$ is actually a rhombus. In a rhombus, the farthest distance between two points is the length of its longer diagonal line. Therefore,

$$\begin{aligned} e = dst(M, O) &\leq \frac{2\epsilon}{\sin(\angle V_2V_1V_3)} \\ &= \frac{2\epsilon}{\sin\left(\frac{\angle V_2V_1V_4}{2}\right)} \\ &\approx \frac{2\epsilon}{\min\left\{\sin\left(\frac{ang(B_{i_1}M, B_{i_2}M)}{2}\right), \sin\left(90^\circ - \frac{ang(B_{i_1}M, B_{i_2}M)}{2}\right)\right\}} \\ &\leq \frac{2\epsilon}{\min\left\{\sin\left(\frac{\arcsin(\gamma \sin(\alpha/2))}{2}\right), \cos\left(\frac{\arcsin(\gamma \sin(\alpha/2))}{2}\right)\right\}}. \end{aligned}$$

■

VI. Two Example Algorithms

In this section, we present two example algorithms in the class defined in Definition 2. The first algorithm has a worst case computational complexity of $O(n^3 \log n)$. (Recall n is the number of beacons. Clearly this is much faster than an exhaustive search in grid points [11].) However, in practice, since the worst-case scenario rarely occurs, it is still not sufficiently fast. Our second algorithm is a heuristic one. Although it does not have a worst-case complexity analysis as the first algorithm, it runs very fast in practice.

Recall that these algorithms work under the condition $k \leq \frac{n-3}{2}$. Thus, we can define $k_{max} = \frac{n-3}{2}$ and get that k_{max} is an upper bound for k, the number of malicious beacons. Both of the algorithms we present in this section find a continuous region r in the intersection of at least $k_{max}+3$ rings and output a point in this region. However, the two algorithms find this continuous region using different methods.

A. Polynomial-time Algorithm

Before we present our polynomial-time algorithm, we require a lemma that gives the relationship between the continuous region and the continuous arcs on its boundary.

Definition 5: A ring is *related* to a continuous arc if the continuous arc is inside but not on the boundary of this ring.

Lemma 3: Suppose that r is a continuous region and c is a continuous arc on the boundry of r. Then r is in the intersection of at least $k+3$ rings if and only if at least $k+2$ rings are related to c.

(We skip the proof of Lemma 3 since it is straightforward.)

The main idea of the polynomial-time algorithm is that, to determine a continuous region in the intersection of at least $k_{max}+3$ rings, we only need to count the number of rings related to each continuous arc and find a continuous arc that at least $k_{max}+2$ rings are related to (It is easy to check if a ring is related to a *continuous* arc by comparing the distance between the arcs end points and the center of the ring to the inner and outer radii of the ring). Once such an arc is found, depending on whether the arc is on an outer boundary circle or an inner boundary circle, a point can be picked from either the inner region or the outer region of the arc respectively. The details of the algorithm are as shown in Algorithm 1.

Lemma 4: The worst-case time complexity of the above algorithm is $O(n^3 \log n)$.

B. Fast Heuristic Algorithm

Although the worst case time complexity of the polynomial-time algorithm is polynomial ($O(n^3 \log n)$) in terms of the total number of beacon nodes, in practice its efficiency needs further improvement. So, we propose our second algorithm, which is heuristic-based and runs even faster in practice.

The heuristic we use is as follows: Note that $k_{max}+3$ is already a large number of rings. Since the region r is

Let S be a set initially containing the two boundary circles of ring R_1.
for $i = 2, \ldots, n$ **do**
  Let S_i be a set initially containing the two boundary circles of ring R_i.
  for each arc in S and each arc in S_i **do**
    if the above two arcs intersect **then**
      Split each of these two arcs using the intersection(s), and replace them in the corresponding arc sets (S or S_i) with the new splitted arcs (result of the splitting operation).
    end if
  end for
  Let $S = S \cup S_i$.
end for
for each arc c_j in S **do**
  Set the corresponding counter λ_j to 0.
  for $i = 1, \ldots, n$ **do**
    if R_i is related to c_j **then**
      $\lambda_j = \lambda_j + 1$.
    end if
  end for
  if $\lambda_j \geq k_{max} + 2$ **then**
    if c_j is on an inner boundary circle **then**
      Output is defined on the side out of this circle.
    else if c_j is on an outer boundary circle **then**
      Output is defined on the side inside this circle
    end if
    Stop the algorithm.
  end if
end for

Algorithm 1: Polynomial-time Algorithm

Count the number of rings intersecting with each ring.
for each ring R_i, in the order of decreasing number of rings intersecting with it **do**
  for each ring $R_j, R_j \neq R_i$, in the order of decreasing number of rings intersecting with it **do**
    Compute the intersection points of the boundary circles of R_i and R_j.
    for $m = 1, \ldots, \kappa$ **do**
      Choose a random intersection point computed above.
      Choose a random point $\overline{O}$ near this intersection point (such that the distance between them is less than ϵ).
      Count the number of rings containing $\overline{O}$.
      if there are at least $k_{max} + 3$ rings containing $\overline{O}$ **then**
        Output $\overline{O}$.
        Stop the Algorithm.
      end if
    end for
  end for
end for

Algorithm 2: Fast Heuristic Algorithm

contained in at least $k_{max}+3$ rings, the rings containing r are intersecting with large numbers of other rings. Therefore, if a ring R_i is intersecting with a large number of rings, it is very likely that R_i contains r. So, we should first consider the rings intersecting with the maximum numbers of other rings. The details of our heuristic algorithm is shown in Algorithm 2.

VII. Extension to 3-Dimensional Localization

So far we have only considered localization in a 2-dimensional space. In certain environments (like mountains, valleys etc.), 3-dimensional localization is needed. In this section, we extend our results to the 3-dimensional space.

We first obtain the necessary condition for robust 3-dimensional localization in the presence of malicious nodes. It turns out that for the 3-dimensional case the maximum number of malicious beacons that can be tolerated is slightly smaller than the 2-dimensional case. The notations and model used here is similar to the 2-dimensional case, except that here the position of each node is represented by three coordinates.

Theorem 4: Suppose that $k \geq \frac{n-3}{2}$. Then, for any distance-based 3-dimensional localization algorithm, for any locations of the beacons, there exists a scenario in which the localization error e is unbounded.

With $k \leq \frac{n-4}{2}$, we can also establish a bounded error for 3-dimensional localization. But to obtain this result, we need to first introduce a few new definitions.

For each beacon B_i, we define a global shell just as we defined the ring for the 2-dimensional case:

$$\tilde{d}_i - \epsilon < dst(B_i, X) < \tilde{d}_i + \epsilon.$$

For simplicity, we still use R_i to denote the above global shell. The globes on the boundary of these shells are called the *boundary globes*; the inner globe of a shell is called an inner boundary globe, while the outer globe of a shell is called an outer boundary circle. A *continuous 3-dimensional region* is part of the space such that its boundary consists of parts of boundary globes, and that no boundary globe goes through its internal. We define a class of 3-dimensional robust localization algorithms as follows: an algorithm is in the class if and only if its output is a point in a continuous 3-dimensional region r such that r is in the intersection of at least $k+4$ global shells.

Definition 6: Consider the planes going through triples of beacons. Denote by $ang(B_{i_1}B_{i_2}B_{i_3}, B_{i'_1}B_{i'_2}B_{i'_3})$ the angle between the two planes $B_{i_1}B_{i_2}B_{i_3}$ and $B_{i'_1}B_{i'_2}B_{i'_3}$—to avoid ambiguity, we require that $0^\circ \leq ang(B_{i_1}B_{i_2}B_{i_3}, B_{i'_1}B_{i'_2}B_{i'_3}) \leq 90^\circ$. The *minimum beacon plane angle* is defined as the minimum of such angles:

$$\alpha^\star = \min_{B_{i_1}, B_{i_2}, B_{i_3}, B_{i'_1}, B_{i'_2}, B_{i'_3}} ang(B_{i_1}B_{i_2}B_{i_3}, B_{i'_1}B_{i'_2}B_{i'_3}).$$

Given the above definitions, we can now state our main (positive) result on 3-dimensional localization.

Theorem 5: For $k \leq \frac{n-4}{2}$, if $\epsilon \ll \min_{B_i} dst(B_i, M)$ and there are neither three beacons in the same line nor four beacons in the same plane, then the error of our robust localization algorithm's output is

$$e < 2\epsilon\sqrt{\frac{1}{\beta^2} + (\frac{1}{\sin\alpha^\star} + \frac{1}{\beta \cdot \tan\alpha^\star})^2}$$

and, $\beta = \min\left\{\sin\left(\frac{\arcsin(\gamma\sin(\alpha/2))}{2}\right), \cos\left(\frac{\arcsin(\gamma\sin(\alpha/2))}{2}\right)\right\}$.

VIII. Evaluation

We have performed extensive experiments to evaluate the performance of our proposed algorithm under varying parameters like beacon node distribution over the deployment area, number of malicious nodes (k), maximum distance measurement error (ϵ) and the number of rings (or distance measurements) used to determine the continuous region. Currently we are not evaluating any network properties like communication overheads for these algorithms because our algorithms are very general and properties like communication overhead would depend on the type of ranging or distance measuring technique used. Thus, we do not use a software network simulator like ns-2 [5] for our simulation experiments. We perform our experiments using C language programs. The network setup for our experiments is as follows: The simulation area is $500\text{m} \times 500\text{m}$. The radio transmission range is 250m. There are 43 beacon nodes and one target node and there is no node mobility. The positions of each of the nodes is selected uniformly over the $500\text{m} \times 500\text{m}$ area.

In our experiments, we have evaluated the heuristic-based algorithm since it is the one with higher practical efficiency. Our experiments have considered two different distributions of distance measurement error: uniform distribution and Normal distribution. For each of these two distributions, we study how the number of malicious beacons (k) and the maximum measurement error (ϵ) influence the localization error and the computational time.

A. Experiments with Uniform Measurement Error

Here, we study the scenario in which the measurement error is uniformly distributed over $[-\epsilon, \epsilon]$. We observe the performance of the heuristic-based localization algorithm for each value of ϵ, when the number of malicious nodes (k) in the network increases. Since the total number of nodes in the network is fixed ($n = 43$), the maximum number of malicious nodes that the algorithm can tolerate is $\frac{43-3}{2}$ (from Theorem 2). We run the simulation of the heuristic-based algorithm for each value of ϵ from 0m to 50m in steps of 10m and each value of k from 0 to 20. In each such run, the beacon and target nodes are assigned new positions, the coordinates of which are uniformly selected over the $500\text{m} \times 500\text{m}$ area. Average localization error (e) is then plotted as an average of the error in localization of the target node over a large number of such runs (around 1000 runs).

From Figure 4, we can see that the average localization error (e) is increasing when ϵ increases, which is very natural. Also, as shown in Figure 4, e is increasing as k increase. This is consistent with our intuition that more malicious beacon nodes should lead to worse localization precision. Figure 5 shows that the average simulation time increases in k, but increases *only very slightly*. This observation is also not surprising since the algorithm is computing the intersection of the same number of rings for each value k. The main reason for the slight increase in simulation time is that a larger number of malicious beacons makes it harder to find the right continuous region in the intersection of $k_{max} + 3$ rings using our heuristic.

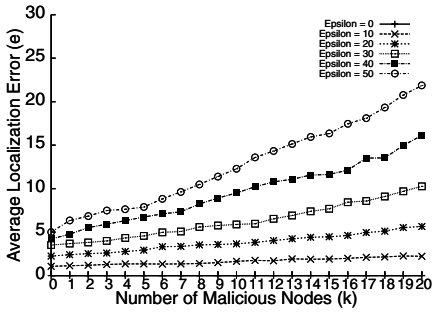

Fig. 4. Localization Error for Uniform Measurement Error

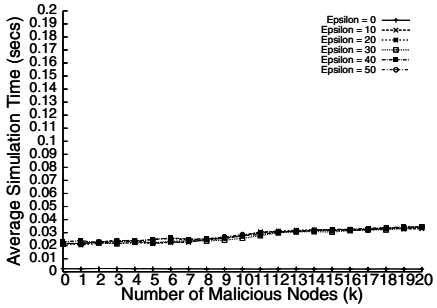

Fig. 5. Simulation Time for Uniform Measurement Error

For all values of k and ϵ, our average localization error is less than 25m and the simulation time is less than 0.035s.

B. Experiments with Normal Measurement Error

To ensure that our evaluation result is not restricted to a uniformly distributed measurement error, we repeat all our experiments with a Normally distributed measurement error. Here we keep all our experiment parameters intact, except that the distance measurement error follows a Normal distribution with mean 0 and variance $\frac{\epsilon}{2}$. However, we need to make sure that the measurement error value is between $[-\epsilon,+\epsilon]$. Therefore, we modify the distribution such that the probability density outside $[-\epsilon, +\epsilon]$ becomes 0; the probability density inside the interval $[-\epsilon, +\epsilon]$ is scaled up a little accordingly.

Figure 6 shows the average localization error for each pair of (k, ϵ) when the measurement error follows the Normal distribution. Figure 7 shows the corresponding simulation time. We can see that the curves are analogous to those in Figures 4 and 5 respectively (except that the localization error increases more slowly in k). Therefore, we can claim that

our evaluation results are valid for different distributions of measurement errors.

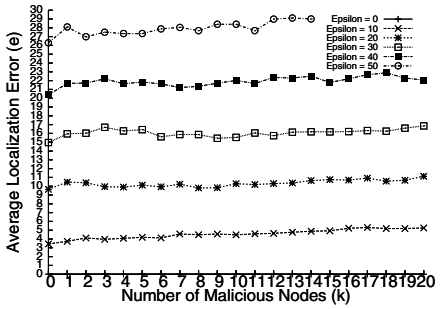

Fig. 6. Localization Error for Normal Measurement Error

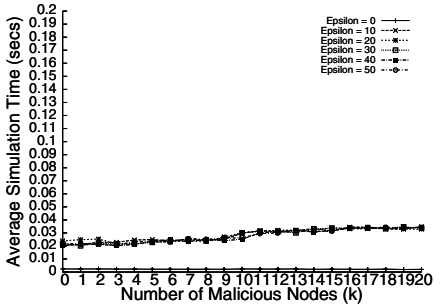

Fig. 7. Simulation time for Normal Measurement Error

IX. Conclusion and Open Question

In this paper, we have theoretically treated the problem of robust distance-based localization in the presence of malicious beacon nodes. We derive a necessary and sufficient condition for having a bounded localization error and identify a class of algorithms that achieve such a bounded error. In addition to this, we propose two algorithms in this class. First, a polynomial-time algorithm that guarantees to finish in polynomial time even in the worst case. We also propose a fast heuristic algorithm that is suitable from the practical standpoint. Also, we extend our current results in 2-dimensional localization to the 3-dimensional case. Finally, through computer simulations we show that the heuristic-based algorithm provides good localization precision with a very small time cost and that it works under different distributions of the distance measurement errors.

An open question is what is the best algorithm to find the intersection of rings, in terms of worst-case complexity and in terms of average computational time? We leave this question for future work.

References

[1] P. Bahl and V. N. Padmanabhan. Radar: an in-building RF-based user location and tracking system. In *IEEE INFOCOM Conference Proceedings*, pages 775–784. IEEE Communications Society, March 2000.

[2] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, pages 28–34, Oct 2000.

[3] L. Doherty, L. E. Ghaoui, and K. S. J. Pister. Convex position estimation in wireless sensor networks. In *IEEE INFOCOM Conference Proceedings*, Anchorage, April 2001. IEEE Communications Society.

[4] Lei Fang, Wenliang Du, and Peng Ning. A beacon-less location discovery scheme for wireless sensor networks. In *IEEE INFOCOM Conference Proceedings*. IEEE Communications Society, March 2005.

[5] Marc Greis. *Tutorial for the Network Simulator "ns"*. VINT group, 2005. http://www.isi.edu/nsnam/ns/.

[6] Tian He, Chengdu Huang, Brian M. Blum, John A. Stankovic, and Tarek Abdelzaher. Range-free localization schemes for large scale sensor networks. In *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 81–95, New York, NY, USA, 2003. ACM Press.

[7] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer Verlag, 1997.

[8] Xiang Ji and Hongyuan Zha. Sensor positioning in wireless ad-hoc sensor. networks using multidimensional scaling. In *Proceedings of IEEE INFOCOM 2004*, March 2004.

[9] Waldir Ribeiro Pires Jr., Thiago H. de Paula Figueiredo, Hao Chi Wong, and Antonio A.F. Loureiro. Malicious node detection in wireless sensor networks. In *18th International Parallel and Distributed Processing Symposium, 2004. Proceedings.*, page 24. IEEE Computer Society, April 2004.

[10] Zang Li, Wade Trappe, Yanyong Zhang, and Badri Nath. Robust statistical methods for securing wireless localization in sensor networks. In *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*, page 12, Piscataway, NJ, USA, 2005. IEEE Press.

[11] Donggang Liu, Peng Ning, and Wenliang Du. Attack-resistant location estimation in sensor networks. In *The Fourth International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pages 99–106. ACM SIGBED and IEEE Signal Processing Society, April 2005.

[12] Donggang Liu, Peng Ning, and Wenliang Du. Detecting malicious beacon nodes for secure location discovery in wireless sensor networks. In *The 25th International Conference on Distributed Computing Systems (ICDCS '05)*, pages 609–619. IEEE Computer Society, June 2005.

[13] David Moore, John Leonard, Daniela Rus, and Seth Teller. Robust distributed network localization with noisy range measurements. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 50–61, New York, NY, USA, 2004. ACM Press.

[14] D. Niculescu and B. Nath. DV based positioning in ad hoc networks. *Journal of Telecommunication Systems*, 2003.

[15] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *The Sixth Annual International Conference on Mobile Computing and Networking(MOBICOM)*, pages 32–43. ACM SIGMOBILE, August 2000.

[16] Saikat Ray, Rachanee Ungrangsi, Francesco de Pellegrini, Ari Trachtenberg, and David Starobinski. Robust location detection in emergency sensor networks. In *IEEE INFOCOM Conference Proceedings*, pages 1044–1053, San Francisco, March 2003. IEEE Communications Society.

[17] Naveen Sastry, Umesh Shankar, and David Wagner. Secure verification of location claims. In *WiSe '03: Proceedings of the 2003 ACM workshop on Wireless security*, pages 1–10, New York, NY, USA, 2003. ACM Press.

[18] Yi Shang, Wheeler Ruml, Ying Zhang, and Markus Fromherz. Localization from connectivity in sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 15(11):961–974, 2004.

[19] Radu Stoleru and John A. Stankovic. Probability grid: A location estimation scheme for wireless sensor networks. In *IEEE Sensor and Ad Hoc Communications and Networks*, pages 430–438. IEEE Communications Society, October 2004.

[20] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. *ACM Transaction on Information Systems*, pages 91–102, Jan 1992.

[21] Kiran Yedavalli, Bhaskar Krishnamachari, Sharmila Ravula, and Bhaskar Srinivasan. Ecolocation: A sequence based technique for rf-only localization in wireless sensor networks. In *Proceedings of the Fourth International Conference on Information Processing in Sensor Networks (IPSN '05)*, Los Angeles, CA, USA, April 2005.