WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2009; **9**:1028–1039
Published online 10 July 2008 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/wcm.652

Target localization and tracking in noisy binary sensor networks with known spatial topology

Xiangqian Liu[1*,†], Gang Zhao[1] and Xiaoli Ma[2]
[1]*Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY 40292, U.S.A.*
[2]*School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, U.S.A.*

Summary

Target localization and tracking are two of the critical tasks of sensor networks in many applications. Conventional localization and tracking techniques developed for wireless systems that rely on direction-of-arrival (DOA) or time-of-arrival (TOA) information are not suitable for low-power sensors with limited computation and communication capabilities. In this paper, we propose a low-complexity and energy-efficient method for target localization and tracking in noisy binary sensor networks, where the sensors can only perform binary detection, and the physical links are characterized by additive white Gaussian noise (AWGN) channels. The proposed method is based on known spatial topology. An efficient wake-up strategy is used to activate a particular group of sensors for cooperative localization and tracking. We analyze the localization error probability and tracking miss probability in the presence of prediction errors. Simulation results validate the theoretic analysis and demonstrate the effectiveness of the proposed approach. Copyright © 2008 John Wiley & Sons, Ltd.

KEY WORDS: localization; tracking; target detection; binary sensor networks; spatial topology

1. Introduction

Wireless sensor networks are composed of nodes with sensing, wireless communication and simple computation capabilities [1]. Most sensor networks consist of a large number of inexpensive and low-power wireless sensors, and usually achieve their objectives such as localization and tracking in a distributed and cooperative way. Existing localization approaches for wireless systems based on direction-of-arrival (DOA) [2] or time-of-arrival (TOA) [3] are not directly applicable in sensor networks with stringent power/size constraints of sensors.

As a low-power and bandwidth-efficient solution, recently binary sensor networks have been proposed for localization and tracking [4–7]. In a binary sensor network, the spatial topology (locations of all sensors) is usually assumed to be known. This could be realized at the time of network deployment or through a location service [8]. A typical sensor can only make a binary decision regarding a target or object of interest, and consequently, only one bit information needs to be sent from a sensor to a cluster head for further processing. For example, in Reference [4], a binary sensor model is proposed in which a sensor can only detect whether an object is moving toward or away from it. With two

*Correspondence to: Xiangqian Liu, Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY 40292, U.S.A.
†E-mail: x.liu@louisville.edu

convex hulls (one is formed by sensors returning one and the other is formed by sensors returning zero), the moving direction of a target can be tracked. However, to distinguish two targets with the same direction, an additional proximity bit is needed. A quadruplet of binary sensors are used in Reference [5] to determine the range of velocity slope, where the computation is done in a distributed manner. In Reference [6], a sensor can only detect a target within a certain range and returns a bit indicating whether a target is present in this range. It uses a weighted average of the locations of sensors having detected the target as the target's estimated location. This method is effective in tracking. However, each node needs to record the duration for which the target is in its range. Three different weight schemes to obtain the predicted location of the target have been proposed in Reference [7]. The performance is verified by extensive simulations. Each node also needs to record the duration for which the target is in its range.

All of the aforementioned methods assume error-free transmission from sensors to a cluster head or a fusion center. More realistic binary sensor networks with noisy links are considered in References [9,10]. Target tracking is formulated in Reference [9] as a hidden state estimation problem over the finite state space of sensors, where the Viterbi algorithm is adopted to track an object. Particle filtering is used in Reference [10] to fuse information collected from sensors. Although these two methods take noise into consideration, the computation involved is highly complex.

In this paper, we develop a low-complexity target localization and tracking method for noisy binary sensor networks. For simplicity, we assume that a sensor sends a '1' to a cluster head if a target exists within its detection range, and a '0' if a target does not exist within that range. This is similar to the unit-disk model used in sensor network routing [11–13]. The proposed method is based on a known network topology where the cluster head relies on the intersection of the detection areas of sensors to localize and track targets. Compared to the existing approaches, our proposed method takes into account noise in wireless links and yet remains low-complexity and bandwidth efficient with minimum power consumption.

The rest of the paper is organized as follows. In Section 2, we propose a low complexity target localization method by exploiting known spatial topology of sensor networks. Two general criteria are designed in Section 3 to evaluate the accuracy of the proposed localization method with different network topologies. In Section 4, we analyze the error probability of the localization method under the constraint of wake-up errors. Monte Carlo simulation results are presented in Section 5 to validate the theoretic analysis. Section 6 concludes the paper.

2. Target Localization and Tracking

2.1. Target Localization Using Spatial Topology

Consider a sensor field with sensors distributed according to a certain topology. The sensor field can be classified into many sensor clusters. Within each cluster, there are two types of sensors: a cluster head (also called the monitor sensor herein) and detecting sensors. The cluster head is usually more powerful, and one of its tasks is to keep in listening status and to wake up a group of detecting sensors (using beacons) in a certain area to localize and track a target in case the target enters the sensor field. A detecting sensor has a certain detection range. For simplicity, we assume that the range is a circle centered at the sensor with radius a. Because of the minute size and low-power consumption requirement, the detecting sensors are assumed to have limited detection and communication capability. For example, the detecting sensors can only detect whether or not a target exists within its range and then send binary messages to the cluster head. Monitor sensors can execute relative complicated calculation and have a longer lifetime than detecting sensors.

Every active detecting sensor (sensor that has been waken-up) transmits one bit of information to the cluster head per duty cycle. The information could be coded to increase robustness against channel noise. For simplicity of analysis, we assume that the uncoded bit sent by the ith sensor is D_i, and if a target is in the sensor's detection range, $D_i = 1$, and otherwise $D_i = 0$. Suppose that the transmission power is ρ. Then, the received signal at the cluster head from the ith sensor can be written as

$$s_i = \sqrt{\rho} D_i + n_i \tag{1}$$

where n_i is additive white Gaussian noise (AWGN) with zero mean and variance σ^2. Similar to Reference [4], we assume that a proper media-access control (MAC) protocol is employed so that signals from different sensors can be separated at the cluster head, and the sensor locations (spatial topology) are known at the cluster head as well. The problem considered here

DOI: 10.1002/wcm

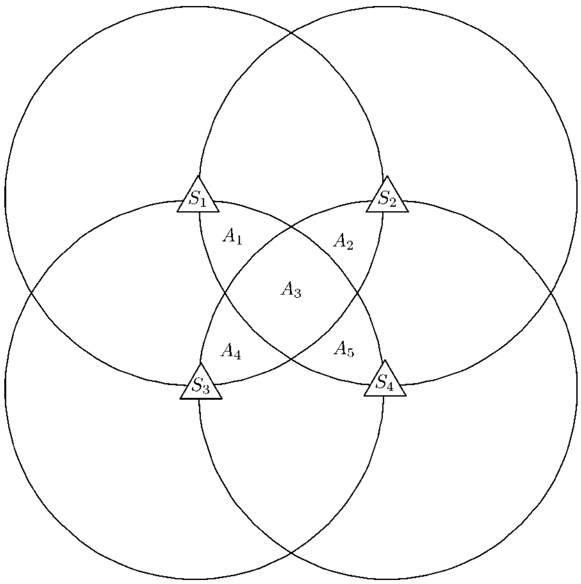

Fig. 1. An illustration of localization in a binary sensor network.

is how to locate the target in the sensor field from s_i, $i = 1, \cdots, M$, where M is the number of active sensors.

We use an example to illustrate our proposed localization method in the following. In Figure 1, suppose that S_1, S_2, S_3, S_4 are four active detecting sensors. This area is partitioned into subareas by the detection ranges of the four sensors. For example, the subarea A_2 is covered by three sensors (S_1, S_2, and S_4). Each subarea is uniquely determined by a detection sequence. For example, if the four active sensors, S_1, S_2, S_3, S_4, return a detection sequence $[D_1, D_2, D_3, D_4] = [1, 1, 1, 0]$, then the cluster head can determine that the target is located at the subarea A_1. We use the center of A_1 to represent the location of the target. Therefore, as long as the topology is known to the cluster heads in advance, calculating the centers of subareas is always possible, although more complicated with irregular topologies.

Apparently, the resolution of this low complexity localization method is to the level of subareas, which may not be as high as that offered by more sophisticated DOA or TOA-based approaches. However, when the density of the sensor field increases, this method can also provide accurate localization that may be sufficient for many applications, at extremely low complexity and power cost. An important component of the proposed scheme is how to wake-up a group of detecting sensors. In the following, we present an energy-efficient wake-up strategy for localization and tracking of targets.

2.2. Tracking With Prediction Errors

When the network is set up, all monitor sensors communicate with their neighbors to divide the whole network into some clusters, each covering a subregion. Here, we apply a simple clustering technique. Suppose that a monitor sensor M_1 has four monitor neighbors: M_i, $i = 2, \cdots, 5$. We use $\perp M_1 M_i$ to denote the bisector of $M_1 M_i$. Then the subregion enclosed by $\perp M_1 M_i$, $i = 2, \cdots, 5$, is a subregion that is monitored by M_1. M_1 activates binary sensors in its subregion if it is necessary and collects information returned by binary sensors in the subregion. Notice that more sophisticated clustering techniques may be applied at increased complexity, for example, References [14–17].

The sensors on the borders of the deployment region will stay active all the time. When a target enters the deployment region, sensor on the borders will detect it and will inform the corresponding cluster head. The cluster head will then wake up the relevant subset of sensors through beacon messages. A number of energy saving wake-up strategies have been proposed, for example, References [18–20]. Most of these protocols use prediction to restrain the sensing region. Suppose a target has been localized at least in two sub-areas in a sequence, then the next position of the target can be predicted according to the approximate direction and velocity of the target's movement, and thus a new set of sensors may be waken up to localize the target. Specifically, suppose that the tracking update interval is Δt, and for a given time $i\Delta t$, let (x_i, y_i) and $(\hat{x}_i, \hat{y}_i)$ denote the two-dimensional coordinates of the true and estimated locations of the target, respectively. Assuming that the target moves in the same direction with a constant speed, a cluster head predicts the next location $(\hat{x}_{i+1}, \hat{y}_{i+1})$ of the moving target from its current estimated location $(\hat{x}_i, \hat{y}_i)$ and the previous estimated location $(\hat{x}_{i-1}, \hat{y}_{i-1})$, according to $\hat{x}_{i+1} = 2\hat{x}_i - \hat{x}_{i-1}$ and $\hat{y}_{i+1} = 2\hat{y}_i - \hat{y}_{i-1}$. Then, the cluster head wakes up four sensor nodes that are closest to $(\hat{x}_{i+1}, \hat{y}_{i+1})$. The detection results of the four sensors are sent back to the cluster head for further localization and tracking decisions, and $(\hat{x}_{i+1}, \hat{y}_{i+1})$ will be updated accordingly. If the predicted location is out of the subregion covered by the current cluster head, it informs another corresponding cluster head to continue to track the target. We define a threshold ξ. If a sensor detects no target in ξ consecutive time slots, it goes to sleep.

The prediction of the target's next location may be wrong due to randomness of the movement of the target. In this case, it is necessary to wake up more than four sensors to locate a target. An illustration of

DOI: 10.1002/wcm

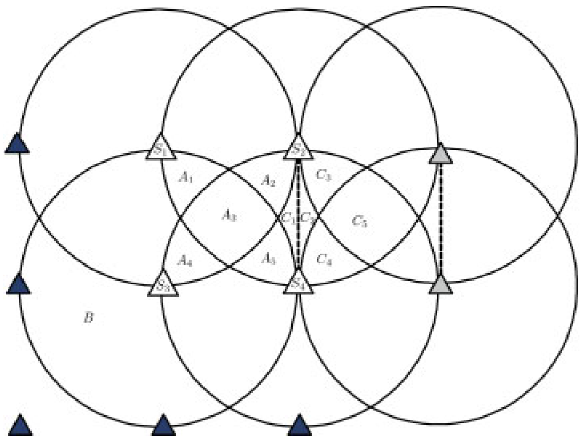

Fig. 2. An illustration of the wake-up strategy in a binary sensor network.

the wake-up strategy is shown in Figure 2. Suppose that the monitor sensor determines that the target is within the square formed by four sensors (indicated by S_1 through S_4). Then, the monitor sensor wakes up these four sensors to further locate the target. We refer this as the first round of detection. However, this prediction may be wrong, for example, if all four sensors return zero to the monitoring sensor. In this case, a second round is necessary, in which the monitoring sensor wakes up more sensors surrounding the initial square, based on the assumption that even if a prediction is wrong, it is most likely not very far away from the true location of the target. The process can continue until the target is located or a pre-set threshold on the number of sensors that can be waken up is met.

Specifically, for the square topology shown in Figure 2, our wake-up strategy works in the following way. In the first round, sensors S_1–S_4 are waken up, and then

(1) If three or more sensors return 1's to the cluster head, the target is considered to be located. No more sensors need to be waken up for this duty cycle.
(2) If only one sensor (e.g., S_3 in Figure 2) returns a '1,' the cluster head wakes up five more sensors (i.e., the five ones indicated by dark-shaded triangles in Figure 2) for a second round detection. Since the area covered by S_3 but not S_1, S_2, or S_4, that is, area B, can only be covered by one or more sensors of the five dark-shaded ones.
(3) If only two sensors on the same edge of the square (e.g., S_2 and S_4) return 1's, the cluster head wakes up two more sensors (i.e., the two lightly shaded triangles to the right of S_2 and S_4) for a second round detection, because the target can be in subareas C_i, $i = 1, \cdots, 5$, as shown in Figure 2.
(4) If only two sensors on the diagonal of the square (e.g., S_1 and S_4) return 1's, a localization error occurs and the four sensors are asked to retransmit signals to the cluster head.
(5) If no sensor returns a '1,' the cluster head wakes up all 12 sensors surrounding the square formed by S_1–S_4 for a second round of detection.

Depending on the result of the second round of detection if activated, the monitor sensor may have to wake up additional sensors for a third round detection, similar (but not exactly the same) to the procedure described above, which is omitted here due to space limitation. This process continues until the target is located or a pre-set number of rounds is reached due to energy limitation.

3. Analysis of Localization Accuracy

In this section, we design two criteria to evaluate the accuracy of the proposed target localization mechanism. We use the criteria to compare two different network topologies. We hope that our criteria can provide useful guidelines for designing efficient network topologies based on specific performance requirements. The topologies used here as examples are the square topology shown in Figure 3 and the hexagon topology shown in Figure 4. Here, we assume that the target is localized in the correct subarea. Localization error due to channel noise is discussed in the next section.

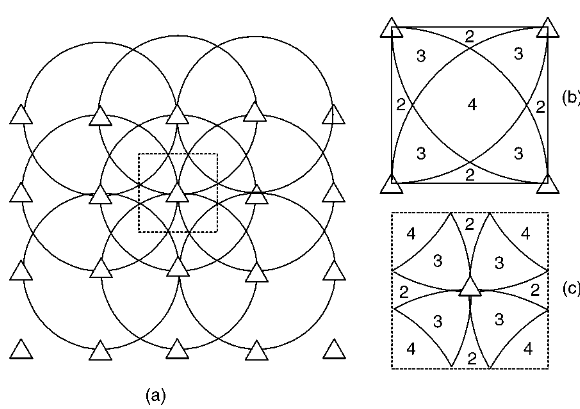

Fig. 3. A sensor network with square topology: (a) the sensor field; (b) zoom-in coverage; and (c) a micro-cell of one sensor. The small triangles denote the sensors, and the circles are the detection ranges of the sensors.

DOI: 10.1002/wcm

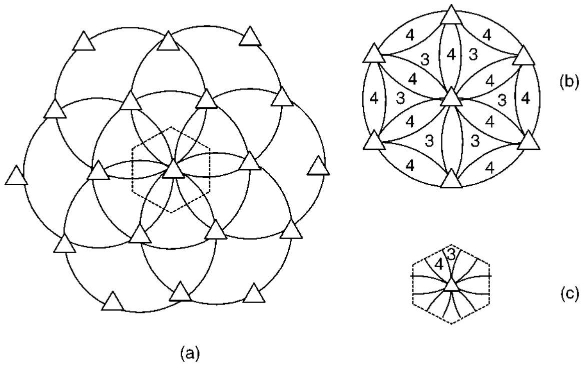

Fig. 4. A sensor network with hexagon topology: (a) the sensor field; (b) zoom-in coverage; and (c) a micro-cell of one sensor.

As shown in Figure 3(a), in a square topology, the sensors are placed on the vertices of a square lattice. This topology has been used, for example, in ANTs [21]. Here, we assume that the detection radius a is equal to the distance of two adjacent sensors in the lattice. A zoom-in coverage in the sensor field is shown in Figure 3(b), where the numbers in each subarea denote the number of sensors that cover this subarea. Suppose the sensor field is large. Without considering the edge effect, a typical micro-cell, or a basic unit area for one sensor is given in Figure 3(c). The corresponding illustration for the hexagon topology is given in Figure 4.

3.1. Worst Case Accuracy

Since our method only locates the target to a sub-area, it is desirable that the sub-areas are as small as possible to increase the localization accuracy. One way to quantify the performance is to find the largest sub-area as the worst case accuracy indicator of target localization for a network topology. Clearly, the smaller the sub-area of the worst case, the better the localization resolution. This criterion shows the resolution of the localization method in an extreme case. It implies that to obtain good localization accuracy, we need to design the network topology in such a way that the sensor field is split into sub-areas as evenly as possible for a given number of sensors.

Suppose the sensor field is large. Without considering the edge effect, a typical micro-cell for one sensor for the square topology is given in Figure 3(c). The area of the micro-cell in Figure 3(c) is a^2. It is easy to verify that a '4' sub-area in Figure 3(b) is $(1+\pi/3-\sqrt{3})a^2$. Thus, the normalized area of a '4' sub-area is $(1+\pi/3-\sqrt{3})$. Similarly, we can find the normalized areas for other sub-areas. We list them in Table I. A typical micro-cell of the hexagon topology is shown in Figure 4(c), from which we can obtain the normalized areas of the sub-areas as in Table II.

Table I. Square topology.

No. of sensors covering a sub-area	Normalized area	No. of sub-areas
2	$2-\frac{\sqrt{3}}{2}-\frac{\pi}{3}$	2
3	$\frac{\sqrt{3}}{2}+\frac{\pi}{12}-1$	4
4	$1+\frac{\pi}{3}-\sqrt{3}$	1

Table II. Hexagon topology.

No. of sensors covering a sub-area	Normalized area	No. of sub-areas
3	$2-\frac{\pi}{\sqrt{3}}$	2
4	$\frac{2\pi}{3\sqrt{3}}-1$	3

Reading from Table I, we obtain that the largest subarea for square topology is the one covered by four sensors, which accounts for 31.51 per cent of a typical micro-cell. However, the worst case for hexagon topology is only 20.92 per cent of a micro-cell. Apparently, the hexagon topology outperforms the square topology with this criterion. Note that it is not difficult to see from Figures 3(b) and 4(b), that the hexagon topology splits the field more evenly than the square topology does.

3.2. Average Accuracy

The worst case criterion quantifies one aspect of the accuracy of target localization of a network topology. Within a fixed area, the more sub-areas in general we have, the more accurate location in average we can obtain. This criterion measures how many sub-areas a network topology has, and it implies that we have to design the network topology in a way such that the field is split into as many sub-areas as possible with a given number of sensors. This criterion does not contradict with the worst case criterion. In fact, they complement each other in designing a sensor network for more accurate target localization.

From Figure 3(c), we observe that one micro-cell contains seven sub-areas for the square topology, while for the hexagon topology, one micro-cell only has five sub-areas as shown in Figure 4(c). Notice that in Figure 3(c), each area indicated with a '2' is only

DOI: 10.1002/wcm

counted as a half of a sub-area, and each area indicated with a '4' is counted as one-fourth of a sub-area, because that is the case when we consider the micro-cell in the context of the sensor field. The number of sub-areas is calculated similarly in Figure 4(c). Thus, the average sub-area is $1/7 = 14.29$ per cent of a unit area for the square topology, and the average sub-area for the hexagon topology is $1/5 = 20$ per cent of a unit area. It is ready to say that the square topology has higher average accuracy than the hexagon topology.

Because the communication channel is noisy, localization error is unavoidable. In the following section, we analyze the localization error probability for the proposed localization mechanism, in the presence of possible prediction errors of the monitor sensor.

4. Analysis of Localization Error Probability

Taking into account the noise effect of the physical channel, the communication between sensors and the cluster head is no longer perfect (as shown in Equation (1)). We first consider the localization error probability when the prediction of the monitoring sensor is error-free, that is, the proper four sensors are waken up. Then, prediction error is taken into account in our analysis.

4.1. Zero Prediction Error

At the cluster head, hard-decision is performed on the signal received from each sensor. In other words, the cluster head decides whether D_i is 1 or 0 from s_i in a maximum likelihood sense. Based on the decisions, the cluster head can locate the target to a single subarea in the field. However, if because of noise, the cluster head makes a wrong decision on D_i, then the target may be detected erroneously in a wrong subarea. The localization error probability depends on noise power σ^2, signal power ρ, and the number of sensors involved.

From Equation (1), it is ready to verify that the average probability of error for the decision on each sensor's signal (i.e., decide 1 as 0, or *vice versa*) is

$$P_e = Q\left(\sqrt{\frac{\rho}{2N_0}}\right) \tag{2}$$

where the Q-function is defined as $Q(x) = (1/\sqrt{2\pi}) \int_x^{\infty} \mathrm{e}^{-t^2/2}\mathrm{d}t$.

For the aforementioned square topology, correct target localization in a subarea requires correct decisions on signals sent by four sensors if there is no prediction error. Therefore, the localization error probability is

$$P_{\text{square}} = 1 - (1 - P_e)^4 \tag{3}$$

Suppose that the number of subareas in a sensor field is N, and the localization error probability for the nth subarea is R_n. If the location of the target is uniformly distributed in a sensor field, then the average localization error probability of the sensor network is

$$P = \sum_{n=1}^{N} a_n R_n \tag{4}$$

where a_n is the normalized area for the nth subarea. According to Equation (4), one has to consider network topologies case by case to obtain the exact localization error probability. However, a good approximation can be obtained for a type of networks where the sensors are uniformly placed in the field, which is stated as follows.

Proposition 1. *Suppose that a sensor field has a total area A, the detection area of each sensor is c, and the number of sensors in the field is S. If the sensors are uniformly placed in the field, then the average error probability in Equation* (4) *can be approximated as*

$$\bar{P} = 1 - (1 - P_e)^k, \text{ with } k := \lceil \frac{cS}{A} \rceil \tag{5}$$

Using Equation (5), we can calculate that for the square topology, the parameter k is $k = \lceil \pi a^2 N/(a^2 N) \rceil = 4$. Thus, according to Proposition 1, the approximate localization error probability with the square topology is $\bar{P}_{\text{square}} = 1 - (1 - P_e)^4$, which matches with P_{square} in Equation (3).

4.2. Non-Zero Prediction Error

We use the square topology as an example again to analyze the localization error probability in the presence of prediction error. We assume that the actual location of the target follows a distribution $f(x, y)$ with respect to the predicted target location (x_0, y_0).

Let $P_s(i)$ be the probability that the target is located in the square region surrounded by the sensors waken up till the ith round, which can be calculated as $P_s(i) = \int_{S(i)} f(x, y)\mathrm{d}x\mathrm{d}y$, where $S(i)$ denotes the area surrounded by the sensors waken up till the ith round. Let $P_i(j)$ denote the probability that a target is covered

DOI: 10.1002/wcm

by j sensors waken up till the ith round. We can calculate $P_i(j)$ as $P_i(j) = \int_{C(i,j)} f(x, y)\mathrm{d}x\mathrm{d}y$, where $C(i, j)$ is the area covered by j sensors till the ith round. For example, as shown in Figure 1, $C(1, 3)$ is the area of the four subareas indicated with A_1, A_2, A_3, and A_4, each of which is covered by three sensors.

If only one round of detection can be used, the localization error probability is given by $P_1 = 1 - (1 - P_e)^4 P_s(1)$. If up to two rounds of wake-up are possible, the localization error probability can be calculated as follows. We first activate four sensors to detect the target. If they return three or more 1's and the monitor sensor has detected them correctly, the target is properly localized. This situation can occur only when the target is located in the subareas of the first square covered by three or more sensors. Therefore, the probability that the target can be localized in the first round is $(1 - P_e)^4(P_1(3) + P_1(4))$. If only one sensor returns 1, according to our wake-up strategy, five additional sensors are activated. Now the decision of the monitor sensor is based on information of $4 + 5 = 9$ sensors, so the probability of correct localization in this case is given by $(1 - P_e)^9 P_1(1)$. If two sensors on the same edge of the square return 1, two more sensors are activated. The probability of correct localization is $(1 - P_e)^6 P_1(2)$. If no sensor returns 1 in the first round, 12 more sensors are activated. In this case, the target may be located in subareas such as C and H or outside of the area covered by the second round as shown in Figure 1. The probability of correct localization in this case is $(1 - P_e)^{16}(P_s(2) - P_1(1) - P_1(2) - P_1(3) - P_1(4))$. Therefore, if up to two rounds of detection are allowed, the localization error probability is

$$
\begin{aligned}
P_2 = 1 &- (1 - P_e)^4(P_1(3) + P_1(4)) - (1 - P_e)^6 P_1(2) \\
&- (1 - P_e)^{16}(P_s(2) - P_1(1) - P_1(2) - P_1(3) \\
&- P_1(4)) - (1 - P_e)^9 P_1(1)
\end{aligned}
\tag{6}
$$

It can be shown that if up to n rounds of wake-up are allowed, the error probability is given by (for $n \geq 3$)

$$
\begin{aligned}
P_n = 1 &- (1 - P_e)^4(P_1(3) + P_1(4)) - (1 - P_e)^6 P_1(2) \\
&- (1 - P_e)^9 P_1(1) - (1 - P_e)^{16} \\
&\times \Big[P_s(2) - P_1(1) - P_1(2) - P_1(3) - P_1(4)\Big] \\
&- \sum_{i=3}^{n} \Big\{(1 - P_e)^{(4(i-1)^2+2)} P_{i-1}(2) \\
&+ (1 - P_e)^{(4(i-1)^2+3)} P_{i-1}(1) \\
&+ (1 - P_e)^{4i^2}\Big(P_s(i) - \sum_{h=1}^{4} P_{i-1}(h)\Big)\Big\}
\end{aligned}
\tag{7}
$$

In Reference [7], a different sensor model has been proposed. A sensor with a nominal sensing range R can always detect a target's presence if it is within $R - R_e$ range from the sensor. The detection probability drops off continuously as the distance increases between $R - R_e$ and R. Actually, this model can be applied to our localization by combining the detection probability and average probability of error P_e into one error probability P_e'. Replacing P_e with P_e' in the above analysis, we obtain similar results.

Remark 1. *We note that it is possible to relax the assumption that the received signal power is the same for each sensor. Essentially, to take into account different realization of received signal power, we can model the wireless channel as a flat-fading channel that* ρ *in Equation* (1) *is a Rayleigh random variable. Subsequently,* P_e *in Equation* (2) *should be replaced by binary detection error in Rayleigh channels instead of AWGN channel, which is*

$$
P_e = \frac{1}{E[\rho]/N_0}\mathrm{e}^{-\rho N_0/E[\rho]}
$$

All other steps in the performance analysis remain the same.

5. Simulation Results

5.1. Target Localization Under Uniform Sensor Placements

We present the Monte Carlo simulation results to validate the performance analysis on localization error probability of the proposed scheme. Suppose that each sensor's detection range is of unit length. A square sensor field of side length 200 units is covered by sensors deployed with a square topology. The received signal-to-noise ratio (SNR) at the cluster head is defined as ρ/σ^2. There is one target to be localized in each realization. For each SNR value, we generate 5×10^6 realizations of random target locations in the sensor filed. We assume that the actual location of the target follows the two-dimensional standard normal distribution with respect to the center (x_0, y_0) of the four waked-up sensor nodes.

DOI: 10.1002/wcm

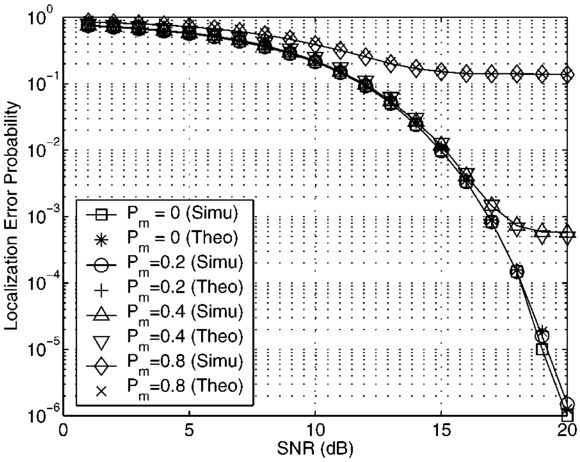

Fig. 5. Localization error probability with up to two rounds of sensor wake-up allowed. 'Theo' stands for theoretic analysis results, 'Simu' stands for simulation results, and P_m is the prediction error of the monitoring sensor.

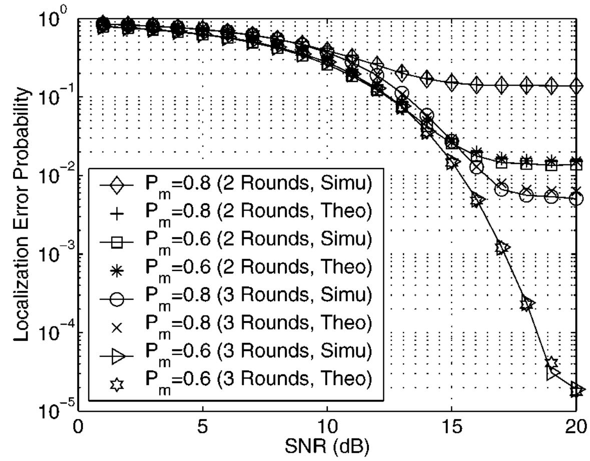

Fig. 6. Comparison of localization error probability when a maximum of two and three rounds of sensor wake-up are allowed. 'Theo' stands for theoretic analysis results, and 'Simu' stands for simulation results.

In Figure 5, we plot the localization error probability of the proposed mechanism under different prediction errors, where P_m denotes the monitoring sensor's prediction error probability. When $P_m = 0$, four sensors are waken up and only one round of detection is necessary. When $P_m > 0$, up to two rounds of detection are allowed. It can be seen from Figure 5 that the simulation results match well with the theoretic analysis results in all cases. An error floor is observed for the cases when the prediction error is large, which is due to the fact that a certain percentage of the targets are out of the coverage area of the sensor waken up in the first two rounds. We also notice that when the prediction error is small (e.g., $P_m \leq 0.2$), up to two rounds of wake-up are sufficient to obtain a localization error probability close to that of the case with zero prediction error.

When the prediction error is large, we may increase the allowable wake-up rounds to reduce the localization error probability. This is the subject of Figure 6, where we compare the localization error probability when a maximum of two and three rounds of detection are allowed. It is clear that an additional round of wake-up sensors can significantly reduce the localization error probability (e.g., by more than one order of magnitude with 80 per cent prediction error, and more than two orders of magnitude with 60 per cent prediction error).

So far, we have assumed that each sensor is able to determine correctly whether the target is within its detection radius. In practical scenarios, however, there may be detection errors, in addition to communication errors caused by the noisy channel from sensors to the cluster head. We have tested the impact of detection errors on localization by simulations. The results are shown in Figure 7. The simulation setup is similar to the previous case with $P_m = 0$ except that the detection error P_d is non-zero. From Figure 7, as expected, it can be seen that the probability of localization error increases significantly as P_d increases.

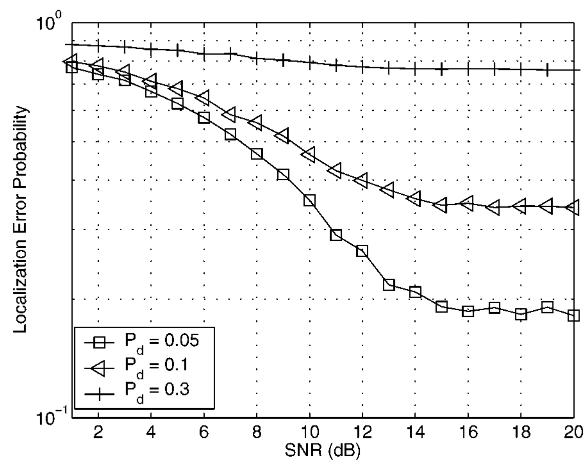

Fig. 7. Localization error probability in the presence of detection error. P_d is the detection error.

Additionally, we have evaluated to use repetition code to improve the localization performance in noisy channel. In the test, instead of 1 bit, 3 bits are used to code the detection result. If the detection result is 1, a sensor node transmits 111 to the cluster head, otherwise it transmits 000. In this case, decoding error of 1 bit out of 3 is tolerable. The simulation

DOI: 10.1002/wcm

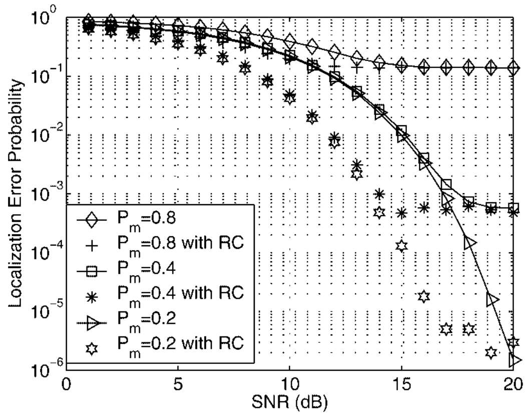

Fig. 8. Comparison of localization error probability with and without using repetition code. 'RC' denotes repetition code.

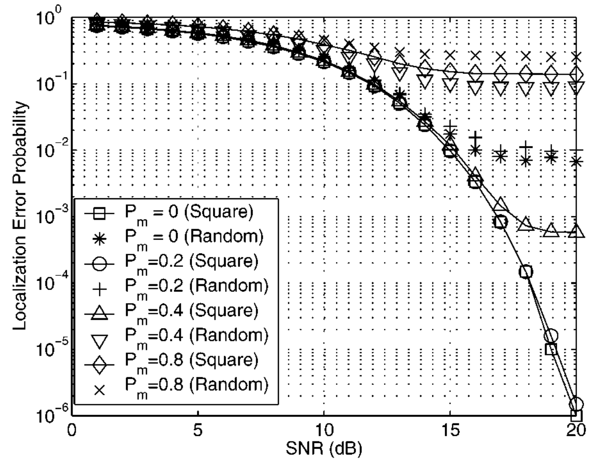

Fig. 9. Comparison of localization error probability with square topology and random topology.

result is shown in Figure 8, which shows that the use of repetition code does improve localization performance, especially when the localization error is dominated by the noisy channel (i.e., at low SNR and when the prediction error is low).

5.2. Target Localization Under Random Sensor Placements

The experiments in Subsection 5.1 assume a square topology. We now evaluate the localization performance of the proposed approach under random sensor topology. Suppose that each sensor's detection range is a random variable uniformly distributed in $[1-\tau, 1+\tau]$, where $\tau = 0.2$. The deployment region is a square sensor field of side length 200 units. We randomly place 40 000 nodes in the sensor field in each realization according to uniform distribution. There is one target to be localized in each realization. For each SNR value, we generate 5×10^6 realizations of random target locations in the sensor filed. As before, we assume that the actual location of the target follows the two-dimensional standard normal distribution with respect to the center of the predicted region. Furthermore, the locations of all sensors in a cluster are assumed known to the cluster head. The cluster head activates up to four sensors closest to the center of the predicted target location. In this case, the cluster head is still able to localize a target to a unique subarea depending on how many sensors can cover that subarea. If only one sensor (sensor A) returns 1, four more sensors closest to A will be activated. If no sensor returns 1, eight more sensors are to be activated to detect the target in a second round.

Figure 9 compares the localization error probability under square topology and random topology. We observe that only at low to moderate SNR, square topology and random topology have similar error probability. As SNR increases, the gap becomes larger. The results show that our proposed approach works best with regular network topology, but also gives comparable localization performance with random topology at low SNR.

5.3. Target Tracking

In this section, we evaluate the performance of the proposed algorithm to track a moving target in a sensor network with square topology. For the purpose of comparison, the simulation setup is similar to that of Reference [19], where the sensors are deployed in a $600 \times 600\,\text{m}^2$ region, and each sensor's sensing range is set at 25 m.

The tracking update interval is Δt. For a given time $i\Delta t$, the predicted next location is given by $(2\hat{x}_i - \hat{x}_{i-1}, 2\hat{y}_i - \hat{y}_{i-1})$, while the true location is randomly generated according to a two-dimensional Gaussian distribution centered at $(2x_i - x_{i-1}, 2y_i - y_{i-1})$ with standard deviation σ_p. A cluster head wakes up four sensor nodes that are closest to the predicted location, and one round of recovery may be invoked using the recovery mechanism described in Section 2 if the four sensors fail to locate the target. We tested both our algorithm and the DPT algorithm in Reference [19] for $i = 1, \cdots, 1000$ after proper initialization. If a target is lost, we record one tracking miss, and re-initialize the tracking. The miss probability is defined as the percentage of tracking misses out of the 1000 tracking points.

DOI: 10.1002/wcm

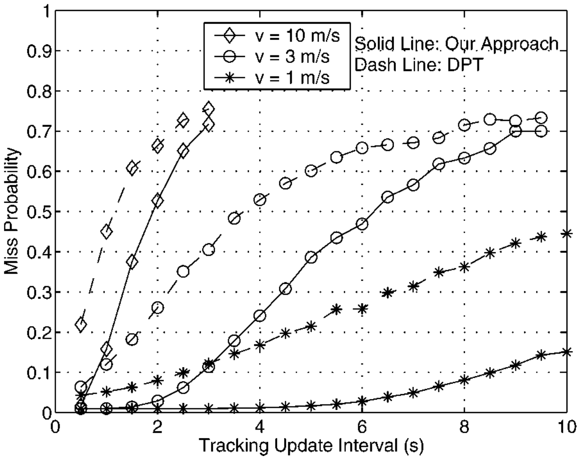

Fig. 10. Comparison of miss probability of our approach and DPT under different tracking update intervals.

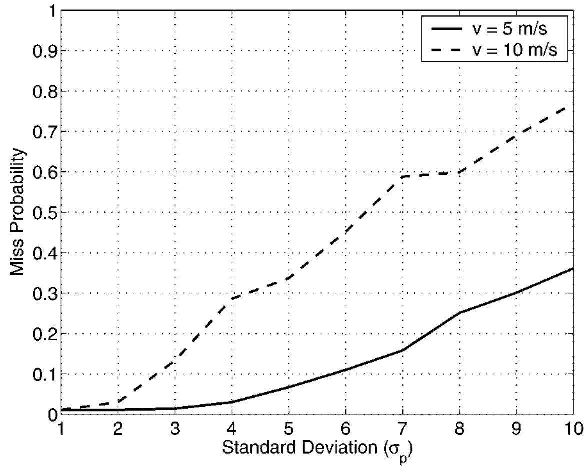

Fig. 12. Miss probability of our approach for different standard deviations of the target's true position.

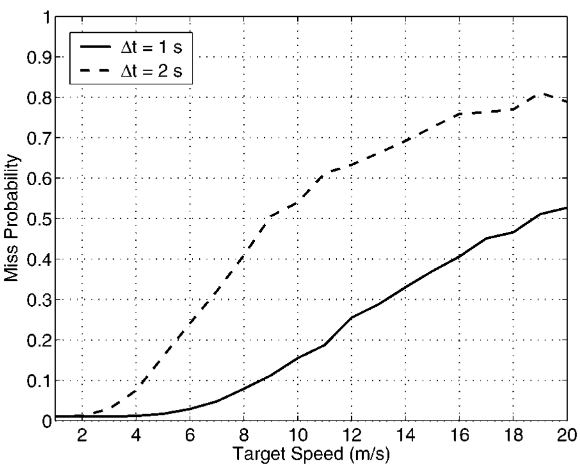

Fig. 11. Miss probability of our approach under different target speeds.

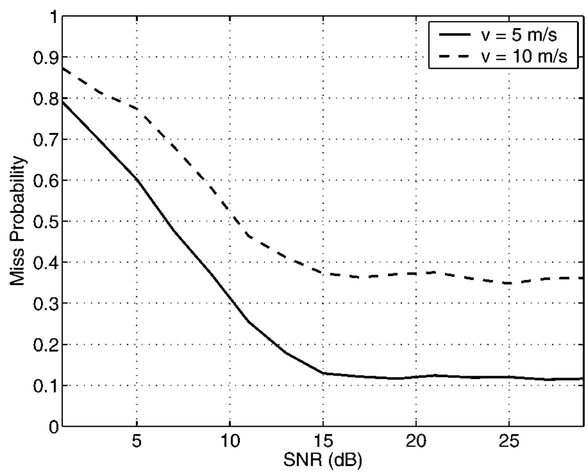

Fig. 13. Miss probability of our approach *versus* SNR.

In the first test, we compare the miss probability of our algorithm and DPT. In this test, we assume that SNR is sufficiently high such that the localization error due to channel noise can be ignored. DPT uses three sensor nodes to locate a target, and it goes into the first round of recovery when none of the three nodes can detect the target. For simplicity, in our test, at most one round of recovery is used. We set the stand deviation $\sigma_p = 3.3v\Delta t$, where v is the target speed. Figure 10 plots the miss probability of the two algorithms *versus* the tracking update interval Δt. The target speed is set at $v = 1, 3,$ and 10 m/s. It can be observed from Figure 10 that the miss probability of both algorithms increase as the target speed and/or the tracking update interval increase. As the speed increases, the miss probability increases significantly because σ_p is proportional to the speed. This is further verified by Figure 11, which shows the miss probability of our proposed algorithm *versus* the target speed. In general, our proposed algorithm offers a lower miss probability than DPT.

Figure 12 depicts the change of miss probability of our algorithm as a function of σ_p, where the unit of σ_p is $v\Delta t$ and $\Delta t = 1s$. As expected, the large σ_p is (or equivalently, in average the farther the true location of the target deviates from the predicted location), the higher the miss probability is.

In the second test, the channel noise is taken into account in tracking of a moving target. Figure 13 shows the miss probability of our approach *versus*

DOI: 10.1002/wcm

SNR, where we set $\sigma_p = 3.3v\Delta t$ and $\Delta t = 1s$. We observe that when SNR is sufficiently high, the miss probability is almost constant. This shows that when SNR is low, the miss probability is dominated by detection errors due to channel noise, and when SNR is high, the miss probability is dominated by prediction error.

6. Conclusions

We have proposed a low-complexity method for target localization and tracking by exploiting spatial topology of a binary sensor network in noisy environments. An efficient wake-up strategy is used to provide predicted location information and activate a particular group of sensors for cooperative localization. Two criteria are designed to quantify the localization accuracy of the proposed method in different network topologies, where square and hexagon topologies are used as examples of evaluation. We have analytically quantified the localization error probability with possible prediction errors, which is further validated by Monte Carlo simulations. It is shown that in the presence of prediction errors, additional rounds of sensor wake-up can significantly reduce the localization error probability. The tracking performance of the proposed approach is evaluated by simulations where the impact of a number of parameters is studied.

Acknowledgments

This research paper was supported in part by the U.S. Army Research Laboratory and the U.S. Army Research Office under grant number W911NF-06 -1-0415.

References

1. Chong C-Y, Kumar SP. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE* 2003; **91**(8): 1247–1256.
2. Swindlehurst A, Kailath T. Azimuth/elevation direction finding using regular array geometries. *IEEE Transactions on Aerospace and Electronic Systems* 1993; **29**(1): 145–156.
3. Li X, Pahlavan K. Super-resolution TOA estimation with diversity for indoor geolocation. *IEEE Transactions on Wireless Communications* 2004; **3**(1): 224–234.
4. Aslam J, Butler Z, Constantin F, Crespi V, Cybenko G, Rus D. Tracking a moving object with a binary sensor network. In *Proceedings of the ACM Conference on Embedded Networked Sensor Systems*, Los Angeles, CA, November 2003; pp. 150–161.
5. Karras P, Mamoulis N. Detecting the direction of motion in a binary sensor network. In *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, Taichung, Taiwan, June 2006; pp. 420–427.
6. Mechitov K, Sundresh S, Kwon Y, Agha G. Cooperative tracking with binary-detection sensor networks. *Technical Report UIUCDCS-R-2003-2379*, Department of Computer Science, University of Illinois at Urbana-Champaign, 2003.
7. Kim W, Mechitov K, Choi J-Y, Ham S. On target tracking with binary proximity sensors. In *Proceedings of the IEEE Information Processing in Sensor Networks (IPSN)*, Los Angeles, CA, April 2005; pp. 301–308.
8. Priyantha N, Chakraborty A, Balakrishnan H. The cricket location-support system. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom)*, Boston, MA, August 2000.
9. Oh S, Sastry S. Tracking on a graph. In *Proceedings of the ACM/IEEE Information Processing in Sensor Networks (IPSN)*, Los Angeles, CA, April 2005; pp. 195–202.
10. Djuric MVPM, Bugallo MF. Signal processing by particle filtering for binary sensor networks. In *Proceedings of the IEEE Digital Signal Processing Workshop*, Taos Ski Valley, NM, August 2004; pp. 263–267.
11. Karp B, Kung HT. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Boston, MA, August 2000; pp. 243–254.
12. Kuhn F, Wattenhofer R, Zhang Y, Zollinger A. Geometric ad-hoc routing: of theory and practice. In *Proceedings of the ACM Symposium on the Principles of Distributed Computing (PODC)*, Boston, MA, July 2003; pp. 243–254.
13. Kuhn F, Wattenhofer R, Zollinger A. Worst-case optimal and average-case efficient geometric ad-hoc routing. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, Annapolis, MD, June 2003; pp. 267–278.
14. Banerjee S, Khuller S. A clustering scheme for hierarchical control in multi-hop wireless networks. In *Proceedings of the ACM/IEEE INFOCOM*, Vol. 2, Anchorage, AK, April 2001; pp. 1028–1037.
15. Lin CR, Gerla M. Adaptive clustering for mobile wireless networks. *Journal of Selected Areas in Communications* 1997; **15**(7): 1265–1275.
16. Cerpa A, Elson J, Estrin D, Girod L, Hamilton M, Zhao J. Habitat monitoring: application driver for wireless communications technology. In *Proceedings of the ACM SIGCOMM Workshop on Data Communications*, San Jose, Costa Rica, April 2001.
17. Farina A, Golino G, Capponi A, Pilotto C. Surveillance by means of a random sensor network: a heterogeneous sensor approach. In *Proceedings of the 8th International Conference on Information Fusion*, Vol. 2, Philadelphia, PA, July 2005.
18. Pattem S, Poduri S, Krishnamachari B. Energy-quality tradeoffs for target tracking in wireless sensor networks. In *Proceedings of International Workshop on Information Processing in Sensor Networks (IPSN)*, Palo Alto, CA, April 2003; pp. 32–46.
19. Yang H, Sikdar B. A protocol for tracking mobile targets using sensor networks. In *Proceedings of the IEEE Sensor Network Protocols and Applications*, May 2003; pp. 71–81.
20. Brooks RR, Ramanathan P, Sayeed AM. Distributed target classification and tracking in sensor networks. *Proceedings of the IEEE* 2003; **91**(8): 1163–1171.
21. Kirkpatrick S, Schneider JJ. How smart does an agent need to be. *International Journal of Physics* 2005; **16**(1): 139–155.

DOI: 10.1002/wcm

Authors' Biographies

Xiangqian Liu received his B.S. degree in Electrical Engineering from Beijing Institute of Technology, Beijing, in 1997, M.S. degree in Electrical Engineering from the University of Virginia, Charlottesville, in 1999, and Ph.D. in Electrical Engineering from the University of Minnesota, Minneapolis, in 2002. Since 2002, he has been with the Department of Electrical and Computer Engineering at the University of Louisville, first as an Assistant Professor, then as an Associate Professor. His research interests are in the areas of wireless communications and signal processing, including multiuser detection, channel modeling and estimation, multi-antenna systems, spectral estimation, sensor array processing, as well as signal processing techniques for wireless sensor networks.

Gang Zhao received his M.S. and B.S. degrees in Computer Science from Shanghai Jiao Tong University, China, in 2002 and 2005, respectively. He is currently pursuing his Ph.D. in the Electrical and Computer Engineering Department at the University of Louisville. His current research focuses on multi-hop routing and target tracking in wireless sensor networks and ad hoc networks.

Xiaoli Ma received her B.S. degree in Automatic Control from Tsinghua University, Beijing, China in 1998, M.S. degree in Electrical Engineering from the University of Virginia in 2000, and Ph.D. in Electrical Engineering from the University of Minnesota in 2003. From 2003 to 2005, she was an Assistant Professor of Electrical and Computer Engineering at Auburn University. Since 2006, she has been with the School of Electrical and Computer Engineering at Georgia Institute of Technology. Her research interests include transceiver designs and diversity techniques for wireless time- and frequency-selective channels, channel modeling, estimation and equalization, carrier frequency synchronization for OFDM systems, routing and cooperative designs for wireless networks.

DOI: 10.1002/wcm