Ad Hoc & Sensor Wireless Networks Vol. 7, pp. 295–323
Reprints available directly from the publisher
Photocopying permitted by license only

An Integrated Protocol for Maintaining Connectivity and Coverage under Probabilistic Models for Wireless Sensor Networks

MOHAMED HEFEEDA* AND HOSSEIN AHMADI

School of Computing Science, Simon Fraser University, Canada
E-mail: {mhefeeda, hahmadi}@cs.sfu.ca

Received: February 18, 2008. Revised: July 20, 2008. Accepted: September 13, 2008.

We propose a distributed connectivity maintenance protocol that explicitly accounts for the probabilistic nature of wireless communication links. The proposed protocol is simple to implement and it achieves a given target communication quality between nodes, which is quantified by the minimum packet delivery rate between any pair of nodes in the network. We demonstrate the robustness of the proposed protocol against random node failures, inaccuracy of node locations, and imperfect time synchronization of nodes using extensive simulations. We extend our connectivity protocol to provide probabilistic coverage as well, where the sensing ranges of sensors follow probabilistic models. Because it employs both probabilistic communication and sensing models, the proposed protocol is more suitable for real sensor network environments than other protocols in the literature that assume a simple disk model for communication and sensing. We compare our protocol against others in the literature and show that it activates fewer number of nodes, consumes much less energy, and significantly prolongs the network lifetime.

Keywords: Connectivity maintenance protocols, coverage protocols, probabilistic sensing models, probabilistic communication models.

1 INTRODUCTION

In the recent few years, there has been significant research interest in designing and using wireless sensor networks for various applications such as intrusion

*M. Hefeeda is also with the Computer Engineering and Systems Department, Mansoura University, Mansoura, Egypt.

detection, area surveillance, and forest fire detection. Two of the fundamental problems in wireless sensor networks are network connectivity and area coverage. A network is connected if every pair of nodes can communicate with each other. Area coverage implies that an event happening at any point in the monitored area can be detected by at leas one node. The connectivity and coverage problems can be addressed separately or jointly. For example, several protocols have been proposed in the literature to maintain the network connected [9–11] and to ensure that the area is covered [7,31,35], while other protocols, e.g., [15], have been designed to achieve both connectivity and coverage at the same time. In this paper, we first develop a new connectivity maintenance protocol, then extend it to ensure coverage as well.

To study network connectivity, many previous works represent the network as an undirected unweighted graph, where network connectivity is equivalent to graph connectivity. In the graph representation, there is an edge between two nodes if they are within the communication range of each other. The communication range of a node is typically assumed to be a disk with radius r_c, where r_c is referred to as the communication range of a node. We call this the disk communication model, which results in a *deterministic* connectivity model for the network. The deterministic connectivity model started in wired networks, and then used widely in wireless ad hoc and sensor networks. While it is fairly accurate in wired networks, several papers [1,16] argue that the deterministic connectivity model is not appropriate for wireless networks. This is because it has been experimentally shown that communication ranges of nodes are not nice regular disks. Rather, they are better represented by probabilistic models. Therefore, two wireless nodes cannot said to be 'connected' or 'disconnected' in the perfect sense. Instead, a link between a pair of wireless nodes should have a probability of data delivery between these two nodes. In addition, it is neither sufficient nor precise to state that the network is simply connected. Rather, a quantitative measure of the quality of communication between arbitrary nodes in the network is needed.

Several connectivity maintenance protocols in the literature, e.g., [11,32, 33], use the deterministic connectivity model, because it facilitates the design and performance analysis of the protocols. By relying on the deterministic connectivity model, these protocols may not function properly in real environments. In addition, these protocols fail to provide any assessment of the quality of communication between nodes in a wireless sensor network. Similar to communication models, sensing models have been shown to be probabilistic in nature [3,6]. In this paper, we take a first step in designing an integrated connectivity and coverage protocol for more realistic communication and sensing models. We realize that communication and sensing models will depend, among other factors, on the sensor technology, physical phenomena being sensed, and environment in which sensors are deployed. This indicates that no single model will be sufficient for all applications. Therefore, we design

our protocol to be flexible in order to accommodate different communication and sensing models and to be useful for wide range of applications.

1.1 Paper contributions

Our contributions can be summarized as follows.

- We propose a quantitative measure for the quality of communication between nodes in sensor networks by defining the probability of packet delivery between arbitrary nodes in the network. We analytically derive this probability for common node organization schemes such as grid, triangular lattice, and uniform node organizations. The probability of packet delivery represents a basic communication quality metric from which higher level metrics can be inferred. For example, higher packet delivery rates imply higher throughput and shorter delay in the sensor network, because more packets make it through the network to their destinations without the need for retransmissions.
- We propose a distributed connectivity maintenance protocol to achieve a given target communication quality between nodes. The proposed protocol is simple to implement, and we demonstrate its robustness against random node failures, inaccuracy of node locations, and imperfect time synchronization of nodes using extensive simulations. We show that the proposed protocol minimizes the number of activated nodes and consumes much less energy than other protocols in the literature. In addition, the operation of the proposed protocol does not depend on the specifics of the adopted communication model, which enables it to be used with different models and in various environments, including the commonly-used disk model. Furthermore, because our protocol ensures a given communication quality, it can be used for critical applications such as backbone construction [25] in large-scale sensor networks, where a subset of nodes are chosen to deliver data from the whole network.
- We extend the proposed connectivity protocol to provide probabilistic coverage as well, where the sensing ranges of sensors follow probabilistic models. To the best of our knowledge, our proposed protocol is the first protocol to employ both probabilistic communication and probabilistic sensing models at the same time. Therefore, our protocol is more suitable for real sensor network environments than most others in the literature.
- We describe how the proposed protocol can provide a controllable quality of coverage and connectivity. This can be used by sensor network designers and operators to optimize the operation of their networks by trading off unnecessary reliability or coverage assurance with savings

in number of sensors. This may yield saving in energy consumption and ultimately extending the network lifetime.

1.2 Paper organization

The rest of this paper is organized as follows. In Section 2, we summarize the related works. In Section 3, we define the probabilistic connectivity notion and derive the probability of packet delivery in common node organization schemes. In Section 4, we present our connectivity maintenance protocol, and then extend it to provide coverage as well. In Section 5, we rigorously evaluate and compare our protocol against other protocols in the literature. Section 6 concludes the paper.

2 RELATED WORK

We summarize the related works in the following subsections.

2.1 Connectivity under the disk communication model

Because of its importance, the connectivity problem in sensor networks has received significant research attention. Several connectivity maintenance protocols have been proposed in the literature. We divide these protocols into two classes. In the first class, the protocol exchanges some messages to discover the connected components in the network [9, 10, 34]. For example, SPAN [10] maintains a list of neighbor nodes based on the received hello messages. Then, each node checks whether there exists a pair of neighbors that cannot reach each other directly or via one or two hops. If this is case, the node becomes active; otherwise, it turns itself off to save energy. PEAS [34] and ASCENT [9] send probing messages. A node in PEAS uses probing messages to discover whether there are other working nodes in the probing range, and it goes to sleep if it finds any. PEAS uses the number of working nodes in the probing range to set the sleep duration. ASCENT [9] uses the probing messages to estimate the reachability between neighboring nodes by measuring the packet loss rates, and uses this information to decide on which nodes should stay on. This class of protocols suffers from high communication overhead, which consumes a nontrivial fraction of nodes' energy.

The second class of connectivity maintenance protocols uses information about the communication range of sensors to maintain connectivity [11,32,33]. For example, the Geometric Adaptive Fidelity (GAF) protocol [33] divides the area into square cells such that all nodes inside a cell can communicate with all nodes in neighboring cells. GAF, then, keeps only one node active in each cell. On the other hand, the algorithm in [11] builds a connected dominating set in a distributed manner. These connectivity maintenance protocols rely on the assumption that the communication range is a regular disk, which is an over-simplification of wireless nodes in real environments [1, 16]. Our proposed protocol assumes that the communication ranges follow a probabilistic model,

which is more realistic. In addition, our protocol is more general and can support the disk communication model as well. To demonstrate this generality, we compare our protocol versus two highly-cited deterministic connectivity protocols in the literature: one from the first class, SPAN [10], and another from the second class, GAF [33]. The simulation results show that our protocol outperforms both of them.

2.2 Connectivity under probabilistic communication models

Several measurement studies [1, 16, 23, 36] have been conducted and a few analytical models [39] have been proposed to understand the communication behavior of wireless links in realistic environments. Other works [27,36] study the impact of realistic communication models on various network protocols such as MAC and routing protocols. Our work is orthogonal to these studies, in the sense that our proposed protocol can employ any communication model and we concentrate on connectivity maintenance protocols. For example, the recent work in [39] proposes a detailed communication model that accounts for channel conditions, e.g., shadowing and path loss exponent, and radio receiver characteristics, such as encoding, modulation, and hardware differences. This communication model yields the packet reception rates between wireless nodes. As explained later, this packet reception rate is the main input to our connectivity maintenance protocol.

Recently, there have been some efforts to develop realistic models for connectivity in wireless sensor networks. One approach employs a geometric random graph representation of the network to reflect the probabilistic behavior of wireless communications [5, 14]. In this case, there is an edge between each pair of nodes with a probability related to the distance between them. The work in [14] assumes that this probability is given by the log-normal shadowing model [22]. Using this model, the authors prove two theorems. The first theorem states that the graph is connected with high probability if each pair of nodes have an edge with probability at least $n/\log n$. The second theorem shows that the probability that the graph is k-connected is equal to the probability at which the minimum node degree is at least k. Both theorems were previously proven for geometric random graphs using the disk communication model [21]. The work in [5] derives the probability that a node in the network is isolated based on the node deployment density. The authors also show that this node isolation probability is an upper bound on the probability of having the network connected. The authors of [19] improve on [5] by giving closed-form equations for the node isolation probability. Unlike our work, [5, 14, 19] do not propose distributed protocols to maintain connectivity under probabilistic communication models.

2.3 Integrated coverage and connectivity

A closely related problem to connectivity is coverage, where a subset of deployed nodes are activated such that any event in the monitored area is detected by at least one sensor. Similar to the communication range, the

sensing range of sensors can either be modeled as a simple regular disk or assumed to follow a probabilistic model. Several distributed coverage protocols have been proposed for the disk model, e.g., [7, 31, 35]. For example, OGDC [35] tries to minimize the overlap between the sensing circles of activated sensors, while CCP [31] deactivates redundant sensors by checking that all intersection points of sensing circles are covered. Other earlier protocols include PEAS [34] and Ottawa [29].

Several studies [3,6,17,37,38] have argued that probabilistic sensing models capture the behavior of sensors more realistically than the deterministic disk model. For example, through experimental study of passive infrared (PIR) sensors, the authors of [6] show that the sensing range is better modeled by a continuous probability distribution, which is a normal distribution in the case of PIR sensors. The work in [17] analytically studies the implications of adopting probabilistic and disk sensing models on coverage, but no specific coverage protocol is presented. In [3], the sensing range is modeled as layers of concentric disks with increasing diameters, where the probability of sensing is fixed in each layer. A coverage evaluation protocol is also proposed. Although the authors mention that their coverage evaluation protocol can be extended to a dynamic coverage protocol, they do not specify the details of that protocol. Therefore, we could not compare our protocol against theirs. In [38], the authors assume that the sensing capacity decreases exponentially fast after certain threshold. The authors also design a probabilistic coverage protocol based on that model.

In most sensor networks, both the coverage and connectivity problems need to be solved simultaneously such that deployed sensors collect data to represent the whole monitored area, and this data is then delivered to a processing center for possible actions. Therefore, integrated coverage and connectivity protocols have been proposed in the literature, e.g., [31] and [15]. The protocol in [15] assumes disk models for both communication and sensing ranges, and works as follows. A distributed sleep control protocol is invoked to evaluate the perimeter coverage of sensors and to put some sensors in sleep mode while maintaining the required coverage level. Then, a power control protocol is used to set the transmission ranges of sensors to save energy and achieve a given connectivity degree. This work assumes fine control of the transmission ranges, which may not be possible in realistic sensors. In addition, for low-power radio communications, which is the case in sensor networks, the energy consumed in determining the appropriate transmission range may exceed the energy saved from reducing the transmission range. This is because the maximum transmission ranges for sensors are typically small (up to few hundred meters) and the power consumed in the receiving and sending electronic circuits is significant compared to the power consumed in the transmission antenna for small distances; see for example [30] for elaborate analysis of energy consumption in sensors. The authors of [31] integrate the CCP coverage protocol with the SPAN connectivity maintenance protocol [10]

to provide an integrated CCP-SPAN protocol that achieves both coverage and connectivity. CCP-SPAN assumes a disk model for the sensing range.

None of the above protocols accounts for both probabilistic communication and sensing models at the same time. Therefore, to the best our knowledge, our proposed protocol is the first to provide both coverage and connectivity under more realistic sensing and communication models. Furthermore, we show that our integrated protocol can easily employ the disk model for sensing and communication ranges. And, through simulation, we show that our protocol outperforms the widely-cited integrated coverage and connectivity protocol in the literature, CCP-SPAN [31].

Finally, we should mention that, under the disk assumption for sensing and communication ranges, the protocols in [31] and [15] can provide k-coverage ($k \geq 1$), where every point is sensed by at least k sensors, and c-connectivity ($c \geq 1$), where there are at least c communication paths between any pair of sensors. Whereas, our proposed protocol provides β-coverage and α-connectivity, where $\alpha, \beta \leq 1$, but it employs realistic sensing and communication models. Redundant coverage and connectivity under probabilistic models is not yet well-defined in the literature, and part of our future work is to precisely define and develop protocols to achieve it.

3 PROBABILISTIC CONNECTIVITY MODEL

In this section, we present a simple probabilistic connectivity model. Using this model, we can quantify the quality of communication between nodes in sensor networks. We start by defining a quantitative metric for communication quality. Then, we derive bounds for this metric in common node organization schemes. This derivation will be used in designing connectivity maintenance protocols in Section 4.

3.1 Communication quality

The main function of a sensor network is to deliver data gathered by sensors to a processing center for possible actions. Therefore, we believe that the successful data delivery between any pair of nodes in the network is a good candidate for quantifying the communication quality in a sensor network. We quantify successful data delivery from node u to another node v by the probability that v correctly receives a packet transmitted by u. We call this probability the node-to-node packet delivery rate. From the sensor network design perspective, we are interested in the minimum node-to-node packet delivery rate in the network. Thus, we define the network packet delivery rate, referred to simply as the network delivery rate, as follows.

Definition 1 (Network Delivery Rate). The network delivery rate α of a sensor network is the minimum packet delivery rate between any pair of nodes in the network.

Using the network delivery rate, we can define a probabilistic connectivity model for sensor networks as follows:

Definition 2 (α-connectivity). A sensor network is said to be α-connected if the probability of delivering a packet between any arbitrary pair of nodes (i.e., network delivery rate) is at least α, where $0 \leq \alpha \leq 1$.

In contrast to the deterministic connectivity model, the α-connectivity model provides a quantitative metric for measuring the communication quality in a sensor network. This is not only desirable, but also critical for sensor network applications that do require bounding the probability of losing a potentially important data item, such as intrusion detection systems in military applications. Notice also that higher network delivery rates imply higher throughput and shorter delay in the sensor network, because more packets make it through the network to their destinations without the need for retransmissions. In that sense, the network delivery rate represents a basic communication quality metric from which higher level metrics can be inferred based on the specific application of the sensor network.

We derive bounds for α for different node organization methods in the following subsection. In Section 4, we propose a distributed protocol that achieves α-connectivity in sensor networks. The idea of the protocol is to activate nodes in a distributed manner such that they form a structure for which we can compute α. Notice that it is the activated nodes that form the structure, not all deployed nodes. That is, the initial deployment of nodes in the field can be done randomly, which is a realistic deployment method for large-scale sensor networks especially in hostile and dangerous environments.

3.2 Computing network delivery rates

We model a sensor network as a *weighted* graph $G(V, E)$, where V is the set of all nodes, and E is the set of edges between nodes. Every pair of nodes $u, v \in V$ have an edge $u \rightarrow v$ labeled with a packet delivery rate $p(u, v)$. $p(u, v)$ represents the probability of delivering packets from u to v over the *direct* wireless channel between them. Clearly, $p(u, v)$ depends on the probabilistic communication model used for the communication ranges of sensors. In addition, packets may flow between two nodes through multiple paths. We denote the total probability of delivering packets from node u to node v over all possible paths as $R(u, v)$. We refer to $R(u, v)$ as the node-to-node packet delivery rate.

The above graph representation of sensor networks is fairly general. For instance, it allows the creation of links between distant nodes. It also allows sensors to employ different communication models. It is, however, quite difficult to analytically compute the exact value of the network delivery rate α in this general setting. Therefore, we compute *lower bounds* on α under the following assumptions.

- All sensors use the same probabilistic communication model. This is not unrealistic assumption in many applications. For example, nodes in a surveillance application deployed in open areas could use the log-normal shadowing model [22], which captures path loss, shadowing effects, and Gaussian noise. Similarly, the same model could be used by nodes in a military intrusion detection system that are deployed on the ground at the same elevation. In addition, nodes in a forest fire detection system can all use a communication model that captures the characteristics of the surrounding environment such as the signal reflections from trees. Note that this assumption does not say that all nodes are deterministically identical, rather they follow the same probabilistic model. That is, the packet delivery rates over direct links have the same average $p = p(u, v)$.
- Links starting at the same sender node have independent delivery rates. For example, in Fig. 1, the packet delivery rates $p(u, v)$ and $p(u, w)$ are independent. This assumption is needed to make the analysis tractable, otherwise, the analysis is not possible unless the nature of the dependence between links is completely specified. In our simulations, we do not assume independence and we verify that our results still hold.
- We only consider the delivery rates between immediate neighbors. For example, in Fig. 1, the *direct* delivery rate between nodes u and z is assumed to be zero. Therefore, our calculation of the network delivery rate is conservative and should be viewed as a lower bound. We notice that this is not totally unrealistic, because as the distance between nodes increases the signal fades rapidly and most wireless receivers process a signal only if its level exceeds a certain threshold.
- No retransmissions in the MAC layer. This assumption is needed to find the minimum network delivery rates regardless of the details of the employed MAC protocol, such as the maximum number of retransmissions and the random backoff scheme. This assumption actually makes our analysis more general, and therefore, our results and the

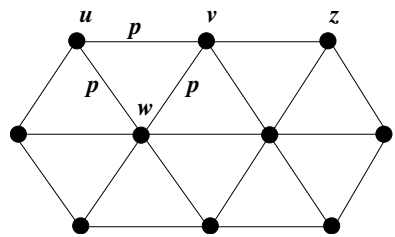
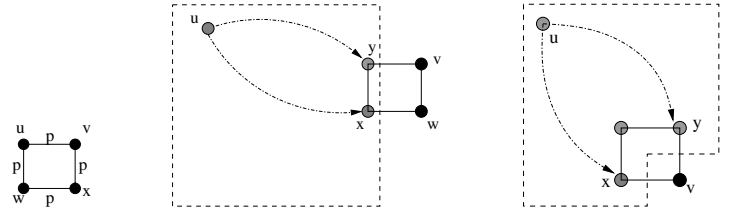


FIGURE 1
Probabilistic connectivity in triangular mesh.



(a) Initial 2×2 mesh (b) Expanding the mesh by 2 nodes (c) Expanding the mesh by 1 node

FIGURE 2
The square mesh construction process used in proving Theorem 1.

proposed connectivity maintenance protocol can be used with different MAC protocols. As shown by our NS-2 simulations (Section 5), which are performed with MAC retransmissions, our analysis indeed provides lower bounds on the network delivery rates.

Under these assumptions, we first derive the lower bound on the network delivery rate α for nodes organized in a square mesh. The following theorem gives this bound.

Theorem 1. *Under the above assumptions, if nodes are activated on a square mesh and the average packet delivery rate between any neighboring nodes is p, then the network delivery rate* α *is at least* $\min\left(\frac{p+p^2-1}{p^3}, p^2-2p\right)$.

Proof. The proof is by construction. We start with a 2×2 mesh and iteratively expand it till it contains all nodes, while maintaining the lower bound in each step. The expansion process can be done in two ways. First, by adding two nodes with the structure shown in Fig. 2(b): We choose any two neighbor boundary nodes x and y and connect them to two new nodes w and v. In the second type of expansion, we choose two boundary nodes x and y with the structure shown in Fig. 2(c) and connect both of them to a new node v. Using these two types of expansion, it is easy to show that any $n \times m$ square mesh can be constructed starting from a 2×2 mesh.

Now, we derive a lower bound on the network delivery rate for nodes in the initial 2 grid and nodes added during expansion process. Then, we take the minimum of them.

1. *Initial Nodes.* Without loss of generality, assume u is the source (Fig. 2(a)) in the initial grid. We calculate the delivery rate at v, w, and x. There are two paths from u to v with lengths 1 and 3. Therefore, the data will be delivered to v with probabilities p and p^3 respectively. The accumulated probability of delivery at v, $R(u, v)$, is $1-(1-p)(1-p^3)$. The same argument holds for w due to symmetry. On the other hand, there are two paths between u and x both with length 2. Hence, we have

$R(u, v) = 1 - (1 - p)^2$. Since $R(u, x)$ has the minimum value among these four nodes, we have:

$$\alpha = 1 - (1 - p)^2 \tag{1}$$

2. *Added Nodes.* In the first expansion type (Fig. 2(b)), data can be delivered from the source to node v through two paths yv and xwv. The delivery rates obtained from each of these paths are $R(u, y)p$ and $R(u, x)p^2$. Since $R(u, x)$ and $R(u, y)$ are both greater than or equal to α, we have $R(u, v) \geq 1 - (1 - p\alpha)(1 - p^2\alpha)$. The same analysis holds for w due to symmetry. On the other hand, it can be easily observed that for the second expansion method, we have $R(u, v) \geq 1 - (1 - p\alpha)^2 \geq 1 - (1 - p\alpha)(1 - p^2\alpha)$.

   Now consider the two nodes with the least node-to-node delivery rate. Call them i and j. We have $R(i, j) = \alpha$. From the above analysis, we have $R(i, j) \geq 1 - (1 - p\alpha)(1 - p^2\alpha)$. Therefore:

$$\alpha \geq \frac{p + p^2 - 1}{p^3}. \tag{2}$$

We conclude that α is greater than or equal to the minimum value of (1) and (2), and the proof follows. □

The above theorem gives the delivery rate if nodes in the sensor network are activated on a square mesh. Next, we derive the lower bound of the network delivery rate if nodes are activated on a triangular mesh.

Theorem 2. *If nodes are activated on a triangular mesh and the average packet delivery rate between any neighboring nodes is p, then the network delivery rate* α *is at least* $(2p - 1)/p^2$.

Proof. We prove this by construction. First, we begin with a triangle. Then, we expand it by adding nodes one by one to the triangular mesh as in Fig. 3. We find the delivery rate between an arbitrary pair of nodes u and v at each step. There are two links connecting v to x and y. Therefore, the accumulated delivery rate at v is $1 - (1 - pR(u, x))(1 - pR(u, y))$. Since $R(u, x)$ and $R(u, y)$ are greater than or equal to α, we get $R(u, v) \geq 1 - (1 - p\alpha)^2$. This result is true for every pair of nodes.

Now, assume two nodes, i and j, with the least node-to-node delivery rate. By definition, we have $R(i, j) = \alpha$. On the other hand, we have $R(i, j) \geq 1-(1-p\alpha)^2$ from the above discussion. Therefore, we have $1-(1-p\alpha)^2 \leq \alpha$, or $\alpha \geq (2p - 1)/p^2$. □

Finally, in [2] we extend the analysis of the network delivery rate to uniform random node activation. This is useful, for example, when nodes are deployed randomly and all of them are activated to ensure coverage and connectivity. This is a special case for sensor networks that are deployed for

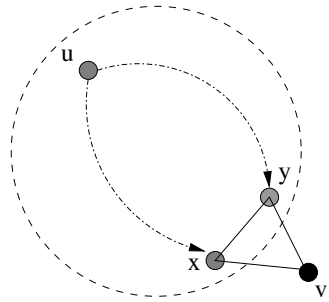


FIGURE 3
The triangular mesh construction used in proving Theorem 2.

a limited time. In such networks, our analysis can provide insights on the required node density to meet a given network delivery rate. These special networks, however, do not make use of distributed protocols that try to activate the minimum subset of nodes need to maintain coverage and connectivity, since all nodes are active. Since the focus of this paper is on distributed connectivity and coverage protocols, we omit the analysis for the random node activation case.

4 PROBABILISTIC CONNECTIVITY MAINTENANCE PROTOCOL

In this section, we present a new Probabilistic Connectivity Maintenance Protocol (PCMP), which employs probabilistic communication models. We start by presenting an overview of our protocol, followed by more details. Then, we show how our protocol can provide probabilistic coverage and connectivity at the same time.

4.1 Overview of PCMP

The goal of PCMP is to activate a subset of deployed nodes such that the probability of delivering packets between any arbitrary nodes in the network is at least α, i.e., keep the network α-connected. To achieve this goal, the protocol activates nodes to form an approximate triangular mesh. The activation process is done in a distributed manner as described below. The spacing between nodes in the triangular mesh is computed to achieve the target network delivery rate. We use the bound proved in Theorem 2 and information from the adopted communication model in computing the spacing. The details of this computation are given in Section 4.2. For now, let us assume that the spacing between nodes is determined to be d. We chose to activate nodes on a triangular mesh for three reasons. First, it enables us to use PCMP with the deterministic connectivity model, in addition to the probabilistic model. In this

case, activating nodes on the triangular mesh has been shown to be optimal in terms of number nodes activated [4,18]. Second, our analysis for the triangular mesh in Section 3.2 provides a simpler and tighter lower bound than the analysis for the square mesh, as confirmed by our simulations. Third, it makes the integration with probabilistic coverage protocols easier, as described in Section 4.3. We *emphasize* that PCMP does *not* require nodes to be deployed on a triangular mesh. It is the activated subset of them that forms a triangular mesh. Node deployment can follow any distribution. In our simulations, we deploy nodes uniformly at random.

The main idea of our protocol is to start the activation process by one node, and iteratively activate other nodes until a triangular mesh-like structure is formed over the whole area. PCMP works in rounds of R seconds each, where in each round a subset of nodes are active to maintain the whole network connected and the rest of the nodes are put in sleep mode to conserve energy. R is chosen to be much smaller than the average lifetime of sensors. In the beginning of each round, all nodes start running PCMP independent of each other. This implies that nodes need to be time-synchronized. In our simulations, we show that only coarse-grained time synchronization is needed and PCMP is quite robust to clock drifts. A number of messages will be exchanged between nodes to determine which of them should be active during the current round, and which should sleep till the beginning of the next round. The time it takes the protocol to determine active/sleep nodes is called the convergence time.

In PCMP, a node can be in one of four states: ACTIVE, SLEEP, WAIT, or START. In the beginning of a round, each node sets its state to be START, and selects a random startup timer T_s proportional to its remaining energy level. The node with the smallest T_s will become active, and broadcasts an activation message to all nodes in its communication range. The sender of the activation message is called the activator. The activation message should contain the coordinates of the activator. That is, PCMP assumes that nodes know their locations, which can be done by any efficient localization scheme such as [12, 24]. In the evaluation section, we show that PCMP is robust to inaccuracy of node locations, and thus require only light-weight localization schemes. The activation message tries to activate nodes at vertices of the hexagon centered at the activator, while putting all other nodes within that hexagon to sleep. A node receiving the activation message can determine whether it is a vertex of the hexagon by measuring the distance and angle between itself and the activator. If the angle is multiple of $\pi/3$ and the distance is d, then the node sets its state to ACTIVE and it becomes a new activator. Otherwise it goes to SLEEP state.

Nodes may not always be found at vertices of the triangular mesh because of randomness in node deployment or because of node failure. PCMP tries to activate the closest nodes to hexagon vertices in a distributed manner. Every node receiving an activation message calculates an activation timer T_a as a function of its closeness to the nearest vertex of the hexagon using the

following equation: $T_a = \tau_a(d_v^2 + {d_a}^2\gamma^2)$, where d_v and d_a are the Euclidean distances between the node and the vertex, and the node and the activator, respectively; γ is the angle between the line connecting the node with the activator and the line connecting the vertex with the activator; and τ_a is a constant. Notice that the closer the node gets to the vertex, the smaller the T_a will be. After computing T_a, a node moves to WAIT state and stays in this state till its T_a timer either expires or is canceled. When the smallest T_a timer expires, its corresponding node changes its state to ACTIVE. This node then becomes a new activator and broadcasts an activation message to its neighbors. When receiving the new activation message, nodes in WAIT state cancel their T_a timers and move to SLEEP state.

4.2 Details of PCMP

In this section, we show how the spacing between activated nodes in the triangular mesh is computed to achieve α-connectivity. We refer to this spacing as d_α. To make our discussion concrete, we will derive d_α for the widely-used log-normal shadowing model. Computing d_α for other communication models, such as the ones in [39], can be done in a similar way. As noted in Section 2, our focus in this paper is not to develop new communication models, but rather to design connectivity maintenance protocols that can employ different realistic communication models.

The nodes activated by our PCMP protocol form an approximate triangular mesh. The spacing between these nodes is at most d_α. According to Theorem 2, the network delivery rate α in the triangular mesh is at least $(2p-1)/p^2$, where p is the average packet delivery rate on a link between two neighboring nodes. That is, $\alpha \geq (2p-1)/p^2$. Therefore, we need:

$$p \geq (1 - \sqrt{1-\alpha})/\alpha \tag{3}$$

to meet the target network delivery rate. p is related to the spacing d_α through the assumed communication model. Thus, we use the communication model to compute d_α to yield the required p. To illustrate, consider the log-normal shadowing model widely used in network simulators, such as NS-2 [28] and OPNET [20], and several previous works, e.g., [14, 27]. In this model, the power of the received signal $P_r(d)$ at a distance d from a sender transmitting at power P_t is given by [22, Sec. 3.9]:

$$P_r(d) = P_t - \left(\overline{PL(d_0)} + 10n \log\left(\frac{d}{d_0}\right) + X_\sigma\right), \tag{4}$$

where X_σ is a zero-mean random variable with Gaussian distribution, n is a constant (path loss exponent) specified by the environment, and $\overline{PL(d_0)}$ is the mean path loss measured at the reference distance d_0, which is usually set to 1 m. Note that $P_r(d)$, P_t, and $PL(d_0)$ are all in dBm. Wireless adapters can

successfully receive data if the signal strength exceeds a certain threshold, say γ. The probability that the signal strength exceeds γ is:

$$\Pr[P_r(d) > \gamma] = \frac{1}{2}\left[1 - \operatorname{erf}\left(\frac{\gamma - \overline{P_r(d)}}{\sigma\sqrt{2}}\right)\right]. \tag{5}$$

Assuming that the signal strength does not significantly change during the transmission of a single packet, the average packet delivery rate p is given by $p = \Pr[P_r(d) > \gamma]$. Solving (3) and (5) for the spacing d_α, we get:

$$d_\alpha \leq d_0 \exp\left[\left(P_t - \gamma - \overline{PL(d_0)} + \sigma\sqrt{2}\operatorname{erf}^{-1}\left(1 - 2\frac{1-\sqrt{1-\alpha}}{\alpha}\right)\right) \Big/ 10n\right]. \tag{6}$$

Setting the spacing between activated nodes on the triangular mesh according to (6) will achieve the target network delivery rate under the log-normal shadowing model.

We emphasize that the operation of our PCMP protocol does not depend on the adopted communication model. PCMP needs only the value of d_α, and the protocol functions the same regardless of the model. Thus, PCMP can be used with different communication models.

4.3 Integrated probabilistic coverage and connectivity

The problem of connectivity is closely related to the problem of coverage: It is typically requested that sensors cover the monitored area and they form a connected network for delivering collected data. Similar to communication ranges, sensing ranges have been shown to deviate from the regular disk model and they are better modeled by probabilistic distributions [3,6,17,37]. In this section, we show how our PCMP protocol can be extended to achieve probabilistic coverage as well. We start by defining the term probabilistic coverage and how it can be achieved using distributed protocols. Then, we present an integrated protocol that achieves both probabilistic coverage and connectivity.

In our previous work [13], we have proposed a coverage protocol that accounts for probabilistic sensing models. We did not address probabilistic communication models in [13]. We summarize the basic idea of the probabilistic coverage protocol below. This is needed to clarify the presentation of the proposed integrated protocol.

Under probabilistic sensing models, the sensing range of sensors is not given by a regular disk. Rather, it is described by a probability function. In this case, we define the probabilistic coverage of an area with a threshold parameter β (β-coverage) as follows.

Definition 3 (β-Coverage). An area A is β-covered ($0 < \beta \leq 1$) by n sensors if $P(x) = 1 - \prod_{i=1}^{n}(1 - p_i(x)) \geq \beta$ for every point x in A, where $p_i(x)$ is the probability that sensor i detects an event occurring at x.

Note that $P(x)$ in the above definition measures the collective probability from all n sensors to cover point x, $p_i(x)$ is specified by the sensing model, and the coverage threshold parameter β depends on the requirements of the target application. The sensing model depends on the physical phenomenon being detected and on the environment in which sensors are deployed, which can be estimated empirically. Several models have been proposed in the literature [3, 6, 17, 37]. For example, the exponential sensing model assumes that the sensing capacity degrades according to an exponential distribution after a certain threshold distance from the sensor. The exponential sensing model is defined as:

$$p(x) = \begin{cases} 1, & \text{for } x \leq r_s \\ e^{-\gamma(x-r_s)}, & \text{for } x > r_s \end{cases} \tag{7}$$

where r_s is a threshold distance below which the sensing capacity is strong enough such that any event will be detected with probability 1, and γ is a factor that describes how fast the sensing capacity decays with distance. This model is conservative as it assumes that the sensing capacity decreases exponentially fast beyond r_s, which means that the achieved actual coverage will be higher than the estimated by the theoretical analysis. Because it is conservative, the exponential sensing model can be used as a first approximation for other sensing models such as those in [3, 6, 17].

The main idea of our probabilistic coverage protocol is to ensure that the least-covered point in the monitored area has a probability of being sensed that is at least β. The least-covered point is the point that has the smallest probability of coverage in the whole area. To implement this idea in a distributed protocol with no global knowledge, we divide the area into smaller subareas. For each subarea, we determine the least-covered point in that subarea, and we activate the minimum number of sensors required to cover the least-covered point with a probability more than or equal to β. To enable the protocol to work optimally under the disk sensing model as well as probabilistic sensing models, we divide the monitored area into equi-lateral triangles forming a triangular lattice. We showed in [13] that the least-covered point by three sensors deployed at vertices of an equi-lateral triangle is at the center of the triangle. Knowing the location of the least-covered point, we can compute the maximum length of the triangle side at which the probability of sensing at the least-covered point is at least β. For example, it is shown in [13] that the maximum separation d_β is $\sqrt{3}\left(r_s - \frac{\ln(1-\sqrt[3]{1-\beta})}{\gamma}\right)$ for the exponential sensing model, and d_β is $\sqrt{3}r_s$ for the disk sensing model. The probabilistic coverage protocol activate nodes to form an approximate triangle lattice, where the side of the triangles is d_β.

Now the integration of the coverage and connectivity protocol becomes straightforward. The integrated protocol computes two spacing values d_β and d_α, where the former assures β-coverage and the latter assures α-connectivity.

The protocol sets the spacing of the approximate triangular mesh formed by the activated sensors as $\min(d_\alpha, d_\beta)$ to make the sensor network α-connected and the area β-covered at the same time. Notice that the protocol does not require any strict relationship between the communication model and the sensing model and therefore it is fairly general. In the evaluation section, we verify that the protocol indeed achieves both α-connectivity and β-coverage.

5 EVALUATION

In this section, we rigorously evaluate our proposed protocol and compare it against others. We first describe our experimental setup. Then, we validate our theoretical lower bounds on network delivery rate derived in Section 3. This is followed by an analysis of potential benefits of using probabilistic communication models. Then, we study the performance of our protocol and show its robustness against node location inaccuracy, node failures, and imperfect time synchronization of nodes. We show that our protocol scales to sensor networks with thousands of nodes. We also verify that our protocol can achieve both probabilistic coverage and connectivity. We then show that our protocol outperforms two of the best connectivity protocols in the literature: SPAN [10] and GAF [33]. Finally, we demonstrate the generality of our protocol by showing that it can provide coverage and connectivity under deterministic models as well. And in this case it outperforms the recent integrated coverage and connectivity protocol in the literature (CCP-SPAN [31]).

5.1 Experimental setup

We use the following setup in all of our experiments, unless otherwise specified. We use NS-2 version 2.30 [28] in the simulation. We deploy 1,000 nodes uniformly at random in a 1 km by 1 km area. All nodes use the log-normal shadowing model given in Eq. (4) for radio communications, with the parameters listed in Table 1. We employ the energy model specified in [34]. In this model, the power consumption for transmission, reception, idling, and sleeping are 60, 12, 12, 0.03 mWatt, respectively. We use an initial energy of 60 Jules for each node.

Parameter	Value
Path-loss exponent n	2.2
Shadowing standard deviation σ	4.0
Reference distance d_0	1 m
Transmission power P_t	1 mW
Reception threshold γ	10^{-9} mW

TABLE 1
Parameters used for the log-normal shadowing communication model

We set the wireless channel bandwidth at 40 kbps. For our PCMP protocol, we have the following parameters. The round length R is 100 seconds, which is much smaller than the network lifetime. The average message size is 34 bytes. The maximum value for the startup timer τ_s is set to $1/E_r$, where E_r is the fraction of the remaining energy in the node.

We repeat each experiment 10 times with different seeds, and we report the average values over all of them. We also show the minimum and maximum values if they do not clutter the graph. Because of the large-scale of our experiments (1,000 nodes) and the detailed simulation of the wireless environment implemented in NS-2, in many cases a single replication of one experiment took several hours of processing on a decent server with eight CPU cores. It is why we repeat each experiment only 10 times.

5.2 Validation of our analysis

We validate that our lower bounds on the network delivery rates in different node deployments still hold when the assumptions made in Section 3.2 are relaxed. In the first set of experiments, we measure the network delivery rate from simulation and compare it against the analytical lower bound. To measure the network delivery rate, we randomly choose a node, and make it broadcast 1,000 small packets of size 8 bytes each to all other nodes. Then, we record the number of received packets at each node. The network delivery rate is the minimum number of received packets by any node divided by 1,000.

For the triangular mesh, we activate 100 nodes (out of the 1,000 deployed) and make them form a triangular mesh over the whole area. We vary the spacing between neighboring nodes d by varying the area over which the triangular mesh is constructed. Varying the spacing between nodes corresponds to varying the probability of delivering packets between neighboring nodes. We run the simulation for each value of d 10 times and we measure the network delivery rate. We also compute the network delivery rate from Theorem 2 for each value of d. We repeat the experiment again, except we vary the transmission power P_t and fix everything else in the communication model. The results shown in Fig. 4 confirm that our lower bound is correct and conservative.

The above experiment is repeated for the square mesh and uniform deployments, and similar results were obtained. The details and figures are given in [2], which all validate our analysis.

5.3 Benefits of using probabilistic communication models

We discuss potential benefits of using probabilistic communication models. Connectivity maintenance protocols that rely on the disk model typically assume a conservative value for the communication range: A value within which the signal is quite strong to be received. Otherwise, the network may become disconnected according to the deterministic model. Therefore, this model may unnecessarily activate nodes to maintain connectivity. In contrast, the proposed probabilistic α-connectivity model can consider the full

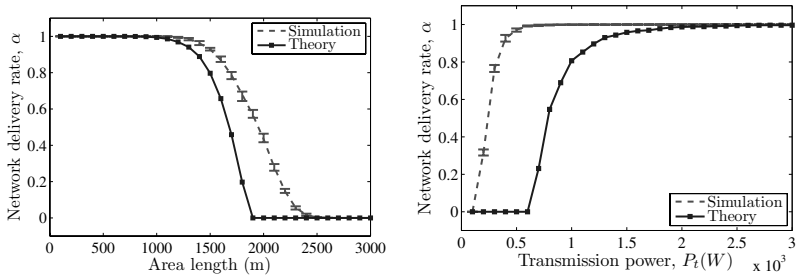


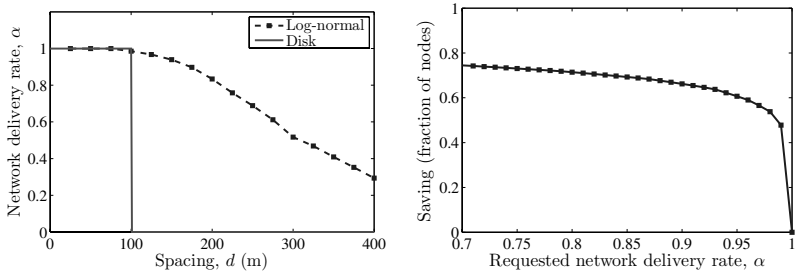
FIGURE 4
Validation of our analysis. Comparing the network delivery rate in the triangular mesh estimated by Theorem 2 versus the achieved delivery rate in the simulation for different: (a) spacing between nodes, and (b) transmission power. For the simulation data, we show the minimum, average, and maximum values.

communication range of nodes. It also provides the network designer with a tunning knob, which is the level of connectivity α. Setting α to higher values will result in fewer lost packets, but will require activating more nodes, and vice versa.

We design an experiment to analyze this tradeoff. We use the log-normal shadowing model and vary the spacing between nodes in the triangular mesh. We measure the network delivery rate and plot the results in Fig. 5(a). On the same figure, we also plot the network delivery rate when the deterministic communication model is used with a communication range set to 100 m, which is computed based on the reception threshold power given in Table 1. According to the deterministic model, the network delivery rate is zero if the spacing between nodes is more than 100 m. Whereas under the more realistic probabilistic model, the network delivery rate gradually decreases. For example, if the application of the sensor network can tolerate 20% loss rate (i.e., $\alpha = 0.80$), we could approximately double the spacing between activated nodes compared to the deterministic model. This leads to significant savings in number of activated nodes, and therefore prolongs the network lifetime. The potential saving in number of activated nodes for different values of α is plotted in Fig. 5(b). Many sensor network applications could benefit from this tradeoff. For example, consider a monitoring system where nodes periodically measure temperature and humidity and report them to a processing center. Since these physical phenomena do not change suddenly, the network could tolerate some losses of the reported data, because the change will persist over several measurement periods.

5.4 Performance and robustness of PCMP

In this section, we study the performance of our PCMP protocol and assess its robustness against inaccuracy in node locations obtained by localization mechanisms, imperfect time synchronization, and random node failures.



(a) (b)

FIGURE 5
Potential benefits of using probabilistic communication models over the deterministic disk model: (a) achieved network delivery rate, and (b) saving in number of activated nodes.

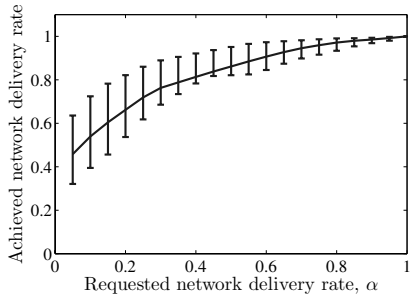


FIGURE 6
PCMP achieves the requested network delivery rate in all cases. The minimum, average, and maximum values are shown.

We run PCMP over 1,000 uniformly deployed nodes that use the log-normal shadowing model. We vary the requested network delivery rate α between 0.1 and 1.0. For each value of α, we compute the spacing between neighboring nodes d_α from Eq. (6), and we run PCMP in the simulation with this value. We measure the achieved network delivery rate by PCMP. The results shown in Fig. 6 demonstrate that our protocol met the requested network delivery rate in all cases.

Next, we study the robustness of PCMP against inaccuracy in node locations. We use the same setup as before except that we add errors to node locations. We add random values in the interval $[-er_{\max}, er_{\max}]$ to both x and y coordinates of the real location of each deployed node. We vary $er_{\max}$ between 0 and 20 m. We compute the network delivery rate after the protocol converges. The results indicate that the network delivery rate is always maintained as shown in Fig. 7(a). Therefore, PCMP is robust against location inaccuracy. There is a small cost, however, with this location inaccuracy. As shown on the same figure (notice the two y-axes), the number of activated

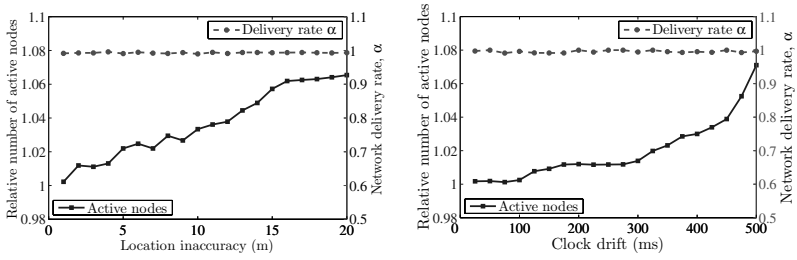


FIGURE 7
Robustness of PCMP against: (a) location inaccuracy, and (b) clock drifts.

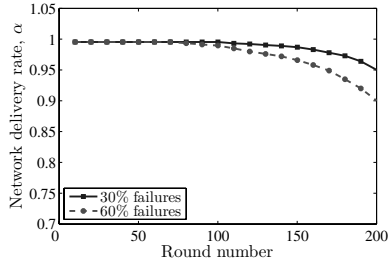


FIGURE 8
Robustness of PCMP against random node failures.

nodes slightly increases in case of inaccurate locations. There is less than 7% increase in number of activated nodes for location errors of up to ± 20 m.

Exact time synchronization of nodes in a large scale sensor network is costly to achieve. We study the robustness of PCMP against the granularity of time synchronization. To do this, we add random values in the interval $[0, d_{\max}]$ to the clock of each node at the beginning of the simulation. We change $d_{\max}$ between 0 and 500 ms. As shown in Fig. 7(b), the network delivery rate is ensured even with high values of clock drift. In addition, the number of active nodes does not increase if the drift is less than the convergence time of the protocol (around 75 ms). This means that our protocol is robust against fairly large clock drifts, and thus, it needs only light-weight, coarse-grained, time synchronization schemes.

Finally, we show that PCMP is robust against random node failures. We choose a fraction f of all deployed nodes to be failed within the first 200 rounds of the protocol execution, and we randomly schedule them to fail. We change the fraction of failed nodes, f, and plot the network delivery rate as time progresses in Fig. 8. The results indicate that PCMP can ensure network delivery rate even with high failure rates.

5.5 Scalability of PCMP

We study the scalability of the PCMP protocol as the number of nodes increases and as the monitored area increases. The NS-2 simulator did not support large-scale experiments with thousands of nodes. Therefore, we have developed our own packet-level simulator in C++. Using our simulator, we could evaluate sensor networks with more than 50,000 nodes. Our simulator is used only to conduct the scalability experiments in this section.

In the first set of experiments, we fix the area size at 2 km by 2 km. We change the number of nodes deployed in the area between 5,000 and 50,000 with a step of 5,000. Other parameters are fixed as described in Section 5.1. For each deployed number of nodes, we repeat the experiment 10 times, and we measure the minimum, average, and maximum values of the convergence time, number of active nodes, and total energy consumed. The energy is measured after running the protocol for 10 rounds, and each round lasts for 100 seconds. Some of our results are shown in Fig. 10. Figure 10(a) indicates that the convergence time of the PCMP protocol does not increase as the node density increases. It also shows that the convergence time is fairly small, the average is about 200 msec. Figure 10(b) plots the fraction of remaining energy in all nodes in the network. The figure shows that as the node density increases, PCMP activates smaller portions of the deployed nodes, thus it conserves energy and prolongs the network lifetime. This set of experiments confirms that PCMP can scale to large-scale sensor networks with thousands of nodes.

In the next set of experiment, we fix the deployed number of nodes at 20,000 and change the area size. The length of the (square) area is varied between 1 km to 5 km, with an increment of 0.25 km. We measure the same metrics as before. As an example, the results for the convergence time are given in Fig. 11. The figure implies that the convergence time slowly increases as the size of the monitored area increases. However, the convergence time is still small for areas as large as 5 km by 5 km. The convergence time is under 1 second in most cases for a round of length 100 seconds. The increase in the convergence time happens because PCMP incrementally constructs an approximate triangular mesh over the area. Larger areas would take longer to cover, because PCMP begins each round with only one active node. If shorter convergence times are desired, PCMP can be configured to start each round with multiple active nodes.

5.6 Integrated probabilistic coverage and connectivity

In Section 4.3, we described how PCMP can maintain both coverage and connectivity under probabilistic sensing and communication models. We conduct an experiment to verify this. In addition to the log-normal shadowing communication model, we use the exponential sensing model [38] for coverage. The exponential model assumes that after a threshold distance r_s, the sensing capacity of a node decreases exponentially fast. We fix r_s, and we run our protocol with various spacing d. Then, we measure the achieved

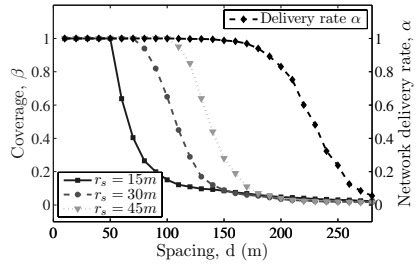
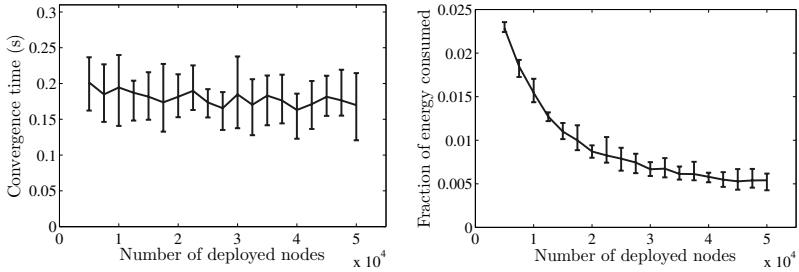


FIGURE 9
Probabilistic coverage and connectivity achieved by PCMP.



(a) (b)

FIGURE 10
Scalability of PCMP as the node density increases.

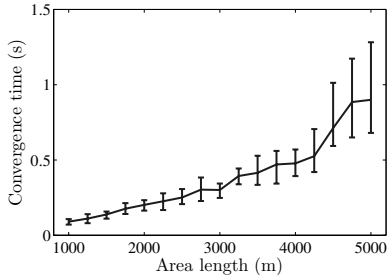


FIGURE 11
Scalability of PCMP as the area size increases.

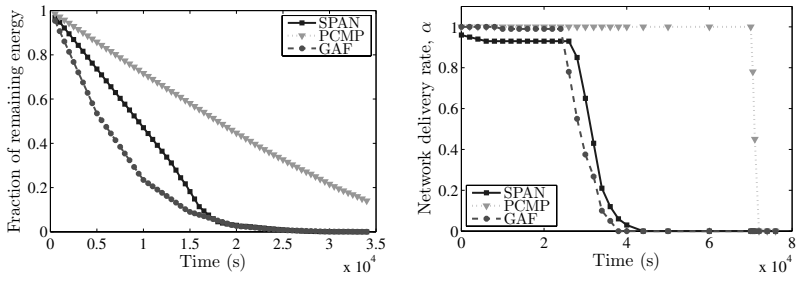
β-coverage and α-connectivity. We repeat for a few values of r_s. The results shown in Fig. 9 indicate that our protocol achieves both goals if the spacing is set according to our discussion in Section 4.3. For example, to achieve 0.95-connectivity, we need d_α to be about 175 m. To archive 0.9-coverage, d_β should be around 115 m. Running PCMP with $d = \min\{d_\alpha, d_\beta\} = 115$ m achieves 0.95-connectivity and 0.9-coverage.

5.7 Comparing PCMP against other connectivity protocols

We compare our PCMP protocol against SPAN [10] and GAF [33] protocols since they are the best and widely cited other protocols in the literature. Both protocols were described in Section 2. We use the NS-2 code for SPAN which is published by its authors at [26]. The code for GAF is included in version 2.30 of NS-2. We use the energy model (described in Section 5.1) for all three protocols.

First, we verify that all protocols indeed achieve the disk-model deterministic connectivity. We check this for two different node deployment densities. We set the communication range of nodes to 100 m. The length of GAF grid cells are set to 44 m, according to the relationship presented in [33]. To measure connectivity, we run a breadth first search to find the largest connected component of nodes. We divide the size of this component by the total number of nodes. The results show that all protocols achieve 100% deterministic connectivity.

Next, we compare the three protocols against a critical metric in sensor networks: energy consumption. We fix all parameters in the simulator and run the three protocols one at a time. We periodically collect the amount of remaining energy in every deployed node. Then, we sum these values and compute the fraction of energy remained in the network with respect to the initial energy at time 0. For each protocol, we run the simulator 10 times, and for long periods (35,000 seconds). The average results are shown in Fig. 12(a). As the figure shows, PCMP consumes much less energy than the other two protocols. For example, after 15,000 seconds from the start, nodes under SPAN and GAF have less than 20% of their initial energy, while using PCMP nodes have 60% of their initial energy. The reasons behind the energy saving of PCMP over GAF is that PCMP activates much fewer number of nodes than GAF: The average number of active nodes under PCMP was always less than 70 in all cases, while GAF activated at least 160 nodes. Nodes in active mode consumes significantly more energy than nodes in sleep mode. On the other hand, SPAN activates slightly less number of nodes than PCMP, but it has much higher communication overhead due to the frequent exchange of hello messages among nodes.



(a) (b)

FIGURE 12
Comparing PCMP against SPAN and GAF: (a) energy consumption, and (b) network lifetime.

Finally, we compare the network lifetime under the three protocols. Since these are connectivity protocols, we plot the average network packet delivery rate as the time progresses. As demonstrated by Fig. 12(b), our protocol extends (almost doubles) the lifetime of the network. This is because of the energy saving as described above.

5.8 Comparing PCMP against another integrated coverage and connectivity protocol

As described in Section 4.3 and verified in Section 5.6, our protocol achieves coverage and connectivity under probabilistic communication and sensing models. In this section, we show that our protocol can be used with deterministic communication and sensing models as well, and outperforms the stat-of-the-art protocol in this category.

We compare our protocol against an integrated protocol called CCP-SPAN [31]: It uses CCP for coverage and SPAN for connectivity. CCP is a distributed coverage protocol, which tries to deactivate nodes providing redundant coverage. To achieve this, a node in CCP checks the intersection points of its sensing circle with other circles of neighboring nodes. If all intersection points are covered, the node turns itself off. A node in SPAN, on the other hand, checks whether each pair of its neighbors can reach each other either directly or through at most two hops. If this is the case, a node turns itself off. The integrated CCP-SPAN protocol checks the two conditions to turn a node off. We use the NS-2 code of CCP-SPAN provided by its authors [8]. The sensing range used in this experiment is $r_s = 50$ m, and the communication range is $r_c = 100$ m.

We compare the number of activated nodes by the two protocols. Again, we fix all parameters in the simulator and we run the two protocol separately for 10 times each. We repeat the whole experiment for different node deployment densities. We plot the results in Fig. 13(a). The figure indicates that PCMP activates almost 50% less nodes than CCP-SPAN to provide the same coverage and connectivity.

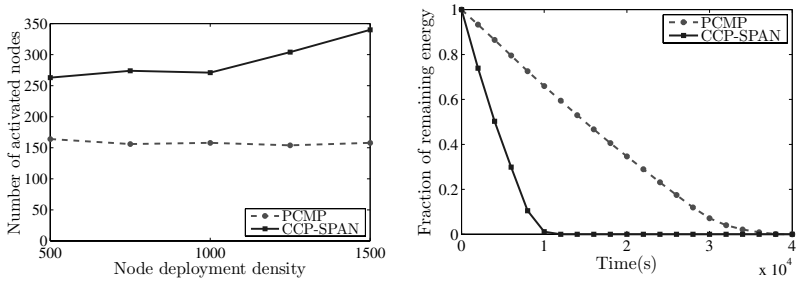


FIGURE 13
Comparing PCMP against CCP-SPAN: (a) Number of activated nodes, and (b) Energy consumption.

In our last experiment, we compare the energy consumption of PCMP and CCP-SPAN as time progresses. We run both protocols separately and periodically report the total amount of energy remained in all deployed nodes normalized by their initial energy. Figure 13(b) shows that the energy consumed by CCP-SPAN is about four times more than that is consumed by PCMP. This implies that sensor networks using our integrated PCMP protocol will have substantially longer lifetimes than if they were to use CCP-SPAN.

6 CONCLUSIONS AND FUTURE WORK

We presented a simple probabilistic connectivity model under which we could quantify the quality of communication between nodes in wireless sensor networks. We introduced the network packet delivery rate as a quantitative metric for communication quality. We derived lower bounds for this metric in three common node deployment schemes: triangular mesh, square mesh, and uniform. Based on the probabilistic connectivity model, we designed a distributed Probabilistic Connectivity Maintenance Protocol (PCMP). PCMP is a fairly general protocol that can employ different probabilistic as well as deterministic communication models, with minimal configuration.

Through extensive simulations in NS-2 with nodes using the log-normal shadowing model for their radio communications, we showed that: (i) PCMP achieves the target network delivery rates; (ii) PCMP is quite robust to several factors common in real environments such as node failures, drifts in node clocks, and errors in node locations; and (iii) Probabilistic communication models expose a tradeoff between packet delivery rates and number of activated nodes, which could be exploited by sensor network designers to optimize the number of deployed nodes. This tradeoff was not possible to analyze under the traditional deterministic communication model.

We compared our protocol versus two of the best connectivity maintenance protocols in the literature: SPAN [10] and GAF [33]. Our simulation results demonstrated that our protocol significantly outperforms them in several aspects, including number of activated nodes, energy consumption, and network lifetime. In addition, we showed how our protocol can be extended to provide probabilistic coverage and connectivity at the same time, and we verified that it indeed achieves both of them through simulation. To the best of our knowledge, our protocol is the first to employ both probabilistic communication and probabilistic sensing models. Therefore, our protocol is more suitable for real sensor network environments than most others in the literature.

Finally, we demonstrated how our protocol can provide both coverage and connectivity under the common deterministic disk model as well. In this case, our simulations showed that our integrated protocol outperforms the state-of-the-art integrated coverage and connectivity protocol in the literature, CCP-SPAN[31], by a wide margin.

The work in this paper can be extended in several directions. One possible extension is to consider different communication models for nodes deployed in the area and forming one network. Different models are needed if heterogeneous nodes are deployed, or the environmental conditions vary significantly from one location to another. For example, some nodes could be deployed on the ground while others are deployed at different heights on a mountain. Another extension is to consider coverage and connectivity with degrees higher than one under probabilistic sensing and communication models.

REFERENCES

[1] Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, and Robert Morris. Link-level measurements from an 802.11b mesh network. In *Proceedings of the SIGCOMM'04*, Portland, OR, August 2004, pp. 121–132.

[2] H. Ahmadi. Probabilistic coverage and connectivity in wireless sensor networks. Master's thesis, School of Computing Science, Simon Fraser University, Canada, August 2007.

[3] Nadeem Ahmed, Salil Kanhere, and Sanjay Jha. Probabilistic coverage in wireless sensor networks. In *Proceedings of the IEEE Conference on Local Computer Networks (LCN'05)*, Sydney, Australia, November 2005, pp. 672–681.

[4] Xiaole Bai, Santosh Kuma, Dong Xua, Ziqiu Yun, and Ten H. La. Deploying wireless sensors to achieve both coverage and connectivity. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'06)*, Florence, Italy, 2006, pp. 131–142.

[5] Christian Bettstetter and Christian Hartmann. Connectivity of wireless multihop networks in a shadow fading environment. In *Proceedings of the ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWIM'03)*, San Diego, CA, September 2003, pp. 28–32.

[6] Qing Cao, Ting Yan, Tarek Abdelzaher, and John Stankovic. Analysis of target detection performance for wireless sensor networks. In *Proceedings of the International Conference on Distributed Computing in Sensor Networks*, Marina Del Rey, CA, June 2005, pp. 276–292.

[7] B. Carbunar, A. Grama, and J. Vitek. Redundancy and coverage detection in sensor networks. *ACM Transactions on Sensor Networks*, **2**(1) (February 2006), 94–128.

[8] CCP-SPAN Reference Implementation. http://www.cs.wustl.edu/∼xing/coverage/ccp-span.tar.gz.

[9] Alberto Cerpa and Deborah Estrin. ASCENT: Adaptive self-configuring sensor networks topologies. *IEEE Transactions on Mobile Computing*, **3**(3) (July–August 2004), 272–285.

[10] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris. SPAN: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wireless Networks*, **8**(5) (September 2002), 481–494.

[11] F. Dai and J. Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, **15**(10) (October 2004), 908–920.

[12] L. Doherty, L. El Ghaoui, and K. Pister. Convex position estimation in wireless sensor networks. In *Proceedings of the IEEE INFOCOM'01*, Anchorage, AK, April 2001, pp. 1655–1663.

[13] M. Hefeeda and H. Ahmadi. A probabilistic coverage protocol for wireless sensor networks. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP'07)*, Beijing, China, October 2007.

[14] R. Hekmat and P. Van Mieghem. Connectivity in wireless ad-hoc networks with a log-normal radio model. *Mobile Networks and Applications*, **11**(3) (June 2006), 351–360.

[15] C. Huang, Y. Tseng, and H. Wu. Distributed protocols for ensuring both coverage and connectivity of a wireless sensor network. *ACM Transactions on Sensor Networks*, **3**(1) (March 2007), 1–24.

[16] David Kotz, Calvin Newport, and Chip Elliott. The mistaken axioms of wireless-network research. Technical Report 2003-467, Dartmouth College, Computer Science Department, July 2003.

[17] B. Liu and D. Towsley. A study of the coverage of large-scale sensor networks. In *Proceedings of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'04)*, Fort Lauderdale, FL, October 2004, pp. 475–483.

[18] M. Ma and Y. Yang. Adaptive triangular deployment algorithm for unattended mobile sensor networks. *IEEE Transactions on Computers*, **56**(7) (July 2007), 946–958.

[19] D. Miorandi and E. Altman. Coverage and connectivity of ad hoc networks in presence of channel randomness. In *Proceedings of the IEEE INFOCOM'05*, Miami, FL, March 2005, pp. 491–502.

[20] OPNET Web Page. http://www.opnet.com.

[21] Mathew D. Penrose. On k-connectivity for a geometric random graph. *Random Structures and Algorithms*, **15**(2) (September 1999), 145–164.

[22] Theodore Rappaport. *Wireless communications: principles and practice*. Prentice Hall, 1996.

[23] N. Reijers, G. Halkes, and K. Langendoen. Link layer measurements in sensor networks. In *Proceedings of the 2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'04).*, FL, USA, October 2003, pp. 224–234.

[24] A. Savvides, C. Han, and M. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom'01)*, Rome, Italy, July 2001, pp. 166–179.

[25] D. Simplot-Ryl, I. Stojmenovic, and J. Wu. Energy-efficient backbone construction, broadcasting, and area coverage in sensor networks. In I. Stojmenovic, editor, *Handbook of Sensor Networks: Algorithms and Architectures*, John Wiley & Sons, Inc, 2005, pp. 343–379.

[26] SPAN Reference Implementation. http://pdos.csail.mit.edu/∼benjie/span/span_ns_1.1.tar.gz.

[27] I. Stojmenovic, A. Nayak, J. Kuruvila, F. Ovalle-Martinez, and E. Villanueva-Pena. Physical layer impact on the design of and performance of routing and broadcasting protocols in ad hoc and sensor networks. *Elsevier Computer Communications*, **28**(10) (June 2005), 1138–1151.

[28] The Network Simulator (NS-2) Web Page. http://nsnam.isi.edu/nsnam/.

[29] D. Tian and N. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications*, Atlanta, GA, September 2002, pp. 32–41.

[30] A. Tiwari, P. Ballal, and F. Lewis. Energy-efficient sensor network design and implementation for condition-based maintenance. *ACM Transactions on Sensor Networks*, **3**(1) (March 2007), 1–23.

[31] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration for energy conservation in sensor networks. *ACM Transactions on Sensor Networks*, **1**(1) (August 2005), 36–72.

[32] Ya Xu, John Heidemann, and Deborah Estrin. Adaptive energy-conserving routing for multihop ad hoc networks. Research Report 527, USC/Information Sciences Institute. Available at: http://www.isi.edu/∼johnh/PAPERS/Xu00a.html, October 2000.

[33] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed energy conservation for ad hoc routing. In *Proceedings of the ACM nternational Conference on Mobile Computing and Networking (MobiCom'01)*, Rome, Italy, July 2001, pp. 70–84.

[34] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang. PEAS: A robust energy conserving protocol for long-lived sensor networks. In *Proceedings of the International Conference on Distributed Computing Systems (ICDCS'03)*, Providence, RI, May 2003, pp. 28–37.

[35] H. Zhang and J.C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc and Sensor Wireless Networks: An International Journal*, **1**(1–2) (January 2005), 89–123.

[36] Gang Zhou, Tian He, Sudha Krishnamurthy, and John A. Stankovic. Models and solutions for radio irregularity in wireless sensor networks. *ACM Transactions on Sensor Networks*, **2**(2) (May 2006), 221–262.

[37] Y. Zou and K. Chakrabarty. Sensor deployment and target localization in distributed sensor networks. *ACM Transactions on Embedded Computing Systems*, **3**(1) (February 2004), 61–91.

[38] Yi Zou and Krishnendu Chakrabarty. A distributed coverage- and connectivity-centric technique for selecting active nodes in wireless sensor networks. *IEEE Transactions on Computers*, **54**(8) (August 2005), 978–991.

[39] M. Zuniga and B. Krishnamachari. An analysis of unreliability and asymmetry in low-power wireless links. *ACM Transactions on Sensor Networks*, **3**(3) (June 2007), 1–34.