

Copyright © 2002, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**DISTRIBUTED LOCALIZATION IN
WIRELESS AD HOC NETWORKS**

by

Slobodan N. Simić and Shankar Sastry

Memorandum No. UCB/ERL M02/26

6 September 2002

**DISTRIBUTED LOCALIZATION IN
WIRELESS AD HOC NETWORKS**

by

Slobodan N. Simić and Shankar Sastry

Memorandum No. UCB/ERL M02/26

6 September 2002

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

DISTRIBUTED LOCALIZATION IN WIRELESS AD HOC NETWORKS

SLOBODAN N. SIMIĆ AND SHANKAR SASTRY

ABSTRACT. We present a distributed algorithm for localization of nodes in a discrete model of a random ad hoc wireless network. We compute the expected value of the position estimate and the probability that the estimate is perfect (one cell in size). This leads to bounds of the average complexity of the algorithm at each node.

1. INTRODUCTION

This article addresses the problem of localization of nodes in a wireless ad hoc communication network. It is motivated by the Sensorwebs and Smart Dust [KKP] projects at UC Berkeley, whose aim is to develop a unified framework for distributed sensor networks.

Recent advances in MEMS, computing, and communication technology have fomented the emergence of massively distributed, wireless sensor networks consisting of hundreds or thousands of nodes. Each node is able to sense the environment, perform simple computations, and communicate with its peers or to an external observer. The challenges these networks present are far beyond the reach of the current theory and algorithms.

One way of deploying a sensor network is to scatter the nodes throughout some region of interest. This makes the network topology random. Since there is no a priori communication protocol, the network is ad hoc. The first task that has to be solved is to localize the nodes, i.e., to compute their positions in some fixed coordinate system. Since most applications (such as tracking an object moving through the network, environmental monitoring, etc.) depend on a successful localization, it is of great importance to design scalable localization algorithms and provide error estimates that will enable us to choose optimal network parameters before deployment.

Due to high long range communication costs and low battery power, it is natural to seek decentralized, distributed algorithms for sensor networks. This means that instead of relaying data to a central location which does all the computing, the nodes process information in a collaborative, distributed way. For instance, they can form computational clusters, based on their distance from each other. The outcome of these distributed, local computations is stored in local memory and can then be, when necessary, relayed to a centralized computing unit. *Robustness* to node failures is another reason to seek distributed rather than centralized algorithms.

Previous work. Since many applications of sensor networks depend on the knowledge of positions of its nodes, much work has been done on the problem of localization. We mention

Date: September 5, 2002.

The first author was partially supported by the NSF grant ITR/SI: Societal Scale Information Systems: Technologies, Design and Applications, number EIA-0122599.

only a few references. In [GBEE02], Girod et al. explore the use of acoustic ranging techniques and commercial off-the-shelf components in localization. Bulusu et al. [BEGH01] work with multiple sensor modalities and study the role of density of nodes in localization. In [NN01], Niculescu et al. propose a distributed hop by hop algorithm which works as an extension of GPS positioning. Priyantha et al. [PCB00] present a location-support system (*Cricket*) for in-building, mobile applications. For a more extensive overview of localization techniques, we refer the reader to the survey article [ASSC02].

We point out that the main purpose of this paper is to rigorously analyze the accuracy and complexity of our algorithm from a probabilistic point of view; its purpose is *not* to deal with technical details of time synchronization, data fusion, communication protocols, etc. We believe that this approach fills a void in the literature as none of the references known to the authors rigorously studies mathematical aspects of localization in sensor networks.

This paper is organized as follows. In Section 2, we establish the setting and introduce the basic terminology and notation. In Section 3, we discuss the basic localization procedure used in the algorithm described in Section 4, and provide basic probabilistic error estimates and simulation results. The proofs are supplied in the Appendix.

The first author would like to thank Sekhar Tatikonda for reading a preliminary version of the paper and providing useful comments.

2. PRELIMINARIES

In this section we introduce the basic framework, terminology, and notation.

We assume that in a square region $Q = [0, s] \times [0, s]$, called the *region of operations*, we randomly scatter N nodes, S_1, \dots, S_N , each of which is equipped with an RF transceiver with *communication range* $r > 0$. In other words, a node S_i can communicate with every node which lies in its *communication region*, which is the disk with radius r centered at S_i . Each node has a unique ID which is a number between 1 and N . The nodes form an *ad hoc network* \mathcal{N} in which there is an edge between S_i and S_j if their distance is less than r . We will call this the *continuous model*. Even though it is a rather simplified model of how the network is formed, we adopt it because it leads to an easier analytical treatment. However, instead of a disk of radius r , the communication range of a node could instead be an annulus (in case a node is able to decide when a nearby node is at a distance greater than some threshold, based, e.g., on signal strength), an angular sector (if a node is equipped, say, with laser transmitters and receivers that can scan through some angle) or an intersection of an annulus and an angular sector. For more information on these types of constraints, please see [Doh00].

We assume that a certain positive number K of nodes know their location in Q , i.e., they are able to compute their position relative to some fixed coordinate system in Q ; in practice this can be achieved by equipping K nodes with GPS or a priori (meaning before deploying the ad hoc network) placing in Q a certain number of *beacons* which can serve to compute the position of the nodes which are within certain distance from them. We call nodes that know their position *known nodes* and all other ones, *unknown nodes*. Furthermore, each node has communication as well as sensing capabilities [KKP].

The problems we address in this paper are:

- Design a *distributed algorithm* for localization of nodes in \mathcal{N} .

- Estimate the *complexity* and *error* of the above algorithm.
- Find an optimal number of known nodes depending on Q , N , and r , which minimizes the error of the algorithm.

Centralized algorithms for localization were studied in [Doh00, DPG01]. The reason we are interested in a distributed rather than a centralized solution is that we envision a massively distributed network in which communication with a centralized computer is expensive both because the power supply of each node is very limited and long-range multi-hop data transmission is costly and often inefficient.

However, one quickly discovers that obtaining analytical estimates even in this simple setting can be rather challenging. Furthermore, in the design of a decentralized algorithm relying on node-based data processing, one must take into account that nodes have very limited computational power. This motivates the following discrete approach to the above problems.

Let $n > 0$ be an integer. Partition Q into n^2 congruent squares called *cells* of area $(s/n)^2$ and suppose that for every known node S , we are only interested in finding the cell which contains S . To make this problem tractable, we make a simplifying assumption that the *communication range is ρ cells in the max metric*, defined by $\text{dist}_\infty((i, j), (i', j')) = \max(|i - i'|, |j - j'|)$. It is possible for several nodes to lie in the same cell. We call this the *discrete model* of the network. For example, we can take

$$(1) \quad \rho = \left\lceil \frac{nr}{s\sqrt{2}} \right\rceil,$$

where $\lceil x \rceil$ denotes the integer part of x . This means that each node S can communicate with every node lying in the square centered at S and containing $(2\rho + 1)^2$ cells. Since all our results are independent of the choice of ρ as a function of r and n , we can clearly select a different value for it. The situation where r is fixed and $s, n \rightarrow \infty$ corresponds to increasing the size of the region of operations, while the situation where r/n and s stay constant while $n \rightarrow \infty$ corresponds to refining the position estimate. We usually think of n as large and r as much smaller than n . In particular, $2\rho + 1 < n$.

The last question posed above can now be reformulated in the discrete model as:

- Find an optimal number of known nodes, depending on n and ρ , which minimizes the error of the algorithm.

It still remains to define what we mean by “the error” of an algorithm, i.e., to define the “metric” by which we measure the accuracy of the algorithm. We will do this in the following section.

3. LOCALIZATION PROCEDURE AND ERROR ESTIMATES

We now describe the underlying procedure for localization of nodes which can be implemented in both centralized and distributed setting. A distributed implementation is given in the following section.

Let S be a random node whose position is unknown. Let S_{k_1}, \dots, S_{k_m} , for some $m > 0$, be its *neighbors* in the network \mathcal{N} whose positions are known. Two nodes S_1, S_2 are neighbors if $\text{dist}_\infty(S_1, S_2) \leq \rho$. We assume that each S_{k_i} knows in which cell it lies, i.e., it is aware of a pair of numbers (x_i, y_i) , where $x_i, y_i \in \{1, 2, \dots, n\}$.

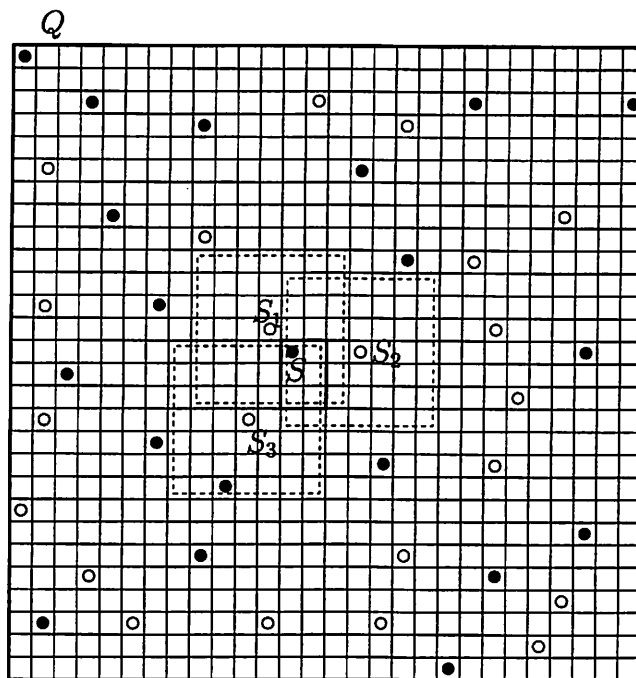


FIGURE 1. Discrete model. Filled circles denote unknown nodes. The dashed line bounds the communication range of the node at its center, with $\rho = 3$.

We will now use the following notation. For integers $1 \leq a < b \leq n$, $1 \leq c < d \leq n$, the “rectangle” $[a, b] \times [c, d]$ will denote the union of all cells with grid coordinates (i, j) , where $a \leq i \leq b$ and $c \leq j \leq d$.

Let

$$(2) \quad B_i = [x_i - \rho, x_i + \rho] \times [y_i - \rho, y_i + \rho].$$

This is the (discrete) communication region of S_{k_i} . Then $S \in B_i$, for all $1 \leq i \leq m$ and therefore

$$S \in Q \cap \bigcap_{i=1}^m B_i.$$

It is easy to see that

$$B_i \cap B_j = [\max(x_i, x_j) - \rho, \min(x_i, x_j) + \rho] \times [\max(y_i, y_j) - \rho, \min(y_i, y_j) + \rho].$$

This simple formula is, in fact, the main technical reason we choose to work in this discrete setting: one needs much less computing power to calculate the intersection of two squares (with sides parallel to coordinate axes) than to compute the intersection of two discs – only the operations of addition, subtraction, min and max are needed.

Thus

$$S \in Q \cap [x_+ - \rho, x_- + \rho] \times [y_+ - \rho, y_- + \rho],$$

where $x_+ = \max(x_1, \dots, x_m)$ and $x_- = \min(x_1, \dots, x_m)$, and similarly for y_i 's. Finally, since $Q = [1, n] \times [1, n]$, we obtain the *estimate* of the position of S :

$$(3) \quad S \in [\max(x_+ - \rho, 1), \min(x_- + \rho, n)] \times [\max(y_+ - \rho, 1), \min(y_- + \rho, n)].$$

On the highest level of abstraction we now have the following algorithm (implementable in both centralized and distributed setting) for each unknown node S :

Step A: *Gather the information about the positions of the known neighbors of S .*

Step B: *Compute an estimate of the position of S via (3).*

Let A_S be the number of cells in the rectangle on the right hand side of (3), i.e.,

$$(4) \quad A_S = \{\min(x_- + \rho, n) - \max(x_+ - \rho, 1) + 1\} \{\min(y_- + \rho, n) - \max(y_+ - \rho, 1) + 1\}.$$

Let K denote the total number of nodes which know their own position. We will assume that K is given and fixed; later on, we discuss how we can deterministically or randomly generate a sufficiently large number of known nodes. We assume that the position of each node is random and uniformly distributed in $[1, n] \times [1, n]$. Then A_S is a random variable with value between 1 and n^2 . In the remainder of this section we address the following questions about A_S :

- What is the expectation $E(A_S)$ of A_S ?
- What is the probability that $A_S = 1$ cell, i.e., that the estimate is perfect?

Having answered these questions, we turn to minimizing $|E(A_S) - 1|$ and maximizing $P(A_S = 1)$ as functions of K to find the optimal value for K .

Denote by Q_ρ the square consisting of cells in Q which are at distance $\geq \rho$ from the boundary. That is, $Q_\rho = [\rho + 1, n - \rho]^2$. Then we claim:

Theorem 3.1. *Let S be a node randomly picked from Q_ρ . The expectation of A_S is*

$$E(A_S) = 1 + 4 \sum_{k=1}^{2\rho} \sum_{l=1}^{2\rho+1} \left[1 - \frac{(2\rho+1)^2 - kl}{n^2} \right]^K.$$

Therefore, with n, ρ fixed,

$$\lim_{K \rightarrow \infty} E(A_S) = 1,$$

that is, the expectation of the size of the estimate tends to one, the perfect estimate, as the number of known nodes tends to infinity.

The proof is provided in the Appendix. Observe that Q_ρ occupies most of Q , since we assume that ρ is much smaller than n .

Given $\epsilon > 0$ and $S \in Q_\rho$, let $K_\epsilon = K_\epsilon(n, \rho)$ be the minimum of all numbers K_0 such that

$$(5) \quad |E(A_S) - 1| < \epsilon,$$

for all $K \geq K_0$. Then

Corollary 3.1.1. *The minimal density of known nodes necessary to achieve (5) satisfies*

$$(6) \quad \frac{K_\epsilon(n, \rho)}{n^2} \leq \frac{\log[8\rho(2\rho+1)] - \log \epsilon}{2\rho+1}.$$

Proof: Consider the estimate in Theorem 3.1. Its right hand side is largest when $k = 2\rho$ and $l = 2\rho + 1$, so

$$E(A_S) \leq 1 + 8\rho(2\rho+1) \left(1 - \frac{2\rho+1}{n^2} \right)^K.$$

Therefore, $E(A_S) - 1 < \epsilon$ if

$$(7) \quad K > \frac{-\log[8\rho(2\rho+1)] + \log \epsilon}{\log\left(1 - \frac{2\rho+1}{n^2}\right)}.$$

Thus K_ϵ is less than or equal to the right hand side of (7). Using the fact that $\log(1+x) \leq x$ ($x > -1$), we obtain that

$$K_\epsilon(n, \rho) \leq n^2 \frac{\log[8\rho(2\rho+1)] - \log \epsilon}{2\rho+1},$$

as claimed. ■

We denote the right hand side of (6) by $\delta_\epsilon(\rho)$ and interpret it as the *critical density* of known nodes necessary for the expectation of the estimate to be ϵ -close to the perfect estimate.

Denote by H_S the number of known neighbors of S .

Theorem 3.2. *Suppose that $S \in Q_\rho$.*

(a) *The conditional probability that $A_S = 1$ given that $H_S = m$ is*

$$P(A_S = 1 | H_S = m) = \left[1 - \left(\frac{2\rho}{n}\right)^m\right]^4.$$

(b) *The probability that $A_S = 1$ is*

$$P(A_S = 1) = \sum_{m=0}^K \binom{K}{m} p^m q^{K-m} \left[1 - \left(\frac{2\rho}{n}\right)^m\right]^4,$$

where $p = (2\rho+1)^2/n^2$ and $q = 1 - p$.

Note that $P(A_S = 1)$ depends on n , ρ , and K . Its graph for $n = 50$ is given above (Fig. 2).

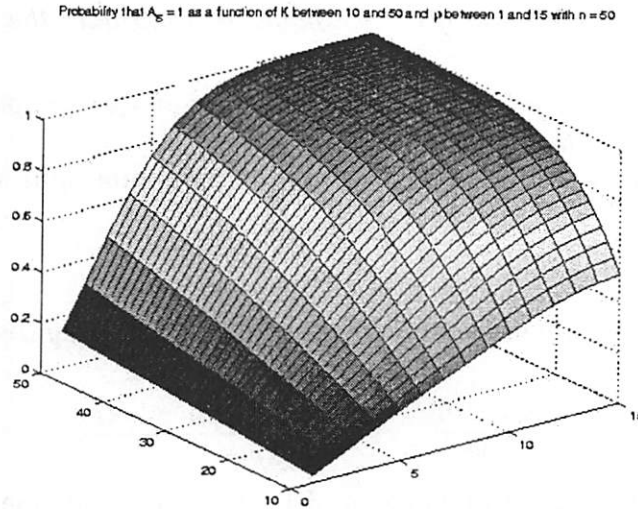


FIGURE 2. Probability that $A_S = 1$ as a function of ρ and K .

Corollary 3.2.1. *We have*

$$P(A_S = 1) \geq \left[1 - \left(\frac{2\rho}{n} \right)^K \right]^4.$$

Therefore, $P(A_S = 1) \rightarrow 1$, as $K \rightarrow \infty$. Given $p_* \in (0, 1)$, $P(A_S = 1) > p_*$ if

$$K > \frac{\log(1 - p_*^{1/4})^{-1}}{\log(n/2\rho)}.$$

There are several ways of generating nodes with known positions. One is to equip a certain number of nodes with GPS. Another is to a priori place in Q a certain number of *beacons* whose positions are known. They can be equipped with some relatively sophisticated device capable of localizing objects which lie at a distance $\leq R$ meters away. Then, if a node lands within R meters from a beacon, its position will be known to that beacon and can be communicated to the node.

Suppose that in the latter scenario, the beacons are capable of exactly localizing any node in a region of area $\alpha|Q|$, for some $0 < \alpha < 1$. Then it is not difficult to see that if N nodes are randomly scattered in Q (assuming uniform distribution), the expected value of K will be

$$E(K) = N\alpha.$$

Furthermore,

Proposition 3.1. *Assuming that K is fixed and given and that S is an unknown node, the expectation of H_S is*

$$E(H_S) = \frac{K}{n^4} [n + \rho(2n - \rho - 1)]^2.$$

Proof: Let the coordinates of S be (x, y) . The communication region of S is

$$\begin{aligned} B(x, y) &= [x - \rho, x + \rho] \times [y - \rho, y + \rho] \cap Q \\ &= [\max(x - \rho, 1), \min(x + \rho, n)] \times [\max(y - \rho, 1), \min(y + \rho, n)]. \end{aligned}$$

Denote by $p(x, y)$ the probability that a node lands in the communication region of $S = (x, y)$. It is easy to see that $p(x, y) = |B(x, y)|/|Q|$, where $|B(x, y)|$ denotes the number of cells in $B(x, y)$. Set $q(x, y) = 1 - p(x, y)$. The probability that $H_S = m$ given $S = (x, y)$ is $\binom{K}{m} p(x, y)^m q(x, y)^{K-m}$. By a standard calculation in elementary probability it then follows that $E(H_S|S = (x, y)) = Kp(x, y)$. Therefore, by Lemma 6.1(a),

$$\begin{aligned} E(H_S) &= E(E(H_S|S = (x, y))) \\ &= \sum_{(x,y) \in Q} E(H_S|S = (x, y)) P(S = (x, y)) \\ &= \frac{1}{n^2} \sum_{x=1}^n \sum_{y=1}^n Kp(x, y) \\ &= \frac{K}{n^4} \left\{ \sum_{x=1}^n [\min(x + \rho, n) - \max(x - \rho, 1) + 1] \right\}^2. \end{aligned}$$

Since

$$\begin{aligned} \sum_{x=1}^n \min(x + \rho, n) &= \sum_{x=1}^{n-\rho-1} (x + \rho) + \sum_{x=n-\rho}^n n \\ &= \frac{(n - \rho - 1)(n + \rho) + 2n(\rho + 1)}{2} \end{aligned}$$

and

$$\begin{aligned} \sum_{x=1}^n \max(x - \rho, 1) &= \sum_{x=1}^{\rho+1} 1 + \sum_{x=\rho+2}^n (x - \rho) \\ &= n + \frac{(n - \rho - 1)(n - \rho)}{2}, \end{aligned}$$

we obtain that $\sum_{x=1}^n [\min(x + \rho, n) - \max(x - \rho, 1)] = (2n - \rho - 1)\rho$. This completes the proof. ■

4. DISTRIBUTED ALGORITHM FOR LOCALIZATION

In this section we present a simple distributed algorithm for localization, based on the ideas in the previous section.

Let S be an arbitrary unknown node in \mathcal{N} . The localization algorithm LOC_S at S then goes as follows.

Step 1: *INITIALIZE* the estimate: $L_S = Q$.

Step 2: *SEND* “Hello, can you hear me?”

Each known neighbor sends back $(1, a, b)$, where (a, b) is its grid position, while each unknown neighbor sends $(0, 0, 0)$.

Step 3: For each response $(1, a, b)$, *UPDATE* the estimate by

$$L_S := L_S \cap [a - \rho, a + \rho] \times [b - \rho, b + \rho].$$

Step 4: *STOP* when all responses have been received. The position estimate is L_S .

Proposition 4.1. *The average complexity of LOC_S is $O(E(H_S))$. In particular, if ρ and n are fixed and K is chosen so that*

(a) $|E(A_S) - 1| < \epsilon$, or

(b) $P(A_S = 1) > 1 - \epsilon$,

then the average complexity is $O(\log \frac{1}{\epsilon})$, as $\epsilon \rightarrow 0$.

Proof: Both the number of communication steps and the number of operations in LOC_S are multiples of H_S , the number of known neighbors of S . If $K = K_\epsilon(n, \rho)$, then $K \leq n^2 \delta_\epsilon(\rho)$, so by Proposition 3.1, $E(H_S) \leq C \log(1/\epsilon)$, proving (a).

To prove (b), set $p_\star = 1 - \epsilon$, use the last estimate in Corollary 3.2.1, and observe that

$$\log \frac{1}{1 - (1 - \epsilon)^{1/4}} = O(\log \frac{1}{\epsilon}). \quad \blacksquare$$

4.1. **Simulation results.** For $\rho = 10$ and $50 \leq n \leq 100$, we chose K as in (6) with $\epsilon = 1$:

$$K = n^2 \frac{\log[8\rho(2\rho + 1)]}{2\rho + 1},$$

and randomly created 30 networks of $N = K + 100$ nodes with these parameters. The average size of the position estimate is shown in Figure 3.

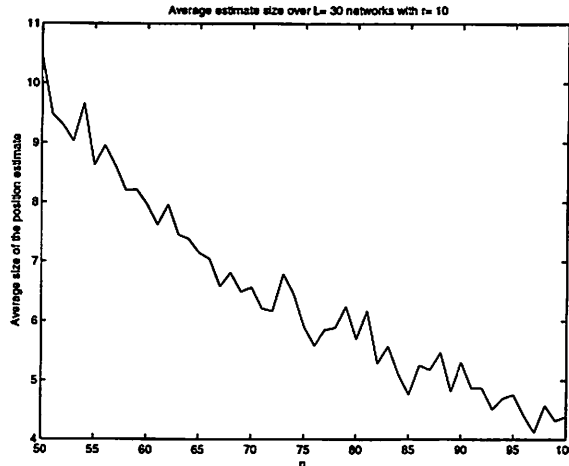


FIGURE 3. Average size of the position estimate.

Observe that the average error of the algorithm steadily decreases as n increases. Since ρ is fixed, this corresponds to increasing the size of the region of operations.

5. CONCLUSION AND FUTURE WORK

The main contributions of this paper are a distributed algorithm for localization of nodes in a wireless ad hoc communication network and estimates of its error. There are many possible improvements of the algorithm. For example, an unknown node could only query *some* of its neighbors and use correlation coding developed in [KDR01]. This would reduce communication costs but increase computation. The algorithm can also be iterated to exploit the newly obtained position estimates of unknown nodes. We plan to address these issues in future work.

6. APPENDIX

6.1. **Proof of Theorem 3.1.** We will need the following two results from basic probability. The reader is referred to the literature for a proof.

Lemma 6.1 ([GS97]). (a) *Let X and Y be two random variables defined on the same probability space. The conditional expectation $\psi(x) = E(Y|X = x)$ satisfies*

$$E(\psi(X)) = E(Y).$$

(b) *If a random variable X has mass function f and $g : \mathbb{R} \rightarrow \mathbb{R}$, then*

$$E(g(X)) = \sum_x g(x)f(x),$$

whenever this sum is absolutely convergent.

Define a function $\phi_m : \mathbb{R}^{2m} \rightarrow \mathbb{R}$ by

$$\phi_m(x_1, \dots, x_m, y_1, \dots, y_m) = (2\rho + 1 + x_- - x_+)(2\rho + 1 + y_- - y_+),$$

where $x_+ = \max(x_1, \dots, x_m)$, $x_- = \min(x_1, \dots, x_m)$, and similarly for y 's.

Now let S be a randomly chosen node with position $(x, y) \in Q_\rho$. Assume $H_S = m$, i.e., S has m known neighbors, and let their coordinates be represented by random variables (X_i, Y_i) , for $1 \leq i \leq m$. Since $\rho + 1 \leq x, y \leq n - \rho$, it follows from (4) that

$$\begin{aligned} A_S &= \{\min(x_- + \rho, n) - \max(x_+ - \rho, 1) + 1\} \{\min(y_- + \rho, n) - \max(y_+ - \rho, 1) + 1\} \\ &= \{x_- + \rho - (x_+ - \rho) + 1\} \{y_- + \rho - (y_+ - \rho) + 1\} \\ &= \phi_m(x_1, \dots, x_m, y_1, \dots, y_m). \end{aligned}$$

In other words, given H_S and (X_i, Y_i) 's, as a random variable, A_S can be represented as

$$A_S = \phi_m(X_1, \dots, X_m, Y_1, \dots, Y_m).$$

Therefore, by Lemma 6.1 (a),

$$\begin{aligned} E(A_S | S = (x, y), H_S = m) &= E(\phi_m(X_1, \dots, X_m, Y_1, \dots, Y_m) | S = (x, y), H_S = m) \\ &= \sum_{(x_1, y_1), \dots, (x_m, y_m) \in B} \phi_m(x_1, \dots, x_m, y_1, \dots, y_m) \times \\ &\quad \times P(X_1 = x_1, \dots, Y_m = y_m | S = (x, y), H_S = m) \\ &= \frac{1}{|B|^m} \sum_{(x_1, y_1), \dots, (x_m, y_m) \in B} \phi_m(x_1, \dots, x_m, y_1, \dots, y_m), \end{aligned}$$

where $B = [x - \rho, x + \rho] \times [y - \rho, y + \rho]$ is the communication region of S .

The probability that S has m known neighbors, where $0 \leq m \leq K$, is

$$\binom{K}{m} p^m q^{K-m},$$

where $p = |B|/|Q| = (2\rho + 1)^2/n^2$ and $q = 1 - p$. It now follows from Lemma 6.1 (b) that

$$\begin{aligned} E(A_S | S = (x, y)) &= E(E(A_S | S = (x, y), H_S)) \\ &= \sum_{m=0}^K E(A_S | S = (x, y), H_S = m) \binom{K}{m} p^m q^{K-m} \\ &= \sum_{m=0}^K \frac{1}{|B|^m} \binom{K}{m} q^{K-m} \sum_{(x_1, y_1), \dots, (x_m, y_m) \in B} \phi_m(x_1, \dots, x_m, y_1, \dots, y_m) \\ &= \sum_{m=0}^K \binom{K}{m} \frac{1}{n^{2m}} q^{K-m} \sum_{(x_1, y_1), \dots, (x_m, y_m) \in B} \phi_m(x_1, \dots, x_m, y_1, \dots, y_m). \end{aligned}$$

Note that for all $1 \leq i \leq m$, $x - \rho \leq x_i \leq x + \rho$ and $y - \rho \leq y_i \leq y + \rho$.

Lemma 6.2.

$$\sum_{(x_1, y_1), \dots, (x_m, y_m) \in B} \phi_m(x_1, \dots, x_m, y_1, \dots, y_m) = \left[(2\rho + 1)^m + 2 \sum_{k=1}^{2\rho} k^m \right]^2$$

Proof: It suffices to prove

$$(8) \quad \sum_{x-\rho \leq x_1, \dots, x_m \leq x+\rho} (2\rho + 1 + x_- - x_+) = (2\rho + 1)^m + 2 \sum_{k=1}^{2\rho} k^m.$$

Consider $\sum x_-$. For $x - \rho \leq k \leq x + \rho$, $x_- = k$ if and only if for some i , $x_i = k$, and all other x_j are in $\{k, k + 1, \dots, x + \rho\}$. If the number of such i 's is l , then there are a total of $\binom{m}{l} (x + \rho - k)^{m-l}$ possibilities. Therefore, the total number of choices for x_1, \dots, x_m in $[x - \rho, x + \rho]$ such that $x_- = k$ is

$$\sum_{l=1}^m \binom{m}{l} (x + \rho - k)^{m-l} = (1 + x + \rho - k)^m - (x + \rho - k)^m.$$

A simple telescopic calculation shows that

$$(9) \quad \begin{aligned} \sum_{x-\rho \leq x_1, \dots, x_m \leq x+\rho} x_- &= \sum_{k=x-\rho}^{x+\rho} k [(1 + x + \rho - k)^m - (x + \rho - k)^m] \\ &= (x - \rho)(2\rho + 1)^m + \sum_{k=1}^{2\rho} k^m. \end{aligned}$$

Similarly one can show that

$$\sum_{x-\rho \leq x_1, \dots, x_m \leq x+\rho} x_+ = (x + \rho)(2\rho + 1)^m - \sum_{k=1}^{2\rho} k^m.$$

Combining the above two results with an easy observation that

$$\sum_{x-\rho \leq x_1, \dots, x_m \leq x+\rho} (2\rho + 1) = (2\rho + 1)^{m+1},$$

we obtain (8). ■

Observe that the sum in Lemma 6.2 does *not* depend on (x, y) .

After a tedious but elementary calculation (consisting mostly of rearranging the terms) which we omit, we obtain the result of Theorem 3.1. ■

6.2. Proof of Theorem 3.2. (a) Assume that $S \in Q_\rho$ and $H_S = m$. Let $X_1, \dots, X_m, Y_1, \dots, Y_m$ be random variables representing the x - and y - coordinates of the known neighbors of S . Define

$$X_- = \min(X_1, \dots, X_m), \quad X_+ = \max(X_1, \dots, X_m),$$

and similarly for Y_- and Y_+ . Recall that $A_S = \phi_m(X_1, \dots, X_m, Y_1, \dots, Y_m)$. Therefore, $A_S = 1$ if and only if $X_+ - X_- = Y_+ - Y_- = 2\rho$.

Let us compute $P(X_+ - X_- = 2\rho | H_S = m)$. Since $x - \rho \leq X_i \leq x + \rho$, for all i , $X_+ - X_- = 2\rho$ is possible if only if $X_- = x - \rho$ and $X_+ = x + \rho$. The event that $X_- = x - \rho$, given $H_S = m$, is complementary to the event that all X_i 's are greater than $x - \rho$ (and less than $x + \rho$, of course). Therefore,

$$\begin{aligned} P(X_- = x - \rho | H_S = m) &= 1 - P(x - \rho + 1 \leq \text{all } X_i \leq x + \rho | H_S = m) \\ &= 1 - \left(\frac{2\rho}{n}\right)^m. \end{aligned}$$

The same equality holds for $P(X_+ = x + \rho | H_S = m)$, and analogously for Y 's. Then

$$\begin{aligned} P(A_S = 1 | H_S = m) &= P(X_- = x - \rho | H_S = m) P(X_+ = x + \rho | H_S = m) \\ &\times P(Y_- = y - \rho | H_S = m) P(Y_+ = y + \rho | H_S = m) \end{aligned}$$

proves (a).

(b) Follows from (a) and the fact that the probability that a node $S \in Q_\rho$ has m known neighbors is $\binom{K}{m} p^m q^{K-M}$, where p, q were defined above. ■

REFERENCES

- [ASSC02] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38:393–422, 2002.
- [BEGH01] N. Bulusu, D. Estrin, L. Girod, and J. Heideman. Scalable coordination for wireless sensor networks: self-configuring localization systems. In *Proc. of the Sixth International Symposium on Communication Theory and Applications (ISCTA 2001)*, 2001.
- [Doh00] L. Doherty. Algorithms for position and data recovery in wireless sensor networks. Master's thesis, UC Berkeley, 2000.
- [DPG01] L. Doherty, K. S. J. Pister, and L. El Ghaoui. Convex position estimation in wireless sensor networks. In *Infocom 2001*, Anchorage, AK, 2001.
- [GBEE02] L. Girod, V. Bychovskiy, J. Elson, and D. Estrin. Locating tiny sensors in time and space: a case study. In *Proc. of ICCD 2002*, 2002. to appear.
- [GS97] G.R. Grimmett and D.R. Strizaker. *Probability and Random Processes*. Oxford University Press, second edition, 1997.
- [KDR01] J. Kusuma, L. Doherty, and K. Ramchandran. Distributed source coding for sensor networks. In *Proc. IEEE Conf. on Image Processing (ICIP)*, October 2001.
- [KKP] J.M. Kahn, R.H. Katzand, and K.S.J. Pister. Mobile networking for Smart Dust. In *ACM/IEEE Intl. Conf. on Mobile Computing and Networking*, Seattle, WA, August 1999.
- [NN01] D. Niculescu and Badri Nath. Ad hoc positionin system (aps). In *Proc. of GLOBECOM 2001*, San Antonio, TX, November 2001.
- [PCB00] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Proc. of 6th ACM MobiCom*, Boston, MA, 2000.

DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCES, UNIVERSITY OF CALIFORNIA, BERKELEY, CA 94720

E-mail address: simic@eecs.berkeley.edu

E-mail address: sastry@eecs.berkeley.edu