Optimized Broadcast Protocol for Sensor Networks

Arjan Durresi, *Senior Member*, *IEEE*, Vamsi K. Paruchuri, *Student Member*, *IEEE*, S. Sitharama Iyengar, *Fellow*, *IEEE*, and Rajgopal Kannan

Abstract—Sensor networks usually operate under very severe energy restrictions. Therefore, sensor communications should consume the minimum possible amount of energy. While broadcasting is a very energy-expensive protocol, it is also widely used as a building block for a variety of other network layer protocols. Therefore, reducing the energy consumption by optimizing broadcasting is a major improvement in sensor networking. In this paper, we propose an optimized Broadcast Protocol for Sensor networks (BPS). The major novelty of BPS is its adaptive-geometric approach that enables considerable reduction of retransmissions by maximizing each hop length. BPS adapts itself and gets the best out of existing radio conditions. In BPS, nodes do not need any neighborhood information, which leads to low communication and memory overhead. We analyze the worst-case scenario for BPS and show that the number of transmissions in such a scenario is a constant multiple of those required in the ideal case. Our simulation results show that BPS is very scalable with respect to network density. BPS is also resilient to transmission errors.

Index Terms—Broadcasting, flooding protocols, wireless sensor networks.

1 Introduction

Recent advances in wireless communications and micro-electro-mechanical systems have enabled the development of extremely small, low-cost sensors that possess sensing, signal processing, and wireless communication capabilities. These sensors can be deployed at a much lower cost than that of traditional wired sensor systems. An ad hoc wireless network of large numbers of such inexpensive but less reliable and accurate sensors can be used in a wide variety of commercial and military applications such as target tracking, security, environment monitoring, and system control.

In wireless sensor networks, it is critically important to save energy. Battery-power is typically a scarce and expensive resource in wireless sensor devices. Hence, energy efficient communication techniques are essential for increasing the lifetime of such wireless networks.

In broadcasting, one node sends a packet to all other nodes in the network. Many applications, as well as various unicast routing protocols, use broadcasting or its variations. Applications of broadcasting include location discovery, establishing routes, and querying. Broadcasting can also be used to discover multiple paths between a given pair of nodes. Many routing protocols propose using localized broadcasting for route maintenance.

A straightforward approach for broadcasting is *flooding*, in which each node is required to retransmit packets when received for the first time. Flooding generates many redundant transmissions, which may cause a serious *broadcast storm problem* [4]. Given the restriction on energy and bandwidth of sensor networks, minimizing the broadcasting overhead is a high priority in protocol design for sensor networks.

Recently, a number of research groups have proposed efficient broadcasting protocols. Centralized broadcasting approaches are presented in [7], [8], [9]. Solutions in [12], [13], [14], [15], [16], [18] utilize neighborhood information to reduce redundant messages in a Mobile Ad Hoc Network. Solutions in [28], [29], [30] deal with data dissemination in sensor networks. The SPIN [28] and Directed Diffusion [30] protocols use *application-specific data-naming* and *routing* to reduce redundant transmissions. In [29], protocols are presented that achieve nonuniform information dissemination through which nodes are updated with *varied accuracy or precision* of information depending upon their requirements. But, it is desirable to have an efficient and application independent broadcasting protocol which can be used as a building block for applications and protocols whenever there is a need for a packet to be sent to all other nodes. Moreover, the data dissemination protocols in [28], [29], [30] are reduced to flooding in the absence of information about sensor interest. In this paper, we propose an optimized Broadcast Protocol for Sensor Networks (BPS), based on an adaptive-geometric approach that enables a significant reduction on retransmissions and communication overhead.

The goals used in the BPS design were:

- *Scalability*: Scalability for sensor networks is a critical factor. For large-scale networks, scalability can be achieved by using distributed protocols. A protocol should be based on localized interactions and should not need global knowledge such as current network topology. Also, a protocol's performance should not deteriorate with the increase of network densities.

- *The authors are with the Department of Computer Science, 298 Coates Hall, Louisiana State University, Baton Rouge, LA 70803. E-mail: {durresi, paruchuri, iyengar, rkannan}@csc.lsu.edu.*

Manuscript received 7 June 2004; revised 10 Dec. 2004; accepted 3 Feb. 2005; published online 15 June 2005.
For information on obtaining reprints of this article, please send e-mail to: tv@computer.org, and reference IEEECS Log Number TC-0190-0604.

- *Energy efficiency*: Because wireless sensors have a limited supply of energy, energy aware communications and computations are essential to wireless sensor networks. For a broadcast protocol, this goal means reducing the number of transmissions.
- *Memory*: Sensors are memory constrained. Therefore, the storage requirements of a protocol should be minimal. Thus, broadcast protocols that do not require neighborhood information are more appropriate for sensor networks.
- *Computation*: Sensors have limited computing power and, therefore, may not be able to run sophisticated network protocols.

BPS minimizes the number of transmissions needed for broadcasting by doing selective forwarding, where only a few self-selected nodes in the network retransmit packets. We assume that each node knows its location, which also is a requirement for various other routing protocols, sensing, target tracking, and other applications. Various techniques like GPS [2], Time Difference of Arrival [25], Angle of Arrival [26], and Received Signal Strength Indicator [24] have been proposed to enable a node to discern its relative location. Recently, a range-free cost-effective solution [23] has been proposed for the same problem. To minimize the number of transmissions, BPS tries to maximize each hop length. That is, after one node has transmitted, it would be desirable for the next transmitting node to be the most distant possible. In an ideal network, the upper bound of the hop length is the nodes' communication ranges. But, in wireless sensor networks in real conditions, as shown in several studies [32], [33], [34], there is no clear correlation between packet delivery and distance among nodes for a significant portion of the communication range. So, there is a *gray area* within the communication range of nodes in which receivers experience significant variable and unstable reception over time. That is, nodes that are geographically far away from the source *may* get better connectivity than nodes that are geographically closer. This can be explained with multipath and fading effects of low-power radio devices commonly used in sensor networks. BPS deals with this situation by having nodes use a self-selection mechanism to decide which one will transmit next. So, after one node has transmitted, the next retransmitting node will be self-selected by being the most distant (bounded by communication range) one from the source that was able to receive (construct) the packet without errors. This is achieved by using a waiting mechanism that imposes on nodes a waiting time inversely proportional to the distance from the source.

The key advantages of our protocol are:

1. BPS minimizes the number of unnecessary transmissions by maximizing the hop length and, consequently, it outperforms other variations of flooding.
2. In BPS, the radio channel quality is measured implicitly for each transmission, therefore the protocol adapts itself and tries to get the best out of the existing radio channel conditions.
3. With BPS, the number of transmissions required decreases as the density of the network increases.
4. In BPS, a node does not need to know locations/addresses of all its neighbors and, hence, BPS does not impose any bandwidth overhead in terms of *hello* messages and has no memory overhead.
5. BPS performs well, even in very large networks.
6. BPS is robust to transmission errors and impairments, as shown by our simulation results.
7. BPS is able to reach a large fraction of nodes even when the nodes are moving at high speeds.

Because of the above-mentioned advantages, BPS is very well suited for sensor networks that operate even in adverse conditions. BPS, unlike conventional broadcast protocol, uses an adaptive-geometric approach, which makes it very scalable and less dependent on network topology compared to other broadcasting protocols.

The rest of this paper is organized as follows: Section 2 discusses related work, Section 3 introduces our protocol, Section 4 presents the analytic performance bounds of BPS, Section 5 presents the simulation results of BPS, and Section 6 concludes.

2 Related Work

Network-wide broadcasting is an essential feature for wireless networks. The simplest method for broadcasting is flooding. Its advantages are its simplicity and reachability. However, flooding generates abundant retransmissions, resulting in battery energy and bandwidth waste. Also, the retransmissions of close nodes are likely to happen at the same time. As a result, flooding quickly leads to message collisions and channel contention. This is known as the *broadcast storm problem* [4].

Broadcasting has been extensively studied for multihop networks. *Optimal* solutions to compute Minimum Connected Domination Set (MCDS) [9] were obtained for the case when each node knows the topology of the entire network (*centralized* broadcasting). The broadcasting protocol introduced in [7] completes the broadcast of a message in $\mathrm{O}(\mathrm{D}\log^2\mathrm{n})$ steps, where "D" is the diameter of the network and "n" is the number of nodes in the network. From the result proven in [8], this protocol is optimal for networks with constant diameter. For networks with a larger diameter, a protocol by Gaber and Mansour [8] completes the broadcast within $\mathrm{O}(\mathrm{D} + \log^5\mathrm{n})$ time slots and it is optimal for networks with $\mathrm{D} \in \Omega(\log^5\mathrm{n})$. These solutions are *deterministic* and guarantee a bounded delay on message delivery, but the requirement that each node must know the entire network topology is a strong condition, impractical to maintain in wireless sensor networks.

Several broadcasting protocols that do not require knowledge of the entire network topology have been proposed. In a counter-based solution [4], a node does not retransmit if it overhears the same message from its neighbors for more than a prefixed number of times and, in a distance-based scheme [4], a node discards its retransmission if it overhears a neighbor within a distance *threshold* retransmitting the same message.

The Source-Based Algorithm [14], Dominant Pruning [12], Multipoint Relaying [16], Ad Hoc Broadcast Protocol [15], and Lightweight and Efficient Network-Wide Broadcast

Protocol [18] utilize two-hop neighbor knowledge to reduce the number of transmissions.

A good classification and comparison of most of the proposed protocols is presented in [20]. It is also concluded that Scalable Broadcast Algorithm (SBA) [14] and Ad Hoc Broadcast Protocol (AHBP) [15] perform very well as the number of nodes in the network is increased. Both these techniques are based on two-hop neighbor knowledge.

The Scalable Broadcast Algorithm [14] requires that all nodes have knowledge of their neighbors within a two-hop radius. This neighbor knowledge, coupled with the identity of the node from which a packet is received, allows a receiving node to determine if it would reach additional nodes by rebroadcasting. Two-hop neighbor knowledge is achievable via periodic *hello messages*; each *hello message* contains the node's identifier and the list of known neighbors. After a node receives a *hello message* from all its neighbors, it has two-hop topology information centered at itself.

AHBP [15] also requires that all nodes have knowledge of their neighbors within a two-hop radius. In AHBP, only nodes that are designated as a Broadcast Relay Gateway (BRG) within a broadcast packet header are allowed to rebroadcast the packet. BRGs are proactively chosen from each upstream sender, which is a BRG itself. A BRG selects a set of 1-hop neighbors that most efficiently reach all nodes within the two-hop neighborhood as subsequent BRGs. Location Aided Broadcast [21] presents three location-aided broadcast protocols to improve communication overhead and shortcomings of various protocols are also summarized.

In self-pruning protocols [14], [19], [13], each node makes its local decision on forwarding status: forwarding or nonforwarding. Dai and Wu [39] compare the performance of various broadcast protocols for ad hoc networks based on self-pruning. Through rigorous simulations, they show that self-pruning helps in achieving high reliability and delivery ratio while, at the same time, keeping the number of retransmissions low. For sensor networks, which are inherently very memory and energy constrained and because of high deployment densities, protocols based on self-pruning might not be appropriate. Self-pruning requires knowledge of at least two-hop neighbors. Sensors being very memory constrained, storing two-hop neighbor information might be prohibitive. Sensor nodes are highly energy constrained. Self-pruning needs periodic hello messages to keep up-to-date neighbor information that might again lead to significant energy consumption.

The drawback of the above Neighbor Knowledge solutions is the need to store two-hop neighborhood information at each node. In large scale sensor networks, especially with high densities, this might impose very high memory overhead. For instance, at a modest density of 20 nodes per R X R region (R being communication range), a node, on average, has over 250 two-hop neighbors and, even if 10 bytes of data corresponding to each neighbor is stored, the total data is over 2.5 kB. This is over 60 percent of free memory left in a sensor node [27]. Also, keeping the neighbor information current involves additional communication overhead.

In Gossip-based routing [3], a node probabilistically forwards a packet so as to control the spreading of the packet through the network; the probability typically being around 0.65. Though this simple mechanism reduces the number of redundant transmissions, there is still a lot of room for improvement.

Several data dissemination protocols [28], [29], [30] have been proposed for sensor networks to disseminate data to interested sensors rather than all sensors. A broadcast protocol is presented in [31] for regular grid-like sensor networks.

In this paper, we propose a new protocol BPS, based on an adaptive-geometric approach. BPS uses geometric calculations in setting *strategic locations* for the next transmitting node, aiming for a maximal hop size. In this point, BPS shares some characteristics with other position-based routing algorithms [40]. But, BPS differs from them [40] by the way it adapts itself to radio propagation conditions. In BPS, the next transmitting node is *self-selected* based on instantaneous channel conditions as the closest one to *strategic locations*. In other algorithms [40], nodes share information and the next transmitting one is selected by the sender, which leads to more overhead and possible inaccurate channel evaluation due to asymmetric channel characteristics. The adaptability makes BPS resilient to transmission errors and radio propagation impairments. BPS needs minimal neighborhood information as neither the neighboring node addresses nor their locations are needed. This eliminates any communication overhead such as *hello messages*. Another property of BPS, as illustrated through simulations, is that the number of retransmitting nodes gradually decreases as the number of nodes in the network increases. BPS is also able to deliver broadcast packets to a large fraction of nodes even in highly mobile environments.

3 Protocol

In this section, we present the optimized Broadcast Protocol for Sensors Networks (BPS). The intuition behind our protocol is that, in order to broadcast a packet over a network, there is no need for all nodes to transmit/retransmit the message. Instead, the goal can be achieved by allowing only a few strategically selected nodes to retransmit the message by trying to maximize the hop length. We assume that nodes know their location. We also assume that nodes have error detection/correction capabilities and nodes will be able to retransmit a packet only if its received/constructed without errors. The strategy to select such nodes was inspired by the Covering Problem, presented in Section 3.1.

3.1 Covering Problem

The Covering Problem can be stated as follows: "What is the minimum number of circles required to completely cover a given two-dimensional space?" Kershner [1] showed that no arrangement of circles could cover the plane more efficiently than the hexagonal lattice arrangement shown in Fig. 1. Initially, the whole space is covered with regular hexagons, with sides R, and, then, circles are drawn to circumscribe them.

This problem can be modified as follows: "What is the minimum number of circles of radius R required to entirely

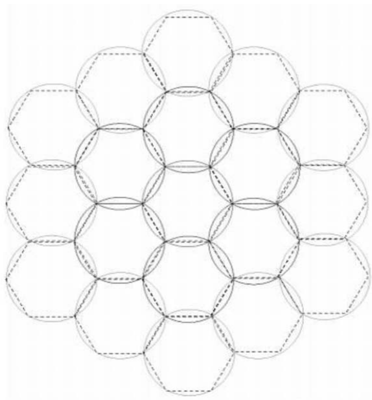

Fig. 1. Covering a plane with circles in an efficient way.

cover a two-dimensional space with the condition that the center of each circle being placed lies on the circumference of at least one other circle?" If R is the node's communication range, then this problem would be that of covering a given area with radio signal. We used this idea to develop our protocol, described in Section 3.2.

3.2 Algorithm

First, we describe the algorithm in ideal conditions. The area to be covered with radio signal is portioned into hexagons. The communication range of nodes determines the hexagons' length of sides. The Source S is at the center of one of the hexagons. In an ideal network, all other transmission nodes are at the vertexes of the hexagons, as shown in Fig. 2. We will call the vertices of hexagons *strategic locations*. The broadcast packets are propagated along the sides of the hexagons (except the first round of transmissions). Any node located inside a hexagon is reachable from at least one of the vertex nodes of the hexagon.

In real conditions, of course, it is impractical to assume that nodes are located at the hexagons' vertices. Thus, if the neighbor nodes are not in the optimal strategy locations, the coverage figure will be *distorted*; moreover, the *distortion* effect may propagate. A simple solution is to select the nearest node to the supposed vertex.

It should also be observed that a node could receive a packet more than once—from different directions and from different nodes, each node specifying a different optimal strategic location (because of distortion). This may cause two nodes very close to each other to retransmit. We propose avoiding these transmissions by having a node keep track of its distance d_m to the nearest node that has retransmitted the packet and having a node retransmit only when its distance to the nearest transmitting node is greater than a threshold Th. The choice of threshold Th is discussed in Section 4.3.

Each broadcast packet contains two location fields, L_1 and L_2, in its header. Whenever a node transmits a broadcast packet, it sets L_1 to the location of the node from which it received the packet and sets L_2 to its own location.

The BPS is as follows:

The Source Node S sets both L_1 and L_2 to its location (S_X, S_Y) and transmits the packet.

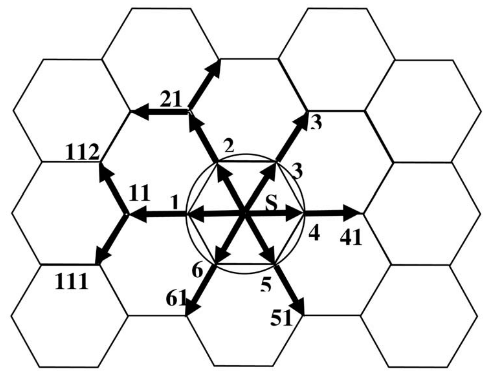

Fig. 2. BPS for an ideal case.

1. Upon the reception of a broadcast packet, a node M first determines if the packet can be discarded. A packet can be discarded under either of the following conditions:
   - If the node has transmitted the packet earlier.
   - If a node which is very close has already transmitted this packet, i.e., if $d_m < Th$.
2. If the packet is not discarded, M determines if the received packet comes directly from the broadcast Source S.
   - If yes, M finds the nearest vertex V (for example, node 1 in Fig. 2) of a hexagon with (S_X, S_Y) as its center coordinates and with $(S_X + R, S_Y)$ as one of its vertexes. It computes its distance l from V and then delays the packet rebroadcast by a delay d given by $d = l/R$.
   - Else, if M hasn't received the packet directly from the source S, but from some other node K, then M selects the nearest strategic location (hexagon vertex). The packet transmission is delayed by $d = l/20 * R$.
3. After the delay d elapses, M again determines if it has received the same packet again and if the packet can be discarded (for the same reasons mentioned above). Thus, delaying enables the selection of a node that successfully received the packet and is the closest to the corresponding strategic location. In the case where the packet cannot be discarded, M updates L_1 to the location of the node from which it received the packet and L_2 to its location, sets d_m to zero, and transmits.

BPS minimizes the number of transmissions by maximizing each hop length. In ideal conditions, the hop length is equal to the communication range of nodes. But, in real conditions, as shown in several studies [32], [33], [34], [35], there is no clear correlation between the packet delivery and the distance among nodes and there is a significant *gray area* within the communication range of nodes where receivers experience variable and unstable reception over time. Even in these conditions, BPS tries to maximize the hop length by selecting among the nodes that correctly received the packet this time (even in the *gray area*) the one that is the closest to its corresponding strategic location (and more distant from the previous transmitting node). This is achieved by having nodes that correctly received self-*delay* the retransmission, where this delay is proportional to the corresponding strategic location. The extra transmissions could not be avoided totally. For example, in very particular propagation conditions, after the most distant available node has transmitted, a less distant node could not listen to this transmission and retransmits itself after its self-delay expires. The chances of these extra transmissions can be reduced by increasing Th (see Sections 4.3 and 5.1).

Choosing low delay values decreases the time needed to broadcast a message all over the network, while high delay values help reduce redundant transmissions in instances where two nodes are of about same distance from the strategic location. The delay function we used causes a packet to be delayed a maximum of 50 ms per retransmission, though, typically, this value lies around 10 ms. In dense networks, the delay values are much less than 10 ms.

The computational complexity of BPS is negligible; when compared to flooding, the major additional computation is finding the node's distance to the nearest optimal point. The bandwidth overhead consists of just a few header fields in the broadcast packet, used to carry location information of two nodes, which is not significant.

TABLE 1
Number of Transmissions Required to Cover (a) a Circular Area or (b) a Rectangular Area in an Ideal Case

Radius of Circular region	Number of transmissions
2R	13
3R	25
4R	43
5R	61
6R	91
7R	127
8R	169

(a)

Size of the rectangular region	Number of Transmissions
3R X 3R	9
4R X 4R	13
5R X 5R	17
6R X 6R	27
8R X 8R	43
10R X 10R	75
4R X 6R	19
6R X 8R	37
8R X 10R	55

(b)

4 Performance Bounds of BPS

In this section, we obtain the analytical bounds on the performance of BPS. The best-case performance of BSP is equivalent to the ideal case. We show that the worst-case performance of BSP is bounded by a constant multiple of the number of transmissions required in an ideal case. This constant is a multiple of Th, which is a system parameter, as described in Section 3. Later in the section, we present the trade-offs involved in selecting Th.

4.1 Best-Case Performance Bound

In this section, we present the number of transmissions required to cover the whole area assuming that we have ideal conditions. For this purpose, we divide the network area into hexagons where, in each vertex, there is a node that retransmits the packet.

Let N_H be the number of hexagons required to cover the entire network of area A. Each regular hexagon's side length is R and area is $3\sqrt{3}R^2/2$. When the area of the network is large compared to the area of one hexagon, N_H can be approximated as:

$$N_H \approx \frac{A}{3\sqrt{3}R^2/2} \quad \text{when } A >> \Pi \mathrm{R}^2. \tag{1}$$

Additional hexagons might be needed to cover the gaps at the boundaries, but this number will be small for very large networks. Also, the network topology will have an effect on this number. But, for regular large networks, (1) provides a very good approximation.

In the ideal case, one transmission occurs at each vertex. Also, each vertex of a hexagon also belongs to two other hexagons. Thus, when the area of the network is large compared to the area of one circle, N_T total number of transmissions can be approximated as:

$$N_T \approx \frac{2 * A}{3\sqrt{3}R^2/2} \quad \text{when A} >> \Pi \mathrm{R}^2. \tag{2}$$

The efficiency is defined as the ratio of the area of the network to the total areas that each broadcast message has covered:

$$Efficiency = \frac{A}{2 * N_c \pi R^2} = 0.413, \tag{3}$$

where N_c is the number of circles in the covering problem, equal to the number of hexagons.

From the above equation, it can be observed that a node receives, on average, 2.4(= 1/Efficiency) messages per node.[1] Also, the above expression shows that the efficiency does not depend on the total number of transmitting nodes. Unlike the previous broadcast protocols that either select the retransmitting nodes with the help of neighbor knowledge or probabilistically, BPS selects the retransmitting nodes based on geometric and link quality criteria. This makes BPS functionality less dependent of the network topology and, hence, this solution is scalable as the number of nodes increases in the region.

The number of transmissions required to cover small circular and rectangular regions in the ideal case scenario are presented in Table 1a and Table 1b, respectively. The number of transmissions required in the ideal case presents a lower bound on the number of transmissions required. As the density of the network increases, the number of transmissions required approaches the lower bound.

1. It should be noted that, because of the broadcast communication nature of sensor networks, each node will receive a message at least twice—once from the neighbor it received the message from and the second from the neighbor it sent the message to.

4.2 Worst-Case Performance Bound

To derive the worst-case performance bound for BPS, we present the worst-case scenario in which a maximum number of transmissions occurs. First, it should be noted that the minimum distance between any two transmitting nodes is controlled by Th, as defined in Section 3. Thus, we claim that, when every transmitting node is at a distance of $Th * \mathrm{R}$ from some node that has transmitted, such a scenario would result in the maximum number of transmissions. Again, this scenario is no different from the ideal case scenario as shown in Fig. 2, except that the transmission range of each node is Th instead of R.

Now, we compute the worst-case bound on the performance of BPS. First, we observe that the number of transmissions needed to cover an area is inversely proportional to the area one single transmission can cover (as explained by (3)). Let n_{ideal} and n_{worst} be the number of transmissions in the ideal case and worst-case scenarios, respectively. Then, it should be observed, from the above argument, that

$$\frac{n_{ideal}}{n_{worst}} = \frac{(Th * R)^2}{R^2} \Rightarrow n_{worst} = n_{ideal}/Th^2. \quad (4)$$

From (4), we can see that the number of transmissions in the worst case is upper bounded by a constant multiple of number of transmissions needed in the ideal case. The constant is determined by Th. In the following section, we discuss the aspects governing the value of Th.

4.3 Role of Threshold Th

The purpose of having the threshold Th is to prevent two nodes that are very close to each other from transmitting, thus reducing the redundancy. The key factors affecting Th are the number of transmissions and the delivery ratio.

Number of transmissions: As Th increases, the number of transmissions decreases. This happens because, when Th increases, the minimum distance between any two transmitting nodes increases. This in turn implies that additional area covered increases and, hence, the number of transmissions needed for covering the entire network decreases.

Delivery Ratio: The Delivery Ratio is the percentage of nodes that received the broadcast. The higher the number of transmissions, the higher the redundancy is and, hence, the higher the probability that a node receives broadcast is. Therefore, for higher delivery ratios, lower Th is preferred.

To elaborate, consider Fig. 3. For simplicity, the transmission range is considered as unity. For a given Th, the additional area covered due to a transmission by a neighbor of S is at least $\Delta_{\mathrm{ILI'L'}}$, where the area of ILI′L′ is:

$$\Delta_{ILI'L'} = \pi - 2 * \Delta_{JILI'} = \pi - 2\theta + Th * \sin\theta$$
$$\theta = \cos^{-1}(Th/2).$$

Now, for higher values of Th, $\Delta_{\mathrm{ILI'L'}}$ is higher and, hence, fewer transmissions are needed to cover the region. But, at the same time, if Th is high, the number of potential neighbors that could retransmit the message is less. To illustrate, consider the shaded region IPNP′I′L. At high values of Th, this area is small and, hence, the probability that some node exists in this area is also low. Thus, at high

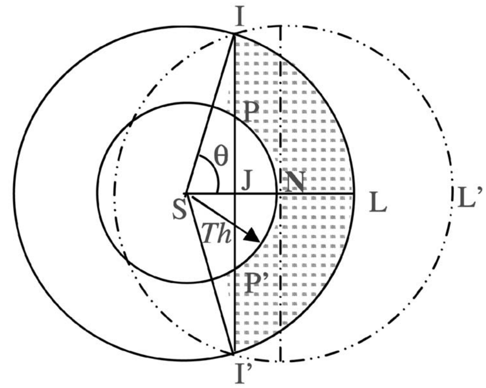

Fig. 3. A scenario illustrating the effect of Th.

Th values there might not be any transmission corresponding to the strategic location L. This might result in some nodes not receiving the broadcast. In Section 5.1, these trade-offs are illustrated using simulation results.

5 Experimental Results

To evaluate the performance of BPS, we have used SensSim [37], [38], a network simulator developed in the Sensor Lab at LSU and built on the OMNeT++ simulation environment [36]. SensSim is more scalable and user friendly than other simulators such as ns-2 [37], [38]. We compared BPS with *flooding*. We also compared our protocol with the Ad Hoc Broadcast Protocol (AHBP) [15] as AHBP is one of the protocols (SBA [14] being the other) that approximates MCDS fairly [20]. A wireless network of different physical areas and different shapes having a different number of nodes was simulated. To be more specific, circular regions of radius varying from R to 10R and rectangular/square regions of size varying from 3R X 3R to 10R X 10R have been simulated, where R is the communication range of each node, which was set to 300m in all our simulations.

The nodes were uniformly distributed all over the region, with the density varying from four nodes per R X R region to 100 nodes per R X R region. Every simulation is repeated until the 95 percent confidence intervals of all average results are within ± 5 percent.

The simulations are aimed at studying the performance of BPS in networks of different sizes and densities. Initially, we studied the effect of different threshold values on the performance of BPS. Then, we concentrated on the algorithm efficiency by studying the performance of BPS in static networks and also in highly mobile networks. Last, we studied the performance of BPS in networks where the coverage area of a node is not circular. The simulation results under each network study are presented in the subsections below.

5.1 Effect of Threshold Th

The purpose of this experiment was to evaluate the effect of different threshold values on the performance of BPS. Figs. 4 and 5 show the simulation results for threshold values of 0.35*R, 0.4*R, and 0.45*R. Apart from the number of transmissions in each case, the delivery ratio in percentage for each case is indicated at each data point. *Delivery Ratio* is

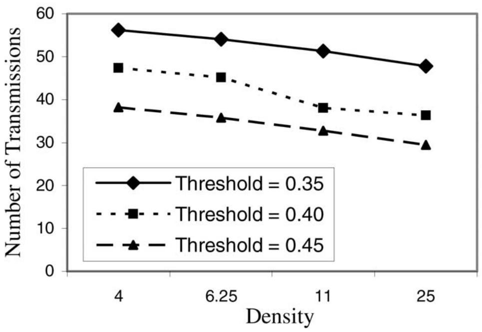

Fig. 4. Effect of Th on the performance of BSP. Network size = 6R X 6R.

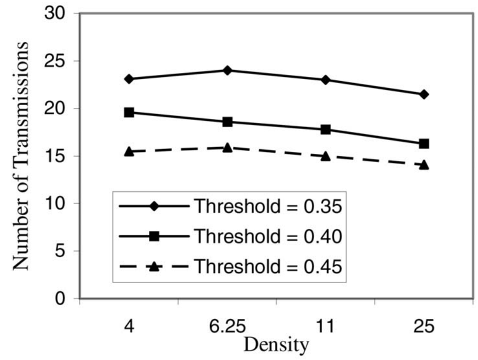

Fig. 5. Effect of Th on the performance of BPS. Network size is 4R X 4R.

the ratio of the average number of nodes that receive the message to the total number of nodes in the network. Figs. 4 and 5 correspond to a network size of 6R X 6R and 4R X 4R, respectively.

For a threshold value of $Th = 0.35 * \mathrm{R}$, a delivery ratio of around 98 percent is achieved and, for $Th = 0.4 * \mathrm{R}$, the delivery ratio is close to 95 percent. But, for $Th = 0.45 * \mathrm{R}$, the delivery ratio falls to around 90 percent. This is understandable because, with the increase in threshold value, the number of retransmitting nodes decreases.

For all further simulations, we use a threshold value of $Th = 0.4 * \mathrm{R}$ and, for each simulation case, we present the minimum and maximum delivery ratio instead of presenting the delivery ratio for each data point.

5.2 BPS Efficiency

In this section, we evaluate the performance of BPS in networks of different sizes and different densities. We include a "best-case" bound provided by the simulation results in *ideal case scenarios*. It is impossible for any algorithm to perform better than the performance in the *ideal case scenario* and unlikely to perform worse than simple flooding. Thus, these two bounds provide a useful spectrum to gauge the performance of our protocol. BPS, when compared to flooding, uses up to 65 to 90 percent fewer messages, depending on the density of the network. For this study, we varied the network size from 3R X 3R to 10R X 10R, while keeping the communication range of each node fixed to 300m. We also varied the network density from four-nodes/R X R region to 100-nodes/R X R region.

First, fixing the density in the region, we simulated the number of transmissions needed to cover a square/rectangular region completely. The coverage figure gets *distorted* considerably and, in most of the cases, no node exists at the strategic location.

In order to quantify the *distortion*, we define *Degree of Distortion* as follows:

Degree of Distortion (DoD) is defined as the average distance between the nearest node that would retransmit the packet and the strategic location normalized to the communication range of the nodes, i.e.,

$$DoD = \frac{1}{|S|} \sum_{\forall i \in S} \frac{d(L_i, L_P)}{R},$$

where S is the set of nodes that have transmitted the packet and $|S|$ is the size of S.

$d(L_i, L_P)$ is the distance between the strategic point L_P and L_i location of node i ($i \in S$) nearest to L_P. R is the sensor's transmission range.

The degree of distortion, as expected, is high in networks with low density as there might not be any nodes close to strategic locations. As the density increases, the distortion decreases because the probability of finding a node closer to the strategic location increases. Table 2 shows DoD for an 8R X 8R network for different densities. Similar DoD values have been observed for other network sizes. This shows that DoD depends only on the density of the network.

Fig. 6 shows two such cases—one for 4R X 4R and another for 6R X 4R regions, both with a density of four nodes per R X R region.

Fig. 7 is a plot between the number of transmissions required to cover entire region for varying densities and for different areas of the region. Network areas up to 10R X 10R have been considered. Fig. 8 presents the results in a different perspective. It gives a plot between the number of transmissions and the density of the network for different network sizes. It can be seen that the number of transmissions required decreases as the number of nodes (density) increases. The number of transmissions at a density of 100 is very near to the number needed in an *Ideal case*. The minimum delivery ratio achieved by BPS was 94.3 percent for the case with network size of 6R X 8R and with a density of 6.25. In all other cases, the delivery ratio was close to 95 percent, with the maximum being 97.3 percent. The results show that the performance of BPS remains very

TABLE 2
Degree of Distortion for Various Densities

Density	4	6.25	16	25
DoD	0.338	0.283	0.168	0.144

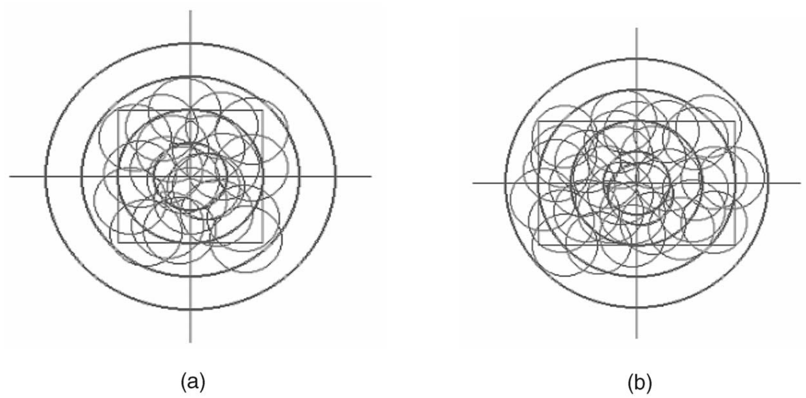

Fig. 6. (a) Example of a distorted figure for a 4R X 4R region with 64 nodes. The number of transmissions is 21. (b) Example of a distorted figure for a 6R X 4R region with 96 nodes. The number of transmissions is 32.

efficient even in large networks; network size does not seem to affect the performance of BPS.

Fig. 9 shows the percentage of nodes in the network retransmitting a broadcast message. The simulations were done in networks of sizes up to 10R X 10R with different node densities. For a given network density, the percentage of retransmissions remains almost a constant for all network sizes. This reflects that BPS performance is not hindered in large networks.

Fig. 10 presents the performance of BPS against flooding. Even at low network sizes, BPS reduces number of transmissions by 66 to 90 percent. Next, we compare BPS with the Ad Hoc Broadcast protocol (AHBP) [14]. Networks of 4R X 4R, 6R X 6R, and 8R X 8R were considered. As shown by Fig. 11, the performance of both BPS and AHBP is very similar, though BPS performs slightly better than AHBP, especially at high network densities. Here, we considered only static networks and, in the next section, we present results for mobile networks where BPS clearly performs much better than AHBP.

5.3 Mobile Networks

This section presents the simulation results of BPS and AHBP in mobile networks. We use the Random Walk mobility model [23] with zero pause time. The range of mean speeds of the nodes is varied from 1 to 20 meters per second. The upper bound corresponds to around 50 miles per hour, which we assume to be a realistic maximum speed of any mobile node.

Fig. 12 presents the effect of mobility on each of the protocols. The simulation is done in a network of 144 nodes and with the network size being 8R X 8R. The performance of BPS remains unaffected as the BPS algorithm uses minimal neighborhood information. But, the performance of AHBP rapidly deteriorates with the increase in speed and its performance is also affected by the *hello interval*.

The two-hop neighbor knowledge-based protocols use *hello messages* to gather the neighborhood information. With a hello interval of t seconds, the two-hop neighbor information (which is obtained through the hello messages of one-hop neighbors) would always be outdated by an average of t seconds. For instance, if $t = 10$ seconds and a node's speed is 36 mph, then the node would have moved up to 100m before its information has been conveyed to one of its two-hop neighbors. Also, once a node gets this information, it is not updated again until 10 sec. Thus, a node could have moved up to 200m before its information is updated at its neighbors.

Also, the average time by which a node's information at a two-hop neighbor is outdated is 15 seconds ($t + (0 + t)/2$), which corresponds to a displacement up to 150m. This shows the intensity of the effect the mobility has on these protocols. Thus, the hello interval t should be very small for efficient performance of two-hop neighbor knowledge-based protocols, which in turn means that the bandwidth overhead due to *hello messages* is very high.

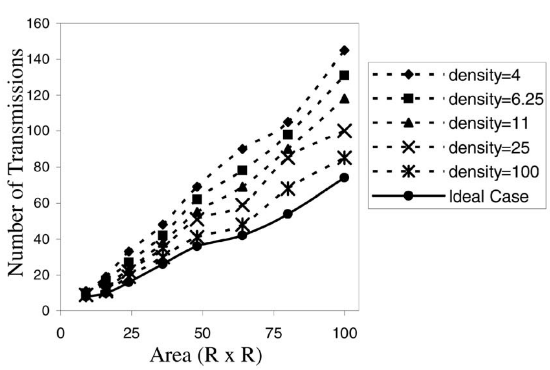

Fig. 7. Number of transmissions required to cover an entire region for different areas.

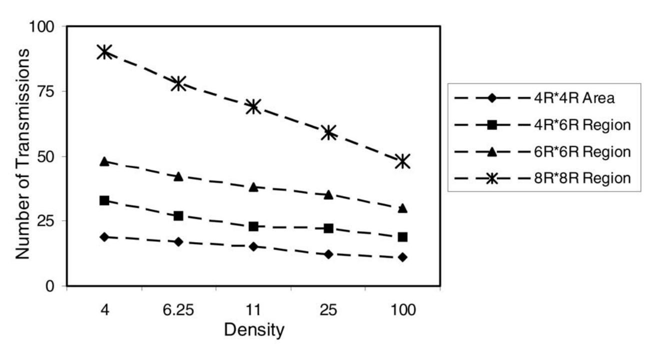

Fig. 8. Number of transmission for varying node densities and for different areas.

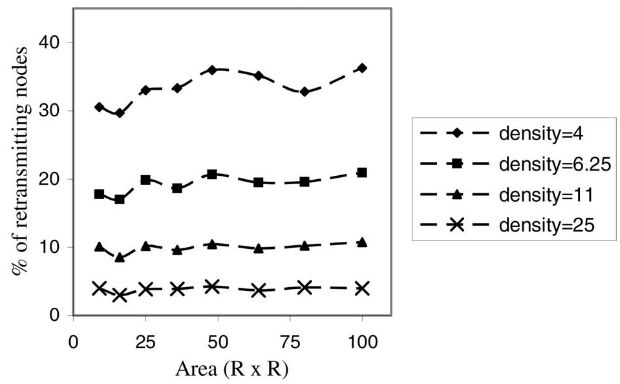

Fig. 9. Percentage of retransmitting nodes for different networks.

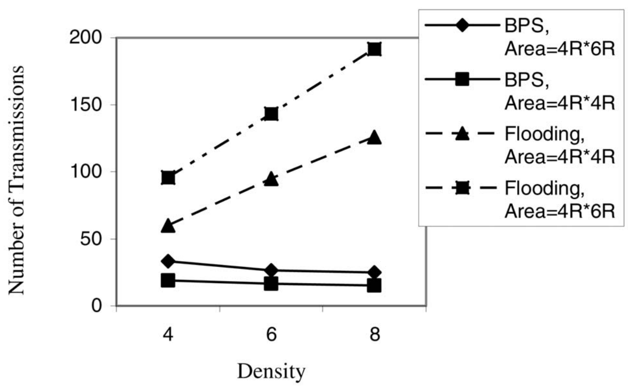

Fig. 10. Performance of BPS and flooding in static networks.

5.4 Effect of Transmission Errors

Sensor networks are characterized by losses due to transmission errors. We simulated the performance of BPS in networks with errors in transmission. Fig 13 compares the performance of BPS and AHBP in a network of size 6R X 6R. A network of 144 nodes has been considered in this case. These simulations were for static networks. Transmission error rates up to 30 percent were simulated and we simulated a uniform transmission error model. We used a uniform transmission error model. While other models can be used, in this case, the model itself is not very important. This is because we are concerned with only one packet transmission over a link and a sensor has to determine if it correctly received that particular packet or not. The error model would be more important in the case of a routing protocol in which data has to be transmitted over a link through several packets and each one of them could meet different conditions.

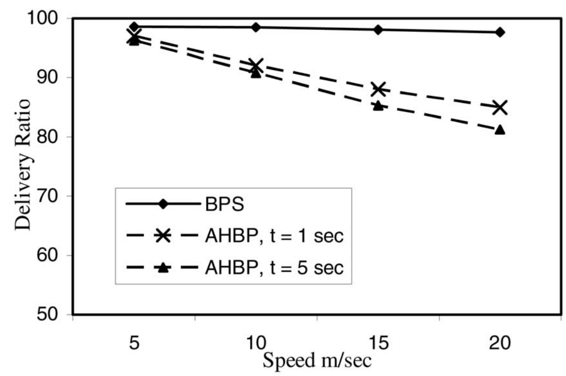

Fig. 12. Effect of mobility on different protocols. Network size is 8R X 8R. Number of nodes = 144.

It can be seen that the performance of BPS degrades gracefully with an increase in transmission errors and BPS was able to achieve a delivery ratio of 84 percent even at a transmission error rate of 30 percent. At the same time, the performance of AHBP degrades rapidly and the delivery ratio is less than 63 percent at an error rate of 30 percent.

The results show the robustness and resilience of BPS. This makes BPS a good choice for wireless networks that operate in adverse conditions. The high delivery ratio of BPS can be attributed to the fact that each node decides on its own whether to retransmit a packet or not and the decision is based on minimal neighborhood information brought by packets themselves. In the presence of transmission errors, the closest node to the strategic location that has received the packet properly will retransmit. Also, one might expect that, at a transmission error rate of 30 percent, on average, around 30 percent of the nodes would not be able to get the packet error free and the delivery ratio should be less than 70 percent. But, it should be noted that most of the nodes in the network receive a packet more than once and from different directions and, hence, the delivery ratio would be significantly better than 70 percent.

In the case of AHBP, each retransmitting node recursively designates some of neighbors as Broadcast Relay Gateways (BRGs) and piggybacks the designated node addresses in the broadcast packet. Thus, if a designated BRG fails to receive the packet error-free, then no other node will be retransmitting instead of this node. Thus, the

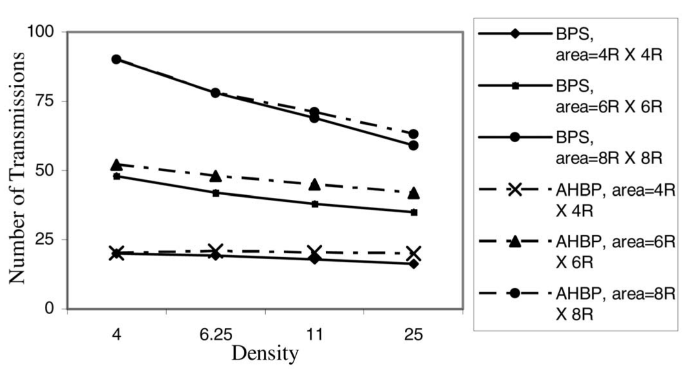

Fig. 11. Performance comparison of BPS and AHBP in static networks.

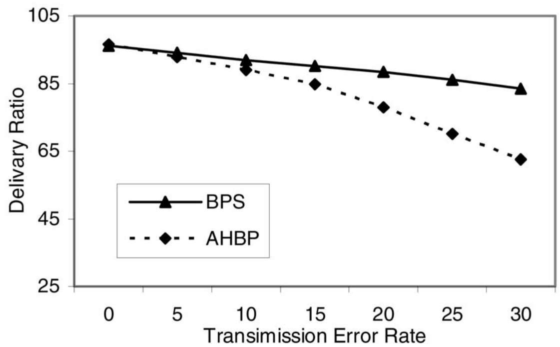

Fig. 13. Performance comparison of BPS and AHBP in the presence of transmission errors. Network size = 6T X R. Number of nodes = 144.

effect of transmission errors is much more profound on AHBP than BPS.

5.5 Effect of Nonuniform Radio Propagation

In this section, we study the performance of BPS in wireless networks where wireless propagation is noncircular. Radio irregularity in sensor networks is investigated in full detail in [35] and nonisotropic radio modes have been proposed. The performance results of BPS with a model very similar to the one proposed in [35] further shows that our algorithm does not assume any correlation between topology and geography.

We use the term *noncircularity* to mean that the range of a node might be different in each direction, the maximum being R, which is the range in an ideal case. The contours of the terrain and obstructions like large buildings contribute in creating such nonuniform radio propagation.

In the simulations, for each node, we generated the coverage area by setting the transmission range in different directions to a random value between $[D * R, R]$, where D is the *Degree of Distortion* and R is the communication range of a node in an ideal scenario. The simulations were done for static networks.

The performance of BPS in case of noncircularity is presented in Fig. 14. Fig. 14 corresponds to a network area of 6R X 6R. It can be observed that the number of transmissions needed grows linearly with the degree of distortion. The delivery ratio in all of the cases was above 94 percent, with the least being around 94.3 percent.

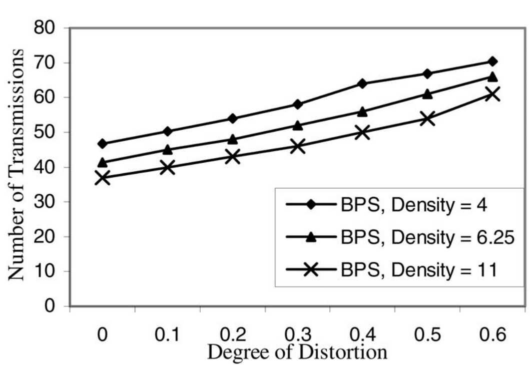

Fig. 14. Effect of nonuniform propagation on BPS. Network size is 1,800 m X 1,800 m.

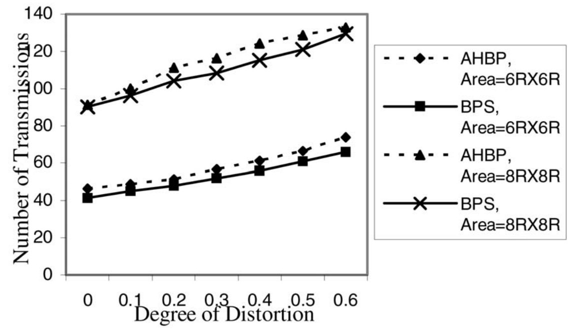

Fig. 15. Performance comparison of BPS and AHBP under nonuniform propagation. Network density = 6.25.

The performance comparison of BPS and AHBP is presented in Fig. 15. The figure is a plot between the number of transmissions and Degree of Distortion for network sizes of 6R X 6R and 8R X 8R and for a network density of 6.25. The performance of both protocols is similar. In both protocols, the number of transmissions increases almost linearly with respect to the *Degree of Distortion*. The effect of mobility is not considered in these simulations.

The purpose of this study was to see the performance of BPS in networks with nonuniform transmission ranges. As shown by Figs. 14 and 15, BPS's performance remains efficient even under such conditions. This can be attributed to the fact that, in BPS, the decision if a node retransmits or not is made locally at each node that receives the packet. Thus, even if a node very close to the strategic location does not get the packet, the reachability is not affected because some other node that received the packet retransmits.

6 Conclusion

Building efficient broadcast protocols for sensor networks is challenging due to the energy, communication, and computation constraints of sensors. In this paper, we proposed a novel optimized Broadcast Protocol for Sensor Networks (BPS).

The adaptive-geometric approach makes BPS very scalable and energy efficient due to the minimum number of transmissions. BPS maximizes each hop length while getting the best out of the existing radio propagation situation. BPS is performed in an asynchronous and distributed manner by each node in the network. In BPS, nodes do not need any neighborhood information, therefore the communication and memory overhead is low. The efficiency of BPS remains very high even in large networks and BPS scales with density. Its efficiency in mobile networks and its robustness even in the presence of transmission errors make it an ideal choice for mobile ad hoc and sensor networks.

References

[1] R. Kershner, "The Number of Circles Covering a Set," *Am. J. Math.*, vol. 61, pp. 665-671, 1939.
[2] G. Dommety and R. Jain, "Potential Networking Applications of Global Positioning Systems (GPS)," Technical Report TR-24, Computer Science Dept., The Ohio State Univ., Apr. 1996.
[3] Z. Haas, J. Halpern, and L. Li, "Gossip Based Ad Hoc Routing," *Proc. IEEE INFOCOM*, June 2002.

[4] S.Y. Ni et al., "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Proc. ACM MOBICOM*, pp. 151-162, Aug. 1999.

[5] J. Li, C. Blake, D.S.J. De Couto, H.I. Lee, and R. Morris, "Capacity of Ad Hoc Wireless Networks," *Proc. Seventh Ann. ACM/IEEE Int'l Conf. Mobile Computing and Networking (MobiCom)*, pp. 61-69, July 2001.

[6] I. Chlamtac and O. Weinstein, "The Wave Expansion Approach to Broadcasting in Multi-Hop Radio Networks," *IEEE Trans. Comm.*, vol. 39, pp. 426-433, Mar. 1991.

[7] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg, "A Lower Bound for Radio Broadcast," *J. Computing Systems Science*, vol. 43, pp. 290-298, Oct. 1991.

[8] I. Gaber and Y. Mansour, "Broadcast in Radio Networks," *Proc. Sixth Ann. ACM-SIAM Symp. Discrete Algorithms*, pp. 577-585, Jan. 1995.

[9] S. Guha and S. Khuller, "Approximation Algorithms for Connected Dominating Sets," *Proc. Fourth Ann. European Symp. Algorithms (ESA)*, pp. 179-193, 1996.

[10] C. Ho, K. Obraczka, G. Tsudik, and K. Viswanath, "Flooding for Reliable Multicast in Multi-Hop Ad Hoc Networks," *Proc. Int'l Workshop Discrete Algorithms and Methods for Mobile Computing and Communication (DIALM)*, pp. 64-71, 1999.

[11] J. Jetcheva, Y. Hu, D. Maltz, and D. Johnson, "A Simple Protocol for Multicast and Broadcast in Mobile Ad Hoc Networks," Internet Draft: draft-ietf-manet-simple-mbcast-01.txt, July 2001.

[12] H. Lim and C. Kim, "Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks," *Proc. ACM Int'l Workshop Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, pp. 61-68, 2000.

[13] J. Wu and F. Dai, "Broadcasting in Ad Hoc Networks Based on Self-Pruning," *Proc. IEEE INFOCOM 2003*, 2003.

[14] W. Peng and X. Lu, "On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks," *Proc. MOBIHOC*, 2000.

[15] W. Peng and X. Lu, "AHBP: An Efficient Broadcast Protocol for Mobile Ad Hoc Networks," *J. Science and Technology-Beijing, China*, 2002.

[16] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint Relaying: An Efficient Technique for Flooding in Mobile Wireless Networks," Technical Report 3898, INRIA-Rapport de Recherche, 2000.

[17] I. Stojmenovic, M. Seddigh, and J. Zunic, "Internal Node Based Broadcasting in Wireless Networks," *Proc. Hawaii Int'l Conf. System Sciences (HICSS)*, 2001.

[18] J. Sucec and I. Marsic, "An Efficient Distributed Network-Wide Broadcast Algorithm for Mobile Ad Hoc Networks," CAIP Technical Report 248, Rutgers Univ., Sept. 2000.

[19] J. Wu and H. Li, "On Calculating Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks," *Proc. Int'l Workshop Discrete Algorithms and Methods for Mobile Computing and Comm. (DIAL-M)*, pp. 7-14, 1999.

[20] B. Williams and T. Camp, "Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks," *Proc. Third ACM Int'l Symp. Mobile Ad Hoc Networking & Computing*, June 2002.

[21] M. Sun and T.H. Lai, "Location Aided Broadcast in Wireless Ad Hoc Network Systems," *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC 2002)*, pp. 597-602, Mar. 2002.

[22] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wireless Comm. & Mobile Computing (WCMC)*, vol. 2, no. 5, pp. 483-502, 2002.

[23] T. He, C. Huang, B.M. Blum, J.A. Stankovic, and T.F. Abdelzaher, "Range-Free Localization Schemes in Large Scale Sensor Networks," *Proc. Ninth Ann. Int'l Conf. Mobile Computing and Networking (MobiCom 2003)*, Sept. 2003.

[24] P. Bahl and V.N. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," *Proc. IEEE INFOCOM '00*, 2000.

[25] A. Savvides, C.C. Han, and M.B. Srivastava, "Dynamic Fine-Grained Localization in Ad-Hoc Networks of Sensors," *Proc. MOBICOM '01*, July 2001.

[26] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) Using AoA," *Proc. INFOCOM '03*, 2003.

[27] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J.D. Tygar, "SPINS: Security Protocols for Sensor Networks," *Wireless Networks J. (WINE)*, Sept. 2002.

[28] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Network," *Proc. Fifth ACM/IEEE Mobicom Conf. (MobiCom '99)*, Aug. 1999.

[29] S. Tilak, A.L. Murphy, and W.B. Heinzelman, "Non-Uniform Information Dissemination for Sensor Networks," *Proc. 11th Int'l Conf. Network Protocols (ICNP '03)*, Nov. 2003.

[30] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *Proc. Sixth Ann. Int'l Conf. Mobile Computing and Networking (MobiCOM '00)*, Aug. 2000.

[31] S. Dolev, T. Herman, and L. Lahiani, "Polygonal Broadcast for Sensor Networks," *Proc. Second IEEE Upstate New York Workshop Sensor Networks*, 2003.

[32] J. Zhao and R. Govindan, "Understanding Packet Delivery Performance Indense Wireless Sensor Networks," *Proc. First Int'l Conf. Embedded Networked Sensor Systems (Sensys 2003)*, pp. 1-13, Nov. 2003.

[33] A. Woo, T. Tong, and D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," *Proc. First Int'l Conf. Embedded Networked Sensor Systems (Sensys 2003)*, pp. 14-27, Nov. 2003.

[34] A. Cerpa, N. Busek, and D. Estrin, "SCALE: A Tool for Simple Connectivity Assessment in Lossy Environments," UCLA CENS-TR #21, 09/2003, http://deerhound.ats.ucla.edu:7777/pls/portal/docs/PAGE/CENS_MAIN/RESOURCES/LARGE_FILES/TR21_SCALE-TECH-REPORT.PDF, 2003.

[35] G. Zhou, T. He, S. Krishnamurthy, and J.A. Stankovic, "Impact of Radio Irregularity on Wireless Sensor Networks," *Proc. Second Int'l Conf. Mobile Systems, Applications, and Services (Mobisys 2004)*, pp. 125-138, 2004.

[36] OMNeT++: Discrete Event Simulation System, http://www.omnetpp.org/, 2005.

[37] S. Basavaraju, "SenSim: A Wireless Sensor Network Simulation Environment," master's thesis, Dept. of Computer Science, Louisiana State Univ., June 2004.

[38] C. Mallanda, "Extensions of SenSim Simulation Platform for Sensor Networks," master's thesis, Dept. of Computer Science, Louisiana State Univ., Dec. 2004.

[39] F. Dai and J. Wu, "Performance Analysis of Broadcast Protocols in Ad Hoc Networks Based on Self-Pruning," *IEEE Trans. Parallel and Distributed Systems*, vol. 15, no. 11, pp. 1027-1040, Nov. 2004.

[40] S. Giordano and I. Stojmenovic, "Position Based Routing Algorithms for Ad Hoc Networks: A Taxonomy," *Ad Hoc Wireless Networking*, X. Cheng, X. Huang, and D.Z. Du, eds., pp. 103-136, Kluwer, 2004.

Arjan Durresi received the BSEE, MS, and PhD degrees (all summa cum laude), all in electronic-telecommunications, in 1986, 1991, and 1993, respectively, and a Diploma of Superior Specialization in telecommunications from La Sapienza University in Rome, Italy, and the Italian Telecommunications Institute in 1991. From 1991 to 1995, he served as a senior software analyst at Telesoft S.p.A, Rome, Italy. From 1995 to 1996, he was a faculty member in the Department of Electronics at the Polytechnic University of Tirana, working on telecommunications. From 1996 to 2003, he was a research scientist in the Department of Computer and Information Science at Ohio State University. In 2003, he joined Louisiana State University, where he is currently an assistant professor in the Department of Computer Science. His current research interests include heterogeneous wireless sensor networks, security, QoS routing protocols, traffic management, grid computing, optical and satellite networks, multimedia networking, performance testing, and bioinformatics. He is an area editor for the *Ad Hoc Networks Journal* (Elsevier) and a guest editor for the *International Journal of Wireless and Mobile Computing* and the *International Journal of Distributed Sensor Networks*. He was the cochair of the First International Workshop on Heterogeneous Wireless Sensor Networks (HWISE 2005), program vice chair of AINA 2004 and ICPADS 2005, and program area chair of AINA 2005. He received the Italian Telecommunication Institute Fellowship and Soros Foundation Fellowships in 1991 and 1996, respectively, and the Lumley Research Award from Ohio State University in 2002. He is a senior member of the IEEE and a member of the IEEE Computer Society.

Vamsi K. Paruchuri received the BTech degree in electronics and communications engineering from Sri Venkateswara University, Tirupati, India, in 2001 and the MS degree in electrical engineering from Ohio State University, Columbus, in 2003. He is currently a PhD candidate in computer science at Louisiana State University, Baton Rouge. His research interests include routing and security protocol design, analysis, and implementation for wireless networks. He is a student member of the IEEE.

S. Sitharama Iyengar (M'88-SM'89-F'95) received the MS degree from the Indian Institute of Science, Bangalore, in 1970 and the PhD degree from Mississippi State University in 1974. He is the chairman and Roy Paul Daniels Chaired Professor of Computer Science at Louisiana State University, Baton Rouge, and is also the Satish Dhawan Chaired Professor at the Indian Institute of Science. His publications include 13 books (authored or coauthored textbooks; Prentice-Hall, CRC Press, IEEE Computer Society Press, John Wiley & Sons, etc.) and more than 280 research papers in refereed journals and conferences. He is the founder and editor-in-chief of the *International Journal of Distributed Sensor Networks*. He has served as a guest editor for the *Journal of Theoretical Computer Science* and the *Journal of Computer and Electrical Engineering*. He has been involved with research in high-performance algorithms, data structures, sensor fusion, data mining, and intelligent systems. Dr. Iyengar was awarded the Distinguished Alumnus Award by the Indian Institute of Science in March 2003. He has served as an associate editor for the IEEE and as a guest editor for the *IEEE Transactions on Knowledge and Data Engineering*, the *IEEE Transactions on Systems, Man, and Cybernetics*, and the *IEEE Transactions on Software Engineering*. He is a fellow of the IEEE, the ACM, and the American Association for the Advancement of Science (AAAS).

Rajgopal Kannan received the BTech degree in computer science and engineering from the Indian Institute of Technology (IIT), Bombay, in 1991 and the PhD degree in computer science from the University of Denver, Denver, Colorado, in 1996. He is currently an assistant professor in the Computer Science Department at Louisiana State University, Baton Rouge. His areas of interest are in wireless sensor networks, data security, optical networks, routing and multicasting protocols, distributed systems, algorithms, and game-theory. He has served on the program committee of ICCCN, MobiDe, etc., and is a reviewer for the US National Science Foundation and several journals and conferences.