Computer Communications 101 (2017) 82–93

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Pseudo geometric broadcast protocols in wireless sensor networks: Design, evaluation, and analysis

Chunchao Liang[a], Sunho Lim[b,*], Manki Min[c], Wei Wang[d]

[a] Department of CS, Eastern Kentucky University, Richmond, KY 40475, USA
[b] T²WISTOR: TTU Wireless Mobile Networking Laboratory, Dept. of CS, Texas Tech University, Lubbock, TX 79409, USA
[c] Department of EECS, South Dakota State University, Brookings, SD 57007, USA
[d] Department of CS, San Diego State University, San Diego, CA 92128, USA

ARTICLE INFO

Article history:
Received 16 November 2015
Revised 17 November 2016
Accepted 18 December 2016
Available online 23 December 2016

Keywords:
Global positioning system
Pseudo geometric broadcast protocol
Performance evaluation
Wireless sensor networks

ABSTRACT

With increasingly popular wireless and low-power sensors/devices, broadcast is one of essential operations to enhance information accessibility and availability. Due to the lack of centralized coordination and limited resources, however, designing an efficient broadcast protocol is challenging in Wireless Sensor Networks (WSNs). Although on-board Global Positioning System (GPS) based broadcast protocols that heavily rely on the geographical information and its accuracy have been proposed, the positional inaccuracy and non-negligible deployment cost may become an issue. Thus, we investigate a pseudo geometric broadcast problem and propose its corresponding protocols, called pseudo geometric broadcast protocols, in resource constrained WSNs: (i) Approximating Neighbor Nodes based Broadcast Protocol (Approx), (ii) Enhanced Ad Hoc Broadcast Protocol (AHBP) with Target Forwarding Nodes (EBP($|N_f|$)), and (iii) Node Distribution Sensitive Broadcast (NDS). The basic idea is that a packet sender approximates the locations of its neighbor nodes and searches a set of forwarding nodes located close to the strategic positions without the support of GPS in a heuristic manner. We develop a customized discrete-event driven simulator using OMNeT++ to conduct our experiments by varying the key simulation parameters, and analyze the performance of broadcast protocols in terms of packet delivery ratio, number of broadcasts, propagation delay, and computational overhead. Extensive simulation results indicate that the proposed broadcast protocols achieve competitive performance and become viable approaches in WSNs.

1. Introduction

The growing presence of WiFi and 4G LTE enables users to pursue seemingly insatiable access to Internet services and information wirelessly. It is predicted that 40.9 billion wirelessly connected devices will be available by 2020, nearly triple the number that exist today [1]. The spread of these devices and hybrid networks is leading to the emergence of an Internet of Things (IoT), where a myriad of multi-scale sensors and heterogeneous devices are seamlessly blended for ubiquitous computing and communication. The prevalence of cloud, social media, and wearable computing and the reduced cost of processing power, storage, and bandwidth are fueling explosive development of IoT applications in major domains (i.e., personal and home, enterprise, utilities, and mobile) [2]. These IoT applications have the potential to create an economic impact of $3.9 trillion to $11.1 trillion annually by 2025 [3]. It is envisioned that wirelessly connected smart and low-power sensors/devices (later in short nodes) will further change life as we live it.

To realize this vision, *broadcast* is one of essential operations to share locally sensed information among each other to enhance information accessibility and availability. Flooding is a simplest broadcast protocol, where each node rebroadcasts any received packet in the network. Since broadcast has the inherent constraints in terms of redundant retransmissions and packet contentions and collisions [4], however, a blind broadcast followed by a series of unconditional forwarding operations is inefficient and even harmful. The redundant retransmissions can also negatively affect the communication performance, such as the network lifetime. Note that when a node intends to transmit a unicast packet, unlike a wired network, all neighbor nodes located within its communication range can overhear the packet, as if it is a broadcast packet [5]. Due to the lack of centralized coordination and limited resources, designing an efficient broadcast protocol is admittedly challenging in Wireless Sensor Networks (WSNs).

* Corresponding author.
E-mail addresses: chunchao.lane@eku.edu (C. Liang), sunho.lim@ttu.edu (S. Lim), manki.min@sdstate.edu (M. Min), wwang@mail.sdsu.edu (W. Wang).

http://dx.doi.org/10.1016/j.comcom.2016.12.012
0140-3664/

In light of this, we investigate a pseudo geometric broadcast problem and propose its corresponding protocols, called *pseudo geometric broadcast protocols*, in WSNs. The definition of pseudo geometric broadcast problem is to select a set of forwarding nodes in a heuristic manner to minimize the number of broadcasts, propagation delay, and computational overhead, but to maximize the packet delivery ratio without the support of an on-board Global Positioning System (GPS) in a pre-deployed network. The number of broadcasts is counted whenever a node broadcasts a newly received packet. If the node received the packet that has been broadcasted before, it does not count but simply drops the packet to avoid a cycle. In this paper, we do not pursue an optimized solution and its corresponding proof, but try to achieve it in best effort. Note that unlike geometric broadcast, where geographical information (e.g., node location) and its accuracy are important, the pseudo geometric broadcast roughly estimates the location without the support of GPS. Since GPSes often suffer from the positioning inaccuracy, non-negligible deployment cost and energy consumption, and intermittent unavailability of GPS signal in an indoor environment, they are limited to be deployed in resource-constrained WSNs. For example, a NAVSTAR GPS has about 50 to 100 (m) error bounds [6], which can negatively affect the communication performance. Thus, the pseudo geometric broadcast that does not rely on GPS is essential. Note that the pseudo geometric broadcast problem in this paper is different from the network coverage problem [7].

The basic idea of the proposed protocols is that a packet sender approximates the location of its neighbor nodes and searches a set of forwarding candidate nodes located close to the boundary of its communication range. The sender further selects the forwarding candidate nodes that are located at the closest to the strategic positions based on a virtual hexagon-based coverage. The proposed approach is different from prior geometric broadcast protocols [8,9], where each node is equipped with an on-board GPS and selects the best forwarding nodes based on the knowledge of adjacent nodes' locations to minimize the number of broadcasts. Our contributions are briefly summarized in three-fold:

- First, we propose three sender-initiated pseudo geometric broadcast protocols in WSNs: (i) Approximating Neighbor Nodes based Broadcast Protocol (Approx), (ii) Enhanced Ad Hoc Broadcast Protocol (AHBP) with Target Forwarding Nodes (EBP($|N_f|$)), and (iii) Node Distribution Sensitive Broadcast (NDS), where $|N_f|$ is a number of forwarding nodes. A simple random backoff mechanism is also proposed to avoid possible packet contentions and collisions because of simultaneous rebroadcasts.
- Second, we modify the AHBP [10] to work in WSNs for the purpose of performance comparison. A variation of the GPS supported Broadcast Protocol for Sensor networks (BPS) [8], denoted as BPS*, is also modified by adding a sender-initiated broadcast approach.
- Third, we evaluate the performance of five broadcast protocols in terms of packet delivery ratio, number of broadcasts, and propagation delay by changing the node density and network size: AHBP, BPS*, Approx, EBP($|N_f|$), and NDS. The COUNT scheme is also used to compare the number of broadcasts measured as a part of our preliminary research in Section 3. In addition, we measure and compare the computation overhead of four broadcast protocols in terms of number of operations in a microscopic way: AHBP, Approx, EBP($|N_f|$), and NDS.

We develop a customized discrete-event driven simulator using OMNeT++ [11] to conduct our experiments and multi-dimensional analyses. Extensive simulation results indicate that the proposed Approx shows a scalable performance in terms of the number of rebroadcasts as the number of deployed nodes increases in the network. The proposed EBP($|N_f|$) and NDS protocols also show competitive performance and become viable approaches in resource constrained WSNs. This paper is significantly extended based on our prior work published in part at [12,13].

The rest of paper is organized as follows. The prior work is reviewed in Section 2. The impact of network densities and sizes is discussed in Section 3. The proposed pseudo geometric broadcast protocols are presented in Section 4.1. Section 5 is devoted for performance evaluation and analysis, followed by the issues for possible extensions of the proposed broadcast protocols. Finally, Section 6 concludes the paper.

2. Related work

In this section, we categorize and review prior broadcast techniques in terms of threshold-, structure-, probability-, and geometry-based approaches.

2.1. Threshold-based approach

When a node sends a packet, all of its neighbor nodes can overhear the packet because of a promiscuous receiving mode. Then an immediately following unconditional forwarding (i.e., flooding) can incur redundant transmissions and lead to the broadcast storm problem [4]. In order to avoid the redundant rebroadcasts that can lead to packet contentions and collisions, several threshold-based broadcast schemes have been proposed, in which their threshold values (i.e., number of same overheard packets, additional covered area, or number of neighbor nodes who have not received the packet) are adjusted based on local connectivity information [4,14]. Under the consideration of distance- and counter-based schemes [4], receiver's signal strength can be measured to approximate the distance to a packet sender [15]. The node located at the farthest from the sender rebroadcasts the packet to increase the coverage area. This approach is further extended to a heterogeneous network topology in the presence of obstacles [16], where nodes are non-uniformly deployed in the network. In [17], since packet losses are unavoidable (e.g., a predefined rebroadcast probability, $p = 0.55 < 1$) during a broadcast period, each node buffers failed broadcast packets and periodically rebroadcasts them to compensate any missed packet.

2.2. Structure-based approach

A broadcast tree between a source and multiple destination nodes is deployed to improve energy efficiency and reliability, i.e., a flood tree [18–20]. Depending on the connectivities in the tree, each node can decide whether to rebroadcast or drop a packet to minimize the redundant rebroadcasts but to maximize the network coverage. In [19], a connected dominating set (CDS) is built in a localized and distributed manner by constructing independent DSs (called dominators) and connecting all nodes (called connectors) in DSs. Then the dominators create a virtual backbone for efficient broadcast. In [20], a single node is elected as a single-initiator in a distributed manner, and it constructs a dominator tree to form a CDS based on the timers. Multiple-initiators are also considered to avoid a single-point of failure at the single-initiator. As pointed out in [18], however, the problem of finding a minimum tree that has the minimum number of forwarding nodes is proven to be NP-complete. Frequent constructing and maintaining the tree can also occur non-negligible communication overhead because of the mobility and energy constraints in mobile ad hoc networks (MANETs) and WSNs, respectively.

2.3. Probability-based approach

Each node decides whether to forward a packet with a retransmission probability p, called a gossip probability [21–24]. The gossip probability can be adaptively adjusted by considering dissemination coverage, redundant retransmission, latency, or communication overhead. In order to adaptively calculate the p, a hybrid scheme [22] is proposed by combining both counter-based [4] and probabilistic approaches. Each node estimates a time-varying local density of the network by counting the number of received packets, and then adjusts the p accordingly with a set of predefined thresholds in MANETs. In [23], each node uses an adaptive gossip probability based on local topological properties (i.e., parent, child, and sibling nodes) in WSNs, where the probability can be changed by unpredictable node and link failures. A. Gaba et al. further analyze prior Gossip3 scheme [21] in terms of the key parameters and propose a probabilistic forwarding scheme [24] based on a heuristic approach in MANETs. Each node periodically estimates its local network density (i.e., a number of neighbor nodes) and dynamically adjusts its gossip probability with the Weibull model, which is formalized by empirical results.

2.4. Geometry-based approach

Nodes are equipped with a GPS, and a subset of nodes is judiciously selected as forwarding nodes in the network. A virtual hexagon-based coverage is popularly deployed and three or six forwarding nodes located at strategic positions (i.e., hexagons' vertices) are selected to minimize the overlapped network area covered. In [8,9,25,26], unlike a blind flooding, either three or six nodes located at hexagon's vertices are selected as the forwarding nodes. An ideal optimal flooding approach [26] is considered as a way to find the minimum number of circles to cover the entire network area. In order to measure upper-bound of broadcast efficiency[1], the circles are arranged in a series of rows in which a single broadcast packet is non-realistically forwarded to only one adjacent node. The distance between two consecutive circles' center is less than or equal to the communication range of node. In a hexagon-flooding scheme [26], an initial packet sender first selects three one-hop neighbor nodes located at the vertices of hexagon as forwarding nodes. Then each forwarding node selects another two neighbor nodes as forwarding nodes thereafter for rebroadcast. This scheme can achieve 68% of the upper-bound efficiency of flooding. [8] proposes an adaptive-geometric scheme, in which a next forwarding node is self-selected based on the distance from the packet sender. The selected forwarding nodes should be located at the closest to the strategic location as well as farthest from the packet sender.

In addition to aforementioned approaches, a great deal of effort has been devoted in the coverage problem and its variants in WSNs. The network coverage problem [7,27–29] includes but not limited to (i) how to find the minimum number of nodes to completely monitor the area of interest, (ii) how to determine whether the network is covered by at least a pre-defined number of nodes, (iii) how to schedule nodes' operating intervals to maximize the lifetime while satisfying the coverage requirements, (vi) how to determine whether the network is covered by the communication range of nodes, etc. The major subjects of the network coverage problem are often a set of discrete points, whole sensor field, or desired coverage characteristics. The traditional set cover problem [30], where the minimum number of subsets is found to cover a given collection of subsets, also has been evolved in various ways in WSNs. In the k-covering problem [31], the objective is to partition nodes into a pre-defined k number of mutually exclusive covers and activate a node every k time unit to energy efficiently monitor the area of interest. The connected set cover problem is to find a connected 1-cover of smallest size, which is reported to be NP-hard. This problem is further extended to find the maximum number of partitions of nodes in the network, leading to the connected set cover partition [32,33]. Here, each partition ensures that the area of interest is fully covered and nodes within a partition are connected.

In summary, an efficient GPS-free broadcast protocols is critical in resource-constrained WSNs, but unfortunately little effort has been devoted in exploring the performance comparison study in the realm of pseudo geometric broadcast protocols.

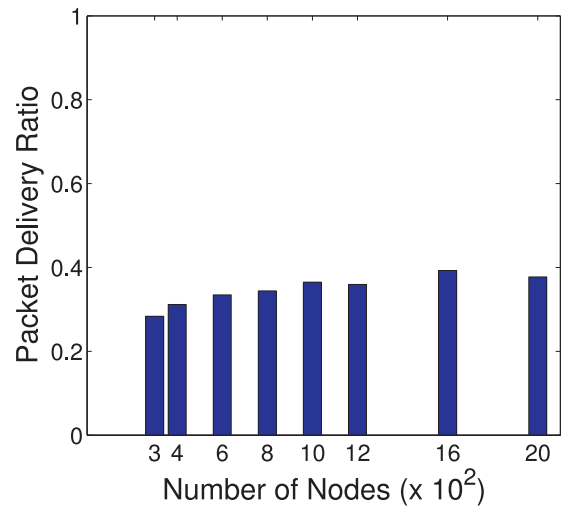


Fig. 1. The packet delivery ratio against different number of nodes deployed in the network.

3. The effect of number of forwarder nodes

In this section, we investigate the effect of number of forwarding nodes for broadcast in the network by changing number of forwarding nodes and network sizes. Due to the unreliable and lossy wireless link, frequent packet collisions, and lack of centralized coordination, broadcast packets are often prematurely terminated while they are propagated in the network. Thus, we experiment a simple broadcast based on a single forwarding node and a virtual hexagon based broadcast with the strategically deployed multiple forwarding nodes, and observe their performance with preliminary results.

For the purpose of comparison, we first experiment a simple broadcast approach, where each packet sender selects a single one-hop apart neighbor node as a forwarding node. The basic idea of this approach is that each sender selects a single node close to the largest number of its uncovered one-hop neighbor nodes, which are in fact two-hop neighbor nodes of the sender, as a forwarding node to maximize the network area covered. For the experiment, we use a 100 × 100 (m^2) rectangular network, where 300 to 2000 nodes are randomly distributed in the network. In this section, we use the same simulation testbed and parameters described in the Section 5.1. In Fig. 1, the packet delivery ratio (PDR) is measured against different number of nodes deployed in

[1] A broadcast efficiency is defined as the ratio of entire network area to the sum of network areas covered by broadcast packets in [26]. The upper-bound efficiency can be achieved when each node is located its communication range (R) apart to minimize the number of rebroadcasts. An area of circle excluding the area overlapped with its adjacent circle can be approximated as $1.913R^2$. If the entire network area can be covered by n number of broadcasts, it can be $n\times 1.913R^2$. Then the upper-bound efficiency can be calculated as $\frac{n\times 1.913R^2}{n\times \pi R^2}$, which is approximately 0.6. Thus, the efficiency of any flooding cannot exceed about 0.6. Here, $n\times \pi R^2$ is the network area when n number of nodes broadcast.

Table 1
The number of broadcasts against different network sizes.

NET_{size}	4R× 4R	7R× 7R	10R× 10R	15R× 15R	20R× 20R
N_{bcast}	13	41	78	180	324

the network. The communication range of each node, R, is 10 (m). Although the number of deployed nodes increases, PDR does not increase much and is still quite low, less than 40%. This result indicates that a broadcast packet forwarded by a single forwarding node can be terminated early before it is propagated to the rest of nodes in the network.

Second, we experiment with a network covered by a set of virtual hexagons to measure a number of broadcasts. Here, each hexagon's side length is the same as the communication range of node, R. Suppose nodes are located in each hexagon's vertex with the support of GPS. Under the assumption of no packet contention and collision, each node broadcasts a packet once. In this case, total number of broadcasts can be at most the number of deployed nodes in the network. Table 1 shows the number of broadcasts against the different network sizes ranging from 4R× 4R to 10R× 10R. This result is similar to [8] in small network sizes. Note that there are non-negligible redundant rebroadcasts because all six nodes located in each hexagon's vertices are selected as forwarding nodes to rebroadcast the packet. In this experiment, if the strategically located nodes are selected as forwarding nodes, the less number of broadcasts can be achieved.

In summary, the simple broadcast protocol based on a single forwarding node can lead to the premature termination of packet propagation and show the low PDR. Multiple number of forwarding nodes located in the strategic positions can reduce the number of broadcasts. In order to design an efficient pseudo geometric broadcast protocol, it is critical to identify a set of strategically located forwarding nodes in resource constrained WSNs, where GPS support is not available.

4. Proposed pseudo geometric broadcast protocols

In this section, we first compare sender- and receiver-initiated broadcast approaches as a basis of proposed broadcast protocols, e.g., who selects a forwarding node (Section 4.1). Then we propose three pseudo geometric broadcast protocols based on a sender-initiated broadcast approach in WSNs (Sections 4.2–4.4). A simple backoff procedure is also proposed to adjust rebroadcast delay and avoid possible packet contentions and collisions (Section 4.5).

4.1. Sender- vs. receiver-initiated broadcasts

In multi-hop broadcasts, either a single node or a set of nodes is often selected as forwarding nodes. A blind broadcasting followed by an unconditional forwarding may cause redundant broadcasts and increase the network traffic, resulting in the broadcast storm problem [4]. Thus, it is critical to judiciously select a forwarding node for efficient broadcasting. Depending on who selects a forwarding node, generally a sender- or receiver-initiated broadcast approach is adopted in the broadcast protocols.

A packet sender selects a set of forwarding candidate nodes based on its local policy in a sender-initiated broadcast. Here, each node (e.g., n_i) can be aware of its one-hop neighbor nodes (G_i) and count the number of one-hop neighbor nodes ($|G_i|$) by exchanging one-hop *Hello* and *Hello_Ack* packets piggybacked with its node *id* [34]. Each node can also be aware of its two-hop neighbor nodes, if its one-hop neighbor nodes' ids are piggybacked with the *Hello* and *Hello_Ack* packets. The sender unicasts the packet piggybacked with its forwarding nodes' ids. Only the designated nodes rebroadcast the packet and others keep quiet. Note that the sender sends a unicast packet aiming at the designated receivers only, but all the sender's adjacent nodes can still overhear it. In a receiver-initiated broadcast, however, the sender simply broadcasts a packet and its all one-hop neighbor nodes will received the packet and decide whether to rebroadcast based on their local policy. For example, each node may use a gossip probability for rebroadcasting. The sender can know whether its neighbors rebroadcast the packet only through either an implicit or explicit acknowledgment.

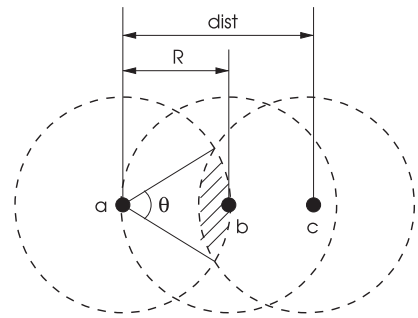


Fig. 2. Each node shares the communication range with its neighbors. Nodes n_a and n_b are located at the boundary of each other's communication range. Here, a dashed line represents a communication range.

In this paper, we deploy the sender-initiated broadcast and embed it into the proposed broadcast protocols. Because the number of forwarding nodes is fixed depending on the proposed protocols, e.g., three to six. This is different from the receiver-initiated broadcast, where the number of forwarding nodes is vary and more importantly the sender has no or less control in selecting a certain number of forwarding nodes for rebroadcast.

4.2. Approx: approximating neighbor nodes based broadcast protocol

The basic idea of the protocol is to approximate a virtual hexagon to minimize the overlapped broadcast areas, called *Approx*. A set of forwarding candidate nodes is iteratively searched, selected, and refined without the support of GPS in a heuristic manner. In the Approx, we do not pursue an optimized solution and its corresponding proof, but try to achieve it in best effort.

In order to select a set of forwarding candidate nodes, nodes close to the boundary of communication range are searched and selected to maximize the broadcast coverage. In Fig. 2, suppose nodes n_b and n_c are located at the radius (R) and *dist* distances apart from a node n_a, respectively. Each node shares the communication range with its neighbors. Generally an overlapped area (shaded area) between two nodes (e.g., n_a and n_c) located at the *dist* distance apart can be calculated as,

$$A_{dist}^{overlap} = 2\left(\pi R^2 \frac{\theta}{2\pi} - \frac{dist}{2}\sqrt{R^2 - \frac{dist^2}{4}}\right). \tag{1}$$

If we consider a case that two nodes (e.g., n_a and n_b) are located at the R distance apart, the *dist* and θ can be replaced with the R and $\frac{2\pi}{3}$, respectively. Then the overlapped area can be calculated based on Eq. (1),

$$A_R^{overlap} = 2R^2\left(\frac{\pi}{3} - \frac{\sqrt{3}}{4}\right). \tag{2}$$

Compared to the area of n_a's communication range (A_{n_a}), $A_R^{overlap}$ shares approximately 39.1% expressed as below,

$$\frac{A_R^{overlap}}{A_{n_a}} = \frac{2R^2(\frac{\pi}{3} - \frac{\sqrt{3}}{4})}{\pi R^2} = \frac{2}{3} - \frac{\sqrt{3}}{2\pi} \approx 0.3910\ldots \tag{3}$$

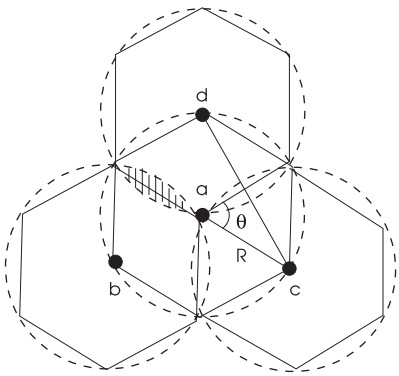


Fig. 3. Three forwarding candidates (n_b, n_c, and n_d) are selected based on the distance among nodes, close to the border of the sender's (n_a) communication range.

Table 2
Summary of the key notations used in the pseudo codes.

Notation	Meaning
n_s	A sender node holding a packet to broadcast.
n_i^*	One-hop neighbor nodes of n_i.
$\|n_s^*\|$	A number of one-hop neighbor nodes of n_s.
$n_{s,i}^*$	A one-hop neighbor node, n_i, of n_s.
$\{n_s^*\}$	A set of one-hop neighbor nodes of n_s.
$n_{s,i,j}^+$	A two-hop neighbor node, n_j, of n_s via $n_{s,i}^*$.
$\{n_{s,i}^+\}$	A set of two-hop neighbor nodes of n_s via $n_{s,i}^*$.
$\{n_s^+\}$, $\|\{n_s^+\}\|$	A set of two-hop neighbor nodes of n_s, where $\{n_s^+\} = \sum_i \{n_{s,i}^+\}$, and its cardinality.
$\{F_s\}$	A set of forwarding nodes of n_s.
$\{B_s\}$	A set of one-hop neighbor nodes close to the boundary of communication range of n_s.
ϵ, δ	System parameters used in Eqs. (4) and (7), respectively.

Since we consider a dense WSN where nodes are uniformly distributed in the network, each node shares the same or similar number of neighbor nodes. For example, if two nodes are located at the same location, each node shares 100% of its communication range with the other. If two nodes are located at more than $2R$ distance apart, no communication range is shared. In Eq. (3), if both n_a and n_b share about 39.1%, each node is then most likely located at the border of the other node's communication range.

In order to identify the nodes located at the communication boundary (B_s) of a sender (e.g., n_s), n_s first counts its one-hop neighbor nodes (e.g., n_i), $|G_s|$. n_s also counts the number of two-hop neighbor nodes located within n_s's communication range, $|G_i \cap G_s|$. Then n_s calculates whether $\frac{|G_i \cap G_s|}{|G_s|}$ is about 39.1% or 39.1% + ϵ, as shown in Eq. (4). If n_i satisfies this equation, it is most likely located at the boundary of n_s's communication range and becomes a forwarding candidate node. Here, the ϵ is a system parameter and provides a margin to flexibly select the nodes close to the communication boundary.

$$B_s = \{n_i \in G_s \mid \frac{|G_i \cap G_s|}{|G_s|} \approx 39.1\% \text{ or } 39.1\% + \epsilon\}. \tag{4}$$

Because multiple nodes can satisfy Eq. (4), we consider a virtual hexagon-based coverage approach to further refine a set of forwarding candidate nodes for minimizing the overlapped broadcast areas in dense WSNs, as shown in Fig. 3. Each node establishes a virtual hexagon within its communication range. In this paper, three forwarding candidates (i.e., n_b, n_c, and n_d) located at $\sqrt{3}R$ apart each other are considered to minimize the overlapped broadcast areas. Here, the *dist* and θ become $\sqrt{3}R$ and $\frac{\pi}{3}$, respectively. Then the overlapped area (i.e., shaded area) can be calculated based on Eq. (1),

$$A_{\sqrt{3}R}^{overlap} = 2\left(\pi R^2 \frac{1}{6} - \frac{\sqrt{3}R}{2}\sqrt{R^2 - \frac{3R^2}{4}}\right) = \frac{\pi R^2}{3} - \frac{\sqrt{3}R^2}{2}. \tag{5}$$

$A_{\sqrt{3}R}^{overlap}$ shares approximately 5.8% compared to A_{n_a},

$$\frac{A_{\sqrt{3}R}^{overlap}}{A_{n_a}} = \frac{\frac{\pi R^2}{3} - \frac{\sqrt{3}R^2}{2}}{\pi R^2} = \frac{1}{3} - \frac{\sqrt{3}}{2\pi} \approx 0.0577\ldots \tag{6}$$

Based on Eq. (6), n_s further searches the nodes close to the boundary of its communication range (i.e., B_s), whether they are located at $\sqrt{3}R$ apart. For example, two nodes (e.g., n_i and n_j) are first selected in B_s, in which both n_i and n_j are in G_s. n_s counts the number of both n_i's and n_j's common one-hop neighbor nodes, $|G_i \cap G_j|$. Then n_s calculates whether $\frac{|G_i \cap G_j|}{|G_s|}$ is about 5.8% or 5.8% + δ, as shown in Eq. (7). If both n_i and n_j satisfy this equation, two nodes are mostly likely located at $\sqrt{3}R$ apart each other. The distances of all the pairs of nodes in B_s are estimated. Here, the δ is a system parameter and provides a margin to flexibly select a pair of nodes located at about $\sqrt{3}R$ apart each other.

$$F_s = \{n_i, n_j \in B_s \ \wedge \ n_i, n_j \in G_s \mid \frac{|G_i \cap G_j|}{|G_s|} \approx 5.8\% \text{ or } 5.8\% + \delta\}. \tag{7}$$

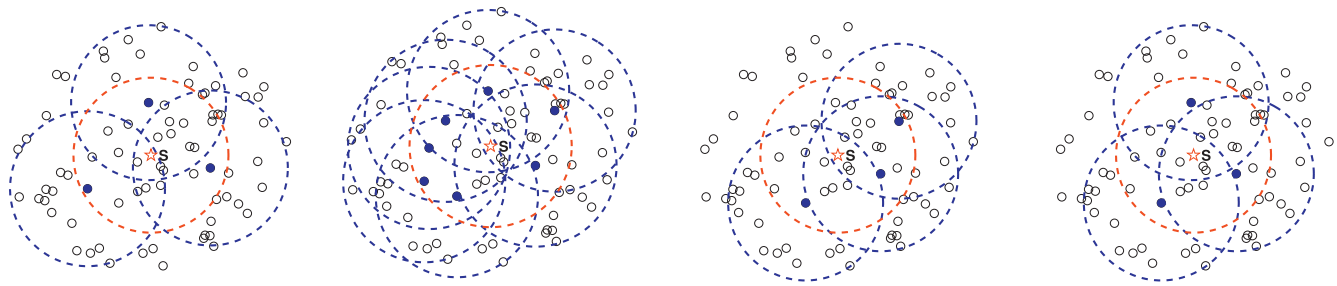
In case of multiple forwarding candidates in F_s, the best three forwarding nodes estimated as close as 5.8% are selected.

In the Approx, the sender may not find enough number of nodes that satisfy the previous two steps (Eqs. (4) and (7)) in case of a sparse or non-uniform network, where nodes are distributed in a group or cluster manner. Especially, the assumption of uniform communication range may not always be applicable in WSNs. For example, each node may have variable communication ranges in an energy harvesting motivated WSN [34]. In order to find enough number of forwarding nodes that satisfy Eqs. (4) or (7), the Approx provides a flexible way to search additional number of forwarding candidate nodes by increasing the ϵ or δ. In this paper, both ϵ and δ are initially set to zero and the sender increases them linearly. Here, we do not use predetermined values for the ϵ and δ. Major operations of the Approx are summarized in Algorithm 1. Here, the Approx selects the best three forwarding nodes in $\{F_s\}$. The key notations used in the pseudo codes are also summarized in Table 2.

Algorithm 1 The Approx.

```
{B_s}, {F_s} = ∅;
for ∀ n_i ∈ {n_s^*} do
    if |{n_i^*} ∩ {n_s^*}| ≤ |n_s^*| × (39.1% + ε) then
        {B_s} = {B_s} ∪ n_{s,i}^*;
    end if
end for
for ∀ n_i, n_j, n_k ∈ {B_s}, where i ≠ j, k ≠ j, k ≠ i do
    if (|{n_i^*} ∩ {n_j^*}| ≤ |n_s^*| × (5.8% + δ)) ∧ (|{n_i^*} ∩ {n_k^*}| ≤ |n_s^*| ×
    (5.8% + δ)) ∧ (|{n_j^*} ∩ {n_k^*}| ≤ |n_s^*| × (5.8% + δ)) then
        {F_s} = {F_s} ∪ {n_i, n_j, n_k};
    end if
end for
```

In this paper, we visualize the locations of selected forwarding nodes to clearly compare the broadcast protocols in Fig. 4. When a packet sender (n_s) selects a set of forwarding nodes, we snapshot the network topologies including the sender's one and two hop neighbor nodes. In Fig. 4(a), three nodes are selected as forwarding nodes. They are located near the boundary of the sender's communication range and approximately $\sqrt{3}R$ apart each other. There are two nodes located closer to the boundary of communication range than that of the selected forwarding nodes. However, they are not



(a) Approx (b) AHBP [10] (c) EBP(3) (d) NDS

Fig. 4. A set of forwarding nodes is selected based on the broadcast protocols. Here, both a sender (n_s) and selected forwarding nodes are marked by a star and shaded circles, respectively. The communication range is represented by a dashed line. The AHBP (Fig. 4(b)) is also depicted for the purpose of comparison with the proposed broadcast protocols.

selected because they are close to each other, and thus the condition of the second step (Eq. (7)) is not satisfied.

4.3. EBP($|N_f|$): enhanced AHBP with target forwarding nodes

The second protocol is based on the number of two-hop neighbor nodes covered in the network, called *Enhanced AHBP with Target Forwarding Nodes (EBP($|N_f|$))*, where the number of forwarding nodes ($|N_f|$) can flexibly be adjusted. Since the EBP($|N_f|$) is extended from a prior pseudo geometric broadcast protocol, called Ad Hoc Broadcast Protocol (AHBP) [10], the AHBP is briefly introduced for comparison purpose in this paper.

In the AHBP, as a Connected Dominating Set (CDS) based approach, a sender repeatedly selects a forwarding node, called as a Broadcast Relay Gateway (BRG), until entire two-hop neighbor nodes are covered. The sender first selects a one-hop neighbor node as a BRG (e.g., n_i) and constructs a local topology graph based on a two-hop neighbor table. The topology graph is reduced by removing the edges of adjacent nodes of n_i. The sender repeatedly selects a BRG, constructs its topology graph, and removes the edges of its adjacent nodes until all two-hop neighbor nodes are covered by at least one BRG. If a two-hop neighbor node of n_s is only adjacent to a one-hop neighbor of n_i, then n_i becomes a BRG. Major operations of the AHBP are summarized in Algorithm 2.

Algorithm 2 The AHBP.

$\{F_s\} = \emptyset$;
for $\forall\ n^*_{s,i} \in \{n^*_s\}$ **do**
  for $\forall\ n^+_{s,i,j} \in \{n^+_{s,i}\}$ **do**
    if $n^+_{s,i,j} \notin \{n^+_{s,k}\}$, where $i \neq k$ **then**
      $\{F_s\} = \{F_s\} \cup n^*_{s,i}$;
      Delete $\{n^+_{s,i}\}$ from $\{n^+_s\}$;
    end if
  end for
end for
while $\{n^+_s\} \neq \emptyset$ **do**
  Find $n^*_{s,i}$ with the largest $|\{n^+_{s,i}\}|$;
  $\{F_s\} = \{F_s\} \cup n^*_{s,i}$;
  Delete $\{n^+_{s,i}\}$ from $\{n^+_s\}$;
end while

In Fig. 4(b), seven nodes are selected as forwarding nodes but they are not always located at the boundary of the sender's communication range. The sender greedily selects a number of forwarding nodes not to miss any two-hop neighbor node. Thus, the AHBP shows more overlapped broadcast areas compared to other protocols.

In the EBP($|N_f|$), unlike the AHBP, a sender selects a one-hop neighbor node as a forwarding node that can cover the maximum number of two-hop neighbor nodes based on a two-hop neighbor table. Here, $|N_f|$ is the target number of forwarding nodes. The sender considers only uncovered two-hop neighbor nodes to select additional forwarding node to reduce the overlapped broadcast areas. The sender continues selecting a forwarding node until all two-hop neighbor nodes are covered or $|N_f|$ number of forwarding nodes are chosen. This ending condition in selecting forwarding node is different from the AHBP, which blindly selects a forwarding node and builds its local topology graph. In the AHBP, since a sender aggressively selects multiple forwarding nodes, the broadcast area of a forwarding node can be overlapped with others' significantly. Thus, the neighbor nodes of the forwarding nodes may redundantly receive the same broadcast packet multiple times. Major operations of the EBP($|N_f|$) are summarized in Algorithm 3.

Algorithm 3 The EBP($|N_f|$).

$\{F_s\} = \emptyset$;
while $|N_f| > 0$ **do**
  Find $n^*_{s,i}$ with the largest $|\{n^+_{s,i}\}|$;
  $\{F_s\} = \{F_s\} \cup n^*_{s,i}$;
  Delete $\forall\ n^+_{s,i,j}$ from $\{n^+_s\}$;
  $|N_f| = |N_f| - 1$;
end while

In Fig. 4(c), the EBP(3) selects three forwarding nodes and results in 12 uncovered two-hop neighbor nodes. The number of forwarding nodes is significantly reduced compared to the AHBP. Unlike the Approx, however, the locations among forwarding nodes are not well-balanced. For example, the area of network coverage can be distorted and this distorted effect may also propagate.

4.4. NDS: node distribution sensitive broadcast

The basic idea of the third protocol is to maximize the network coverage with a balanced number of nodes covered, called Node Distribution Sensitive Broadcast (NDS). This approach is different from the aforementioned Approx and a prior approach, Optimized Broadcast Protocol for Sensor Networks (BPS) [8].

In the Approx, a sender approximates the location of one-hop neighbor nodes and selects a set of forwarding nodes near a strategic position, i.e., the boundary of communication range of a sender. In the BPS, a sender also selects a set of forwarding nodes near a strategic position. Note that unlike aforementioned pseudo geometric broadcast protocols, each node is equipped with a GPS and utilizes the location information of its neighbor nodes. When a node receives a broadcast packet, it discard the packet if the packet

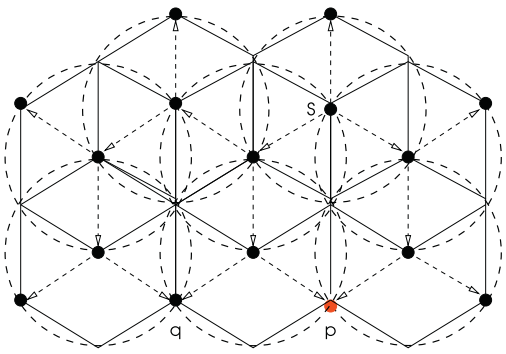


Fig. 5. A node may receive multiple broadcast packets. Here, a dashed arrow indicates the direction of packet propagation in the network.

has been transmitted from a closely located neighbor node, i.e., a one-hop neighbor node located less than $0.4R$ distance apart from the node. If not, the node selects a forwarding node that is the nearest to the strategic location, i.e., hexagon's vertex. Then the node executes a backoff procedure before transmission, in which the backoff period is proportional to the distance to the corresponding strategic location. The drawback is that positional inaccuracy of GPS can significantly affect the communication performance. Both Approx and BPS are designed to minimize the number of broadcasts by maximizing the distance between the sender and its forwarder nodes for each broadcast. In addition, since both Approx and BPS focus on selecting the forwarding nodes that are equally located apart between them and located in strategic positions (see Fig. 4(a)), however, the distribution of two-hop neighbor nodes covered is not fully considered. For example, a forwarding candidate node that covers the most two-hop neighbor nodes may not be selected as a forwarding node.

In the NDS, a sender repeatedly searches one-hop neighbor nodes and counts the number of two-hop neighbor nodes that each neighbor node can cover. Three one-hop neighbor nodes that cover the maximum number of two-hop neighbor nodes are selected as the forwarding nodes. Major operations of the NDS are summarized in Algorithm 4. Here, the NDS selects the best three forwarding nodes in $\{F_s\}$.

Algorithm 4 The NDS.

1: $\{F_s\} = \emptyset$;
2: **for** $\forall\ n_i, n_j, n_k \in \{n_s^*\}$, where $i \neq j$, $k \neq j$, $k \neq i$ **do**
3: **if** $|\{n_{s,i}^+\} \cup \{n_{s,j}^+\} \cup \{n_{s,k}^+\}| > |\{F_s^+\}|$ **then**
4: $\{F_s\} = \{n_i, n_j, n_k\}$;
5: **end if**
6: **end for**

In Fig. 4(d), although these three forwarding nodes are not near the boundary of communication range of the sender, the NDS covers equal or more number of nodes than that of the Approx.

4.5. Rebroadcast delay

In Fig. 5, a sender, n_s, initially broadcasts a packet that is propagated to the three selected forwarding nodes. Then each forwarding node executes a backoff procedure before immediately forwarding the packet to avoid possible packet contentions and collisions. Since n_s piggybacks the forwarding nodes' id in the packet, packet collisions can be minimized within the n_s's communication range. But the nodes located at the three overlapped areas (i.e., about 5.8% of n_s's communication range each) will be affected. A forwarding node may receive multiple broadcasts, i.e., n_p and n_q. Multiple broadcasts can be collided at n_p after three propagation hops via different forwarding nodes. n_q may receive multiple broadcasts but most likely they will not be arrived at the same time because of the different number of hops and the different propagation directions. Thus, n_q will discard the later arriving packet for duplication. In this paper, we deploy a simple CSMA/CA based medium access for the link layer and propose a random backoff mechanism to prevent each forwarding node from sending a packet simultaneously. For example, when a node n_i receives a packet, its random backoff period is, $t_i^{boff} = \text{Uniform}(0, \kappa \cdot |F_i|) \cdot t_{slot}$, where $|F_i|$ and t_{slot} are the number of forwarding nodes of n_i and slot time, respectively. Each slot time is on the order of tens or hundreds of microseconds. The κ is a system parameter to reduce the probability of choosing the same backoff period, i.e., $\kappa = 5$.

5. Performance evaluation

5.1. Simulation testbed

We develop a customized discrete-event driven simulator using OMNeT++ [11] to conduct our experiments. We use a square network area with different node densities, where a number of nodes is randomly distributed in the network. Here, node density is defined as the number of nodes deployed in a rectangular area, $R \times R$, where R is the communication range of each node, 10 (m). A single source node is located at the center of network and broadcasts a 2 Kbyte data packet. The radio model simulates CC2420 with a nominal data rate of 250 Kbps [35]. The radio propagation model is based on the free-space model. The simulation results are the average of 100 simulation runs.

5.2. Simulation results

We primarily compare the performance of pseudo geometric broadcast protocols: *Approx*, *EBP*($|N_f|$), *NDS*, and *AHBP* [10], BPS*, and COUNT. The AHBP is modified to work without considering a mobility in WSNs. The AHBP deploys a sender-based broadcast approach and maintains two-hop neighbor knowledge by periodically exchanging an one-hop *Hello* message. A sender selects a set of forwarding nodes that can efficiently cover the nodes located within two-hop radius. We also modify the GPS supported BPS [8] by adding a sender-initiated approach, denoted as *BPS**, where a sender selects a set of forwarding nodes located in strategic positions (i.e., hexagon vertex). Only designated forwarding nodes will rebroadcast a packet after a random backoff. Unless otherwise specified, we use eight time slots and each slot is 70 (ms) for the rest of paper. In addition, the *COUNT* is based on a set of virtual hexagons in the network under the ideal assumption of no packet contention and collision as shown in Section 3. The number of broadcasts against the different network sizes measured in Table 1 is used for comparison with other protocols.

In this paper, we investigate the impacts of node density, network size, and computational overhead, and measure packet delivery ratio (PDR), packet propagation delay, number of broadcasts, and number of operations. We vary the key simulation parameters, node density and network size.

5.2.1. Impact of node density

We measure the PDR and number of broadcasts against the node density. In Fig. 6(a), all protocols achieve high PDR for entire node densities except the Approx. Since the Approx is designed primarily for a dense network, as pointed out in [12], it shows the lowest PDR in the lower node densities. This is because three well-balanced forwarding nodes are not selected. The AHBP shows the highest PDR because it aggressively selects a number of forwarding

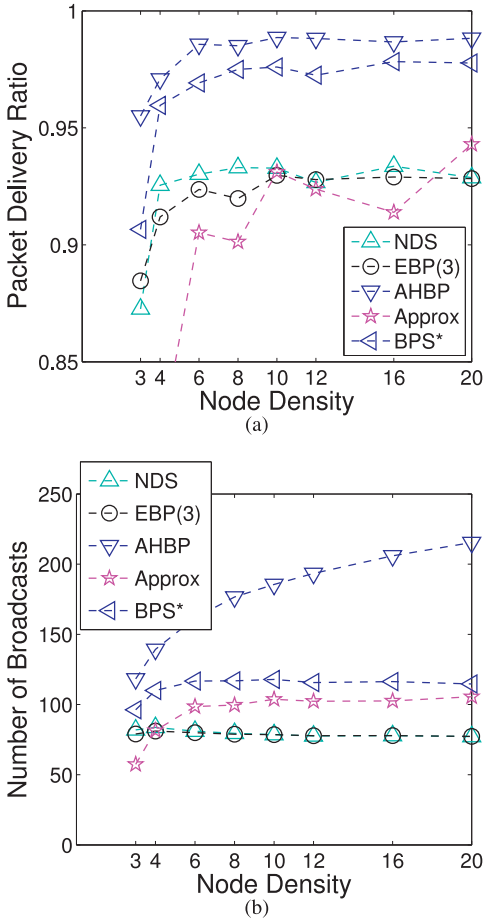


Fig. 6. Comparisons of packet delivery ratio and number of broadcasts.

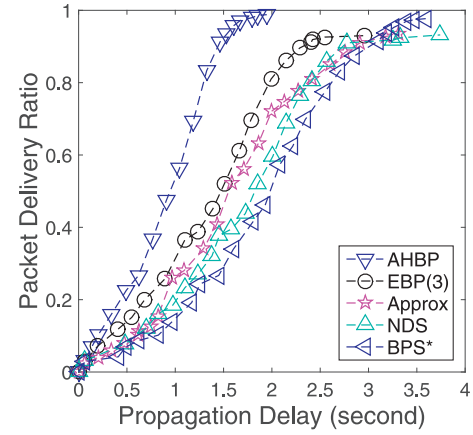


Fig. 7. Comparison of propagation delay and packet delivery ratio.

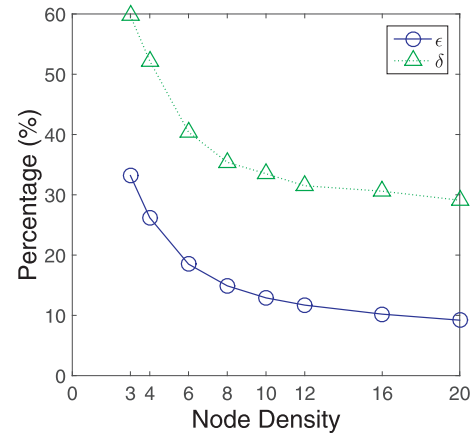


Fig. 8. Changes of ϵ and δ.

nodes to maximize the network coverage. Although a less number of forwarding nodes is selected, the BPS* still shows competitive performance with the AHBP because only strategically located nodes are selected as forwarding nodes. The proposed EBP(3) and NDS show higher PDR than that of the Approx. Since more strategically located nodes can be found as the node density increases, the Approx shows the higher PDR in the largest node density.

In Fig. 6(b), the AHBP shows the largest number of broadcasts as the node density increases. This is because more number of forwarding nodes is selected to cover more number of deployed nodes in the network. Both BPS* and Approx show a scalable performance as the node density increases because the same number of forwarding nodes is selected regardless of the node density. The proposed EBP(3) and NDS show the lowest number of broadcasts, similar performance with the COUNT, because well-balanced forwarding nodes are selected to minimize the number of rebroadcasts.

Fig. 7 shows both packet propagation delay and PDR with 1000 nodes deployed in the network. The AHBP shows the shortest delay because more number of forwarding nodes is selected for rebroadcasting than that of other protocols. The Approx, EBP(3), and NDS show competitive performance. Note that although these three protocols cannot achieve 100% PDR, they show shorter delay than that of the BPS*, in which strategically located forwarding nodes are selected based on the location information.

In the Approx, we show the average values of ϵ and δ depending on the node density in Fig. 8. We vary the number of nodes deployed from 300 to 2000 in the network. For example, when 300 and 600 nodes are deployed in the network, the average number of neighbor nodes is 9.4 and 18.8 respectively. To find the enough number of forwarding candidate nodes that satisfy Eqs. (4) and (7), both ϵ and δ increase. When 300 nodes are deployed in the network, both ϵ and δ are 33.2% and 59.7% respectively. As the number of deployed nodes increases in the network, both ϵ and δ decrease because more number of nodes located at the strategic locations are found. The δ shows higher values than ϵ for entire number of deployed nodes, because it is harder to find the number of nodes satisfying the Eq. (7) compared to that of the Eq. (4).

In the EBP($|N_f|$), we further investigate the impacts of $|N_f|$ (i.e., 3–6) and node density on the performance in Fig. 9. Here, the performance of AHBP is also measured for comparison purpose. In Fig. 9(a), the AHBP shows the highest PDR but the EBP($|N_f|$) also shows competitive performance, i.e., more than 95%. As the $|N_f|$ increases, the PDR increases. The EBP(3) shows the lowest PDR (i.e., $>$ 90%). Since only a set of nodes that covers the largest number of two-hop neighbor nodes is selected as forwarding nodes, the EBP($|N_f|$) can achieve as high PDR as the AHBP when $|N_f| \geq 4$. In Fig. 9(b), the EBP($|N_f|$) shows a scalable performance as the node density increases. Since the AHBP aggressively selects a number of forwarding nodes for rebroadcasting, the number of broadcasts increases as the node density increases. The performance results indicate that the EBP($|N_f|$) can achieve competitive PDR and the lower number of broadcasts with the less number of forwarding nodes compared to the AHBP. Note that the EBP(4) can already achieve 96% PDR with the modest number of forwarding nodes.

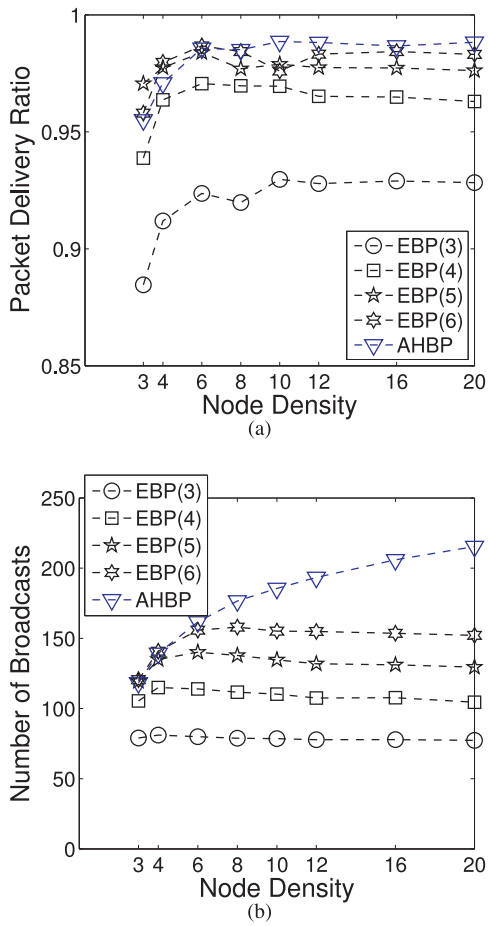


Fig. 9. Comparisons of packet delivery ratio and number of broadcasts in the EBP and AHBP.

5.2.2. Impact of network size

We measure the PDR and number of broadcasts against four different network sizes (i.e., 4R× 4R, 7R× 7R, 15R× 15R, and 20R× 20R) in Figs. 10 and 11, respectively. In Fig. 10, the AHBP shows the highest PDR for the entire node densities and network sizes. The BPS* also shows competitive performance with the AHBP. Under the small network sizes (i.e., 4R× 4R and 7R× 7R), the Approx, EBP(3), and NDS shows low and fluctuated PDR because of the less number of deployed nodes in the networks. As the network size increases, however, these three protocols are close to 95% PDR. The Approx does not perform well with the low node density for entire network sizes.

In Fig. 11, the EBP(3) and NDS show the lowest number of broadcasts for entire node densities and network sizes. Since these protocols judiciously select a set of forwarding nodes that can cover the largest number of two-hop neighbor nodes, the number of broadcasts is minimized and shows a scalable performance. Other protocols also show a scalable performance except the AHBP. Note that the NDS shows a similar or slightly higher performance compared to the EBP(3). Although the NDS repeatedly searches a group of forwarding nodes that can maximize the network coverage, uneven node distribution and packet contentions or collisions may affect the performance.

5.2.3. Computational overhead and analysis

We measure the computational overhead of protocols in terms of the number of operations required for rebroadcasts. In the Flooding, each node rebroadcasts every received packet to all its one-hop neighbors and does not require any additional operation. Since both BPS and BPS* deploy a GPS and aware of the locations of their neighbor nodes, they can easily select a set of forwarding nodes close to the strategic positions. Thus, we analyze and compare the computational overhead of four other broadcast protocols, Approx, NDS, EBP, and AHBP. In these four protocols, each node is aware of its two-hop neighbor nodes via exchanging one-time explicit *Hello* and *Hello_Ack* packets as aforementioned in Section 4.1. Each node maintains the lists of one-hop and two-hop neighbor nodes and keeps them in a $g \times g$ table,[2] where g is the average number of one-hop neighbor nodes in the network.

Approx: A sender searches three one-hop neighbor nodes as forwarding nodes that are close to the boundary of communication range of the sender. The sender first considers all one-hop neighbor nodes (g) and measures whether their corresponding one-hop neighbor nodes share about 39.1% of neighbor nodes with the sender's ($2g$), resulting in $2g^2$ operations. Since the sender may adjust ϵ and δ to find enough number of forwarding candidate nodes, we assume that the sender only considers the one-hop neighbor nodes located at least or more than half of its communication range, $0.5g$, for the sake of simplicity. Second, the sender repeatedly chooses a group of three one-hop neighbor nodes based on the $0.5g$ and measures whether they share 5.8% of neighbor nodes with each other ($2g$), resulting in $\binom{0.5g}{3}3g$ operations. Thus, the total computational overhead of the Approx can be expressed as $2g^2 + g + \binom{0.5g}{3}3g$.

NDS: A sender chooses three one-hop neighbor nodes as forwarding nodes that can cover the maximum number of its two-hop neighbor nodes. The sender considers its g one-hop neighbors and counts their corresponding one-hop neighbor nodes, resulting in $2g^2$ operations. If an one-hop neighbor node is located at the boundary of communication range of the sender, it can cover upto 61% of two-hop neighbor nodes that are not overlapped with the one-hop neighbor nodes of the sender. Depending on the location of one-hop neighbor nodes, however, they cover 41% of two-hop neighbor nodes of the sender in average. In order to maximize the coverage, the sender repeatedly chooses a group of three one-hop neighbor nodes and counts the number of two-hop neighbor nodes covered. These need $\binom{g}{3}$ and $3(0.41g)$ operations, respectively. Then the sender selects the group of three one-hop neighbor nodes that show the highest number of two-hop neighbor nodes covered, leading to another $\binom{g}{3}$ operations. The total computational overhead of the NDS can be expressed as $2g^2 + \binom{g}{3}3(0.41g) + \binom{g}{3}$, but it can be further refined by considering a half of one-hop neighbor nodes that are located closer to the boundary of communication range of the sender ($0.5g$), as shown in the Approx. Thus, the final form of computational overhead can be expressed as $2g^2 + g + \binom{0.5g}{3}3(0.41g) + \binom{0.5g}{3}$.

EBP($|N_f|$): Unlike to the NDS, a sender chooses $|N_f|$ one-hop neighbor nodes as forwarding nodes. The sender considers its one-hop neighbor nodes and counts their corresponding one-hop neighbor nodes, resulting in $2g^2$ operations. The sender selects the one-hop neighbor node as a forwarding node that can maximumly cover two-hop neighbor nodes (g), which are not overlapped with

[2] In the table, the left most column is the list of one-hop neighbor nodes of a sender and each row is the list of their corresponding one-hop neighbor nodes (i.e., two-hop neighbor nodes of the sender). The first column and its corresponding row are sorted by the node id to reduce the number of comparisons. If a sender counts a number of shared neighbor nodes with its one-hop neighbor node, it searches the lists of one-hop (first column) and two-hop neighbor nodes (corresponding row) by using pointers that indicate each element in the lists. The average number of comparisons is g and $2g$ for the best and worst cases, respectively.

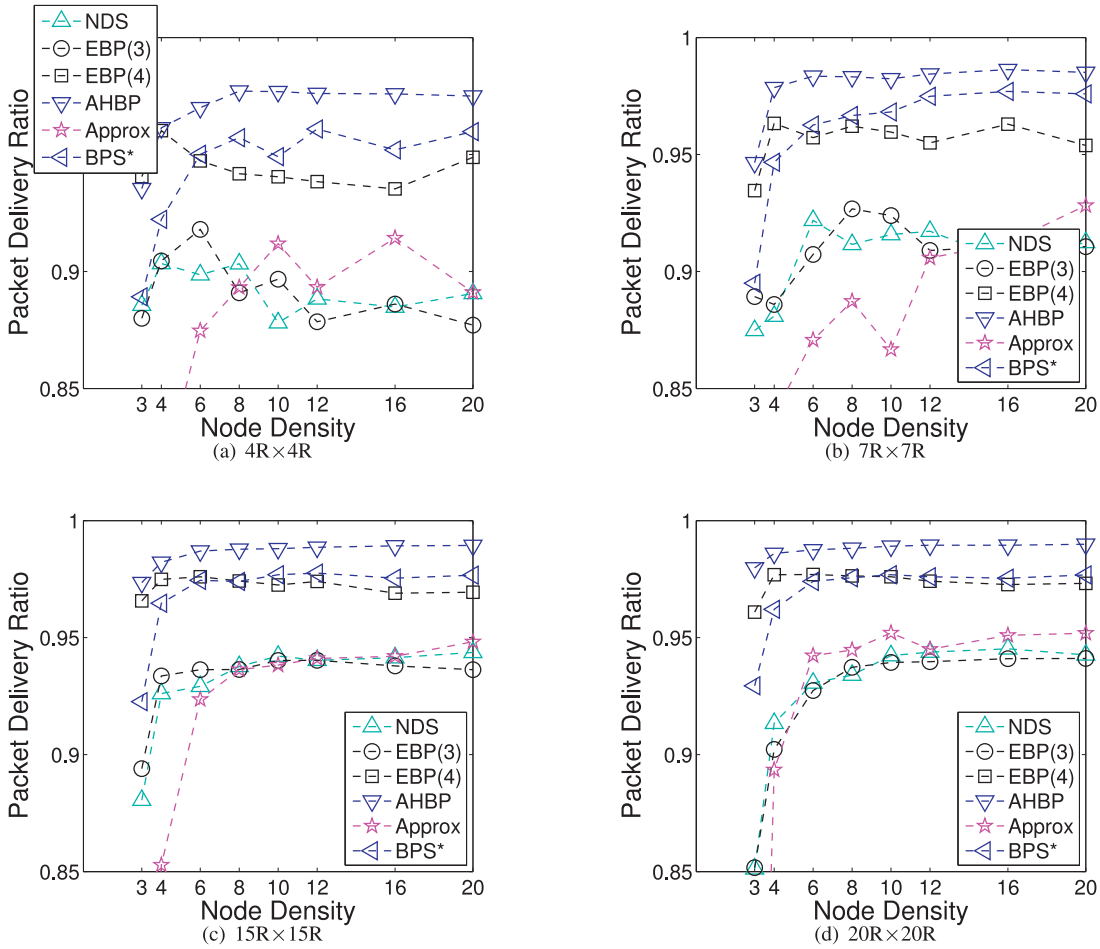


Fig. 10. Comparisons of packet delivery ratio against different network sizes.

the one-hop neighbor nodes of the sender. This selection process continues until the sender selects $|N_f|$ one-hop neighbor nodes to cover all its two-hop neighbor nodes. When $|N_f|$ is equal to three, for example, the NDS(3) requires $(g-1)(0.41g+0.41g)$, $(g-2)(0.35g+0.35g)$, and $(g-3)(0.29g+0.29g)$ operations to select the first, second, and third forwarder nodes, respectively. Here, the first forwarding node is selected based on the 41% of one-hop neighbor nodes of the sender. Likewise, the second forwarding node is considered based on the 35% of one-hop neighbor nodes of the sender without including the first forwarding node. Thus, the total computational overhead of the NDS can be expressed as $3.52g^2 + 0.78g - 3$.

AHBP: A sender continuously chooses a set of one-hop neighbor nodes as forwarding nodes until its all two-hop neighbor nodes are covered. Similar to the NDS and EBP, the sender considers all one-hop neighbor nodes and counts their corresponding one-hop neighbor nodes, resulting in $2g^2$ operations. The sender selects the one-hop neighbor node that maximumly covers its two-hop neighbor nodes, resulting in $0.5g(g-1)(0.41g+0.41g)$. Note that the PDR of EBP(6) is close to the AHBP as shown in Fig. 9(a). In this paper, we assume that the sender chooses upto seven forwarding nodes for the sake of simplicity in calculating the computational overhead. Thus, the computational overhead of the AHBP can be expressed as $3.9725g^2 - 9.285g - 14$.

In summary, Fig. 12 shows the computational overhead of four broadcast protocols. Both EBP(3) and AHBP show low number of operations for entire node densities because a sender simply chooses a set of forwarding nodes based on the number of two-hop neighbor nodes covered without considering the strategic location. As the node density increases, both Approx and NDS show high number of operations. This is because the sender extensively searches three forwarding nodes that are closely located at the boundary of communication range of the sender and are equally located apart between them. Thus, this can lead to low number of broadcasts in both Approx and NDS (see Fig. 11).

5.3. Discussion

In this subsection, we discuss the design issues that can be further investigated and deployed to potentially extend the proposed broadcast protocols.

5.3.1. Non-uniform communication range

In this paper, we implicitly assume that all nodes have a uniform communication range in WSNs. We need to relax this assumption because a node's communication range may not be deterministic or even symmetric in reality. Multi-path reflection, fluctuating signal power, ambient noises, or non-uniform intensity of radiation may affect the communication range [36,37] or the radiation pattern [38]. Note that although each node can estimate its neighbor nodes' distance based on the receiving signal strength without the support of GPS, there is a non-negligible error for the estimated distance. Under time-varying communication ranges, a sender-initiated broadcast approach is better because a sender can flexibly choose a set of forwarding candidate nodes and refine them. Since this non-uniform communication range can affect

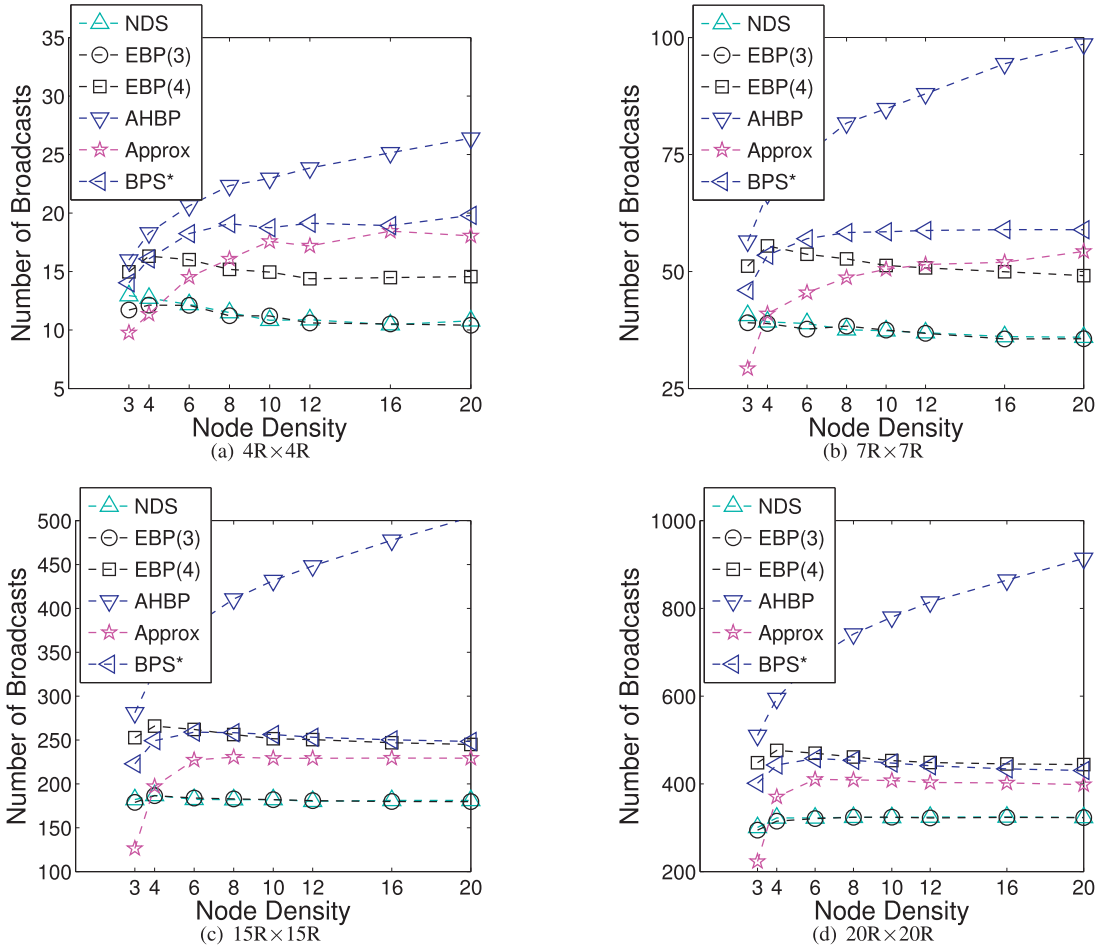


Fig. 11. Comparisons of number of broadcasts against different network sizes.

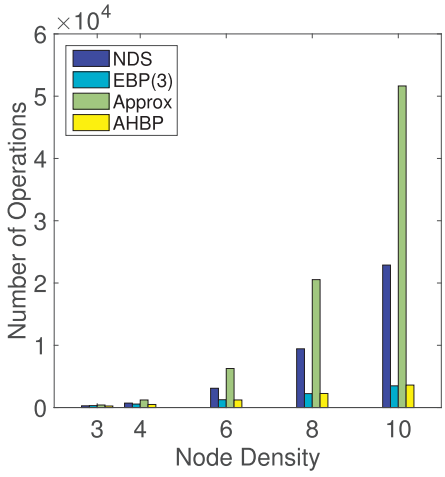


Fig. 12. The comparison of computational overhead in terms of number of operations.

a shape of virtual hexagon, nodes located close to the strategic positions may not be well balanced in the network. Thus, how to efficiently select a set of forwarding nodes becomes an issue to avoid a skewed propagation of broadcast packets.

5.3.2. Energy harvesting motivated WSNs

As a part of rapidly emerging Internet of Things (IoT), where a myriad of multi-scale nodes and devices are seamlessly blended, WSNs will play an important role in building a ubiquitous computing and communication infrastructure. In order to enhance flexible information accessibility and availability, WSNs are often required to operate long-term sensing and communicating operations on the order of weeks or even months. Since most WSNs are powered by batteries, as we also implicitly assumed in this paper, it is essential to extend the lifetime of the batteries. In light of this, energy harvesting from immediate environmental resources (e.g., vibrations, thermal gradients, lights, etc.) has been attracting a considerable attention as a promising technique to either eliminate replacing the batteries or at least reduce the frequency of recharging the batteries [39]. With energy harvesting, each node can deploy variable communication range levels depending on the energy availability and residual energy to provide customized services. This is frequently witnessed in CISCO Aironet 340/350 series, Wi-Fi networks [40], and energy harvesting motivated WSNs [34,39]. In this paper, the proposed EBP($|N_f|$) can be extended by adaptively changing the number of forwarding nodes ($|N_f|$) in the presence of multiple levels of communication range. The longer communication range, more forwarding nodes are selected. Since nodes in the network may have the different level of communication ranges, however, this can aggravate the skewed propagation of broadcast packets.

6. Concluding remarks

In this paper, we proposed three pseudo geometric broadcast protocols based on a sender-initiated broadcast approach in resource constrained WSNs. A sender searches a set of forwarding

candidate nodes and selects the best forwarding nodes located close to the strategic positions without the support of GPS. For performance evaluation study, we conducted extensive simulation experiments in terms of PDR, number of rebroadcasts, propagation delay, and computational overhead and compared the proposed broadcast protocols with other three existing protocols. The simulation results show that the proposed EBP($|N_f|$) and NDS achieve as high PDR as the AHBP and less number of broadcasts compared to other protocols. The AHBP shows the highest PDR and shortest propagation delay but the largest and non-scalable number of broadcasts becomes an issue. The Approx and BPS* also show competitive performance. The EBP(3) and AHBP show lower computational overhead for entire node densities than that of the Approx and NDS. They show a performance tradeoff between number of broadcasts and number of operations. The proposed Approx, EBP($|N_f|$), and NDS also show a scalable performance and thus, they are viable approaches in resource constrained WSNs.

References

[1] ABI research, The Internet of Things Will Drive Wireless Connected Devices to 40.9 Billion in 2020, Aug 2014.
[2] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of things (IoT): a vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (2013) 1645–1660.
[3] McKinsey Global Institute,Unlocking the potential of the Internet of Things, June 2015.
[4] S. Ni, Y. Tseng, Y. Chen, J. Sheu, The Broadcast Storm Problem in a Mobile Ad Hoc Network, in: Proc. ACM MobiCom, 1999, pp. 151–162.
[5] S. Lim, C. Yu, C.R. Das, Randomcast: an energy efficient communication scheme for mobile ad hoc networks, IEEE Trans. Mobile Comput. 8 (8) (2009) 1039–1051.
[6] NAVSTAR GPS Operations, http://tycho.usno.navy.mil/gpsinfo.html.
[7] B. Wang, Coverage problems in sensor networks: a survey, ACM Comput. Surv. 43 (4) (2011).
[8] A. Durresi, V.K. Paruchuri, S.S. Lyengar, R. Kannan, Optimized broadcast protocol for sensor networks, IEEE Trans. Comput. 54 (8) (2005) 1013–1024.
[9] Y. Lai, J. Jiang, Broadcasting with Optimized Transmission Efficiency in Wireless Networks, in: Proc. ICWMC, 2009, pp. 334–339.
[10] W. Peng, X. Lu, AHBP: an efficient broadcast protocol for mobile ad hoc networks, J. Sci. Technol. (2002).
[11] OMNeT++,http://www.omnetpp.org/.
[12] C. Liang, S. Lim, M. Min, W. Wang, Geometric Broadcast Without GPS Support in Dense Wireless Sensor Networks, in: Proc. IEEE CCNC, 2014, pp. 696–701.
[13] C. Liang, S. Lim, M. Min, W. Wang, Network Coverage Sensitive Pseudo Geometric Broadcast Protocols in Wireless Sensor Networks, in: Proc. IEEE CCNC, 2015, pp. 543–548.
[14] Y. Tseng, S. Ni, E. Shih, Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Network, in: Proc. IEEE ICDCS, 2001, pp. 481–488.
[15] H. Miranda, S. Leggio, L. Rodrigues, K. Raatikainen, A Power-Aware Broadcasting Algorithm, in: Proc. IEEE PIMRC, 2006, pp. 1–5.
[16] C. Ellis, H. Miranda, F. Taiani, Count on me: Lightweight Ad-Hoc Broadcasting in Heterogeneous Topologies, in: Proc. M-MPAC, 2009.
[17] S. Pleisch, M. balakrishnan, K. Birman, R.V. Renesse, MISTRAL: Efficient Flooding in Mobile Ad-hoc Networks, in: Proc. MobiHoc, 2006, pp. 1–12.
[18] H. Lim, C. Kim, Flooding in wireless ad hoc networks, Comput. Commun. 24 (3–4) (2001) 353–363.
[19] K. Alzoubi, P.J. Wan, O. Frieder, Message-optimal Connected Dominating Sets in Mobile Ad Hoc Networks, in: Proc. MobiHoc, 2002, pp. 157–164.
[20] K. Sakai, C.H. Huang, W. Ku, M. Sun, X. Cheng, Timer-based CDS construction in wireless ad hoc networks, IEEE Trans. Mobile Comput. 10 (10) (2011) 1388–1402.
[21] Z.J. Hass, J.Y. Halpern, L. Li, Gossip-based Ad Hoc Routing, in: Proc. IEEE INFOCOM, 2002, pp. 1707–1716.
[22] Q. Zhang, P. Agrawal, Dynamic probabilistic broadcasting in MANETs, J Parallel Distrib. Comput. 65 (2) (2005) 220–233.
[23] P. Kyasanur, R.R. Choudhury, I. Gupta, Smart Gossip: An Adaptive Gossip-based Broadcasting Service for Sensor Networks, in: Proc. IEEE MASS, 2006, pp. 91–100.
[24] A. Gaba, S. Voulgaris, K. Iwanicki, M.V. Steen, Revisiting Gossip-based Ad-Hoc Routing, in: Proc. ICCCN, 2012.
[25] V.K. Paruchuri, A. Durresi, D.S. Dash, R. Jain, Optimal Flooding Protocol for Routing in Ad-hoc Networks, in: Proc. IEEE WCNC, 2003.
[26] D. Kim, N.F. Maxemchuk, A Comparison of Flooding and Random Routing in Mobile Ad Hoc Networks, in: Proc. New York Metro Area Networking Workshop, 2003.
[27] C. Huang, Y. Tseng, The Coverage Problem in a Wireless Sensor Network, in: Proc. ACM Int'l Conf. on Wireless Sensor Networks and Applications, 2003, pp. 115–121.
[28] S. Meguerdichian, F. Koushanfar, M. Potkonjak, M.B. Srivastava, Coverage Problems in Wireless Ad-hoc Sensor Networks, in: Proc. IEEE INFOCOM, 2001, pp. 1380–1387.
[29] M. Singh, M. Gore, A solution to sensor network coverage problem, in: Proc. IEEE Int'l Conf. on Personal Wireless Communications, 2007, pp. 77–80.
[30] D. Johnson, Approximation algorithms for combinatorial problems, JCSS 9 (1974) 256–278.
[31] M.K. Mwila, K. Djouani, A. Kurien, Set K-Cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks, in: Proc. ACM/IEEE Int'l Conf. on Information Processing in Sensor Networks, 2004.
[32] N. Pervin, D. Layek, N. Das, Localized Algorithm for Connected Set Cover Partitioning in Wireless Sensor Networks, in: Proc. IEEE Int'l Conf. on Parallel, Distributed and Grid Computing, 2010, pp. 229–234.
[33] D. Saha, N. Das, Distributed area coverage by connected set cover partitioning in wireless sensor networks, arXiv:1401.8152v2 [cs.NI] (2014).
[34] C. Pu, T. Gade, S. Lim, M. Min, W. Wang, Light-Weight Forwarding Protocols in Energy Harvesting Wireless Sensor Networks, in: Proc. Military Communications Conference (MILCOM), 2014, pp. 1053–1059.
[35] Castalia – a simulator for WSNs, http://castalia.npc.nicta.com.au/.
[36] N. Baccour, A. Koubaa, M.B. Jamaa, H. Youssef, M. Zuniga, M. Alves, A Comparative Simulation Study of Link Quality Estimators in Wireless Sensor Networks, in: Proc. IEEE Int'l Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems, 2009, pp. 1–10.
[37] C. Liang, S. Lim, M. Min, W. Wang, TCast: A Transitional Region Aware Broadcast Protocol in Variable Wireless Link Qualities, in: Proc. IEEE CCNC, 2016.
[38] M.K. Mwila, K. Djouani, A. Kurien, The Use of Antenna Radiation Pattern in Node Localization Algorithms for Wireless Sensor Networks, in: Proc. Int'l Conf. on Wireless Communications and Mobile Computing, 2014, pp. 856–862.
[39] S. Lim, J. Kimn, H. Kim, Analysis of Energy Harvesting for Vibration-Motivated Wireless Sensor Networks, in: Proc. Int'l Conf. on Wireless Networks, 2010, pp. 391–397.
[40] A. Akella, G. Judd, S. Seshan, P. Steenkiste, Self-Management in Chaotic Wireless Deployments, in: Proc. ACM MOBICOM, 2005, pp. 185–199.