Efficient Broadcasting Using Network Coding and Directional Antennas in MANETs

Shuhui Yang, *Member*, *IEEE*, and Jie Wu, *Fellow*, *IEEE*

Abstract—In this paper, we consider the issue of efficient broadcasting in mobile ad hoc networks (MANETs) using network coding and directional antennas. Network coding-based broadcasting focuses on reducing the number of transmissions each forwarding node performs in the multiple source/multiple message broadcast application, where each forwarding node combines some of the received messages for transmission. With the help of network coding, the total number of transmissions can be reduced compared to broadcasting using the same forwarding nodes without coding. We exploit the usage of directional antennas to network coding-based broadcasting to further reduce energy consumption. A node equipped with directional antennas can divide the omnidirectional transmission range into several sectors and turn some of them on for transmission. In the proposed scheme using a directional antenna, forwarding nodes selected locally only need to transmit broadcast messages, original or coded, to restricted sectors. We also study two extensions. The first extension applies network coding to both dynamic and static forwarding node selection approaches. In the second extension, we design two approaches for the single source/single message issue in the network coding-based broadcast application. Performance analysis via simulations on the proposed algorithms using a custom simulator and *ns2* is presented.

Index Terms—Broadcasting, directional antennas, network coding, wireless ad hoc networks, simulations.

1 Introduction

Broadcasting is the most frequently used operation in mobile ad hoc networks (MANETs) for the dissemination of data and control messages in many applications. Usually, a network backbone is constructed for efficient broadcasting to avoid the broadcast storm problem caused by simple blind flooding, where only selected nodes that form the virtual backbone, called forwarding nodes, forward data to the entire network.

In MANETs, the forwarding node set for broadcasting is usually selected in a localized manner, where each node determines its own status of forwarding or nonforwarding based on local information [31], or the status of a node is designated by its neighbors [16]. A smaller-sized forwarding node set is considered to be more efficient due to the reduced number of transmissions in the network, which helps to alleviate interference and also conserve energy. The connected dominating set (CDS), as a virtual backbone, has been widely studied [21], where each node is either a forwarding node or a neighbor to a forwarding node in the set. The set is also connected. Finding a minimum CDS is NP-complete.

Li et al. [13] exploited network coding in the broadcast application. They applied coding methods to deterministic forwarding node selection approaches to gain a reduction in the number of transmissions, focusing on reducing the number of transmissions each forwarding node performs. Network coding [12] is defined as allowing intermediate nodes to process the incoming information flows. When a forwarding node, chosen by a certain approach, needs to forward several messages to all of its neighbors, while some neighbors already have some of the messages, this node can combine some of the messages to reduce the number of forwardings, and each neighbor can still get every message via decoding.

For instance, node c gets two messages from nodes a and b, respectively. In order to let a and b have each other's message, c needs to forward both the messages as a traditional forwarding node. With network coding, c only needs to forward one coded message containing both original messages through the XOR operation. Nodes a and b can decode the message with the help of their own messages through the XOR operation. Note that the network coding works only when there are multiple messages broadcast at the same time in the network.

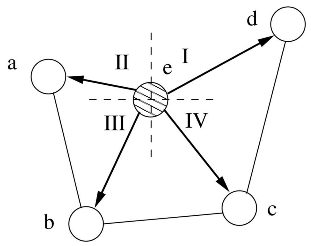
Yang et al. [33] focused on reducing the total number of forwarding directions/sectors of forwarding nodes. Using directional antennas, the omnidirectional transmission range of each node can be divided into several sectors and the transmission can be performed only in selected sectors. Therefore, by reducing the total number of transmission sectors of the forwarding nodes in the network, the interference can be alleviated, as well as the energy consumption.

In this paper, we try to combine the advantages of both network coding and directional antennas to achieve efficiency in broadcasting. We analyze the performance of these two methods and design an algorithm—Efficient Broadcast using Network Coding and Directional Antennas (EBCD), where each node decides its forwarding status using only local information and limited piggybacked broadcast state information. Our proposed design does not simply mix the two existing methods. We take advantage of the effects of interaction in both methods in an effort to achieve even better performance. Additionally, we modify the existing

- *S. Yang is with the Department of Math, Computer Science, and Statistics, Purdue University Calumet, 2200 169th St., Hammond, IN 46323. E-mail: yang246@purdue.edu.*
- *J. Wu is with the Department of Computer and Information Sciences, Temple University, 1805 N. Borad Street, Philadelphia, PA 19122. E-mail: jiewu@temple.edu.*

Manuscript received 26 Oct. 2008; revised 18 Feb. 2009; accepted 24 Feb. 2009; published online 6 Mar. 2009.
Recommended for acceptance by X. Jia.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2008-10-0430.
Digital Object Identifier no. 10.1109/TPDS.2009.44.

1045-9219/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society
Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.



(a)

	A	B	C	D
a	1	1	0	0
b	1	1	1	0
c	0	1	1	1
d	0	0	1	1
e	1	1	1	1

(b)

	P_1	P_2
I	1	1
II	1	1
III	0	1
IV	1	0

(c)

Fig. 1. (a) A sample network, (b) neighbor reception table of node e, and (c) transmission table of node e using coding and directional antennas.

directional antenna method to a dynamic mode. As shown in Fig. 1a, there are four messages, A, B, C, and D, generated from nodes a, b, c, and d, respectively. We assume that node e is selected for forwarding using a forwarding node selection method. Therefore, e needs to forward all four messages, costing four transmissions totally. In network coding-based broadcasts based on two-hop neighborhood information, e can construct a neighbor reception table, as in Fig. 1b, to record the broadcast state information of the received messages. For instance, when a sends out message A, both e and b receive it. Therefore, b is a "covered" node of message A and there is a "1" in the grid at line b, column A. Based on the table, e then codes these four messages into two combined messages to forward, $P_1(= A \oplus C)$ and $P_2(= B \oplus D)$ ($\oplus$ is the XOR operation), using some network coding algorithms. Obviously, every other node can decode these two combined messages together with the messages it already has in order to gain all four of the original messages. For instance, node b has message A, B, and C. When b receives P_2, it can use $P_2 \oplus B$ to extract message D. a can use $P_1 \oplus A$ and $P_2 \oplus B$ to obtain C and D.

With the help of directional antennas, the omnidirectional transmission range of e can be divided into K sectors (K is 4 in this example), as the dashed lines show in Fig. 1a. Then, e can restrict the transmissions of the two combined messages in only some of the sectors, as shown in table (Fig. 1c). For instance, P_1 only needs to be transmitted in sectors I, II, and IV. If we let the consumption of the transmission of one message in one sector be the unit energy consumption, traditional broadcasting, where e transmits all four messages omnidirectionally, costs 16. Broadcasting with network coding costs 8. The broadcast with network coding and directional antennas costs 6. Other than the forwarding nodes, the source nodes can also restrict the transmissions to selected sectors to further reduce the total energy consumption, as long as the message can reach a forwarding node.

Although the forwarding node/edge selection and the further network coding procedures are independent, we show that different underlying forwarding node selection approaches significantly affect the efficiency of network coding. In EBCD, we design the dynamic version of the underlying forwarding node/edge selection approach. We then use a static version without piggybacked information for it to analyze the performance and trade-offs. We find out that the energy conservation of the dynamic and static versions is comparable, although the dynamic version is slightly better. However, since the static version has less overhead, it is more practical. Also, the network coding-based broadcast approach [13] works only when there are multiple sources with multiple messages in the network. We propose two approaches as another extension to EBCD to deal with the single source with single message application: the pipeline-based (PB) approach and spread-out (SO) approach. We also discuss the detailed implementation techniques in the proposed EBCD algorithm, such as the timing issue and the neighborhood information discovery issue.

The contributions of this paper can be summarized as follows:

1. We present the advantages of the combination of the network coding and directional antenna approaches for efficient broadcasting and develop the EBCD algorithm.
2. We extend the EBCD algorithm to a static forwarding node/edge selection version to study the performance variation.
3. We propose two approaches for the application of single source with a single broadcast message.
4. We discuss some implementation techniques in EBCD, including local information collection, timer setting, and mobility handling.
5. We conduct performance analyses through simulations on the proposed algorithms in terms of energy consumption and delivery ratio.

The remainder of the paper is organized as follows: Section 2 introduces some related works and preliminaries in the field. Section 3 presents the proposed efficient broadcast using the *Network Coding and Directional Antennas approach* (EBCD). Section 4 presents two extensions to the proposed (EBCD) approach. One is the static version of EBCD and the other is to solve the single message broadcast issue. Section 5 presents some implementation details of the proposed algorithms. A performance study through simulation is conducted in Section 6. The paper concludes in Section 7.

2 Related Works and Preliminaries

2.1 Broadcast in MANETs

Both probabilistic [29] and deterministic [16], [27], [31] approaches have been proposed for efficient broadcasting. Probabilistic approaches use limited neighborhood information (local information) and require relatively high broadcast redundancy to maintain an acceptable delivery ratio. Deterministic approaches select a few forwarding nodes to achieve full delivery. Most of these approaches are localized, where each node determines its status (forwarding or nonforwarding) based on its h-hop neighborhood information (for small values of h, such as 2 or 3). The decision of forwarding nodes can be made under both static and dynamic local views. In the static approaches, only topology information is considered, whereas in dynamic ones, broadcast state information of the neighborhood is also piggybacked. More efforts have been made on developing efficient broadcast approaches. In [34], an integer programming approach and improved heuristic algorithms, were proposed. In [17], a broadcast scheme that combines the advantages of both probability and counter-based approaches, was developed. In [2], an approach that adjusts the node transmission range was proposed in an effort to achieve an efficient broadcast. Also, some

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

theoretical analyses were conducted. In [26], the energy efficiency limits in wireless broadcasting and the minimal achievable broadcast energy consumption per bit (BEB) were studied theoretically. In [14], a study was conducted on the analytical upper and lower bounds on the broadcast capacity of a wireless network when all of the nodes in the network have the same bounded transmission power and are placed in a square.

The CDS concept can be applied for broadcasting. Wu and Li [32] proposed the first localized solution for CDS construction. Peng and Lu [19] presented a scalable broadcast algorithm, where the status of a forwarding node is computed on the fly. Stojmenovic et al. [27] extended [32] to a dynamic version. Sucec and Marsic [28] developed a dynamic approach without using a backoff delay. Lou and Wu [16] devised a total/partial dominant pruning (TDP/PDP) method based on two-hop topology and broadcast state information. Wu and Dai [31] further proposed a generic CDS formation approach, which can be performed in both dynamic and static modes.

2.2 Network Coding

Network coding [1], [12] can be used to allow the intermediate nodes to combine packets before forwarding. Therefore, network coding can be used for efficient broadcasting by reducing the total number of transmissions. Fragouli et al. [7] quantified the energy savings that network coding has the potential to offer in broadcasting. They also proposed an implementable method for performing the network coding and addressed some practical issues such as setting the forwarding factor and managing generations. Liu et al. [15] derived bounds for the throughput benefit ratio, the ratio of the throughput of the optimal network coding scheme to the throughput of the optimal noncoding flow scheme. They used the general physical communication model found in [9]. In [3], it was shown that designing appropriate MAC scheduling algorithms is critical for achieving the throughput gains expected from network coding and a general framework to develop optimal and adaptive joint network coding. Scheduling schemes were also developed.

In [20], a proactive compensation packet constructed from unforwarded messages is periodically broadcast using network coding to improve the delivery ratio of the probabilistic broadcast approach. Li et al. [13] applied network coding to a deterministic broadcast approach called *partial dominating pruning* (PDP) [16] in a multiple-source broadcast application. They proved that using only XOR operation, the coding algorithm is NP-complete, and developed a greedy XOR-based approach for simplicity. The Reed-Solomon code was exploited to design an optimal Reed-Solomon code-based algorithm.

In [12], an XOR-based approach, known as the COPE architecture, was developed to improve the throughput of unicast traffic in wireless mesh networks. This method exploits the broadcast nature of the wireless medium through opportunistic network coding. Sengupta et al. [23] analyzed the throughput improvements obtained by COPE-type network coding in wireless networks from a theoretical perspective. Network coding can also be used in other applications, such as security services in wireless sensor networks [5], [11], peer-to-peer networks [6], [30], and the MAC layer protocol [8].

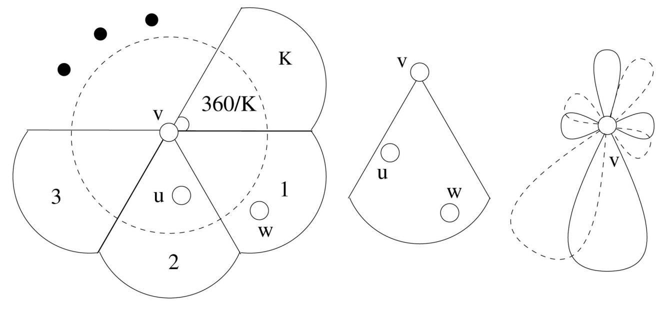


Fig. 2. Directional antenna models in (a) ideally sectorized, (b) adjustable cone, and (c) irregular beam pattern.

2.3 Directional Antennas

Two techniques are used in smart antenna systems that form directional transmission/reception beams: switched beam and steerable beam. The most popular directional antenna model is ideally sectorized, as in [10], shown in Fig. 2a, where the effective transmission range of each node is equally divided into K nonoverlapping sectors, where one or more such sectors can be switched on for transmission or reception. The channel capacity when using directional antennas can be improved and the interference can be reduced. Steerable beam systems can adjust the bearing and width of a beam to transmit to, or receive from, certain neighbors. The corresponding antenna mode is an adjustable cone as shown in Fig. 2b. In practical systems, antenna beams have irregular shapes (as shown in Fig. 2c) due to the existence of side lobes, which may cause inaccurate estimations. We will use model (Fig. 2a) in our following discussion. Some probabilistic approaches for broadcasting using directional antennas are proposed in [10], [24], [25].

Dai and Wu [4] proposed a localized broadcast protocol using directional antennas. Yang et al. [33] put forward the directional network backbone for efficient broadcasting using the directional antenna model in a static manner, where the backbone is suitable for any source node in the network. They designed the concept of a directional connected dominating set (DCDS) for the construction of a directional network backbone. DCDS extended the CDS approach for broadcasting with the help of directional antennas. The minimum DCDS problem is proven to be NP-complete. Using DCDS, not only forwarding nodes, but also forwarding edges of each forwarding node, are designated. The total energy consumption is reduced as well as the interference. They developed the node and edge coverage condition for the DCDS problem. All of the above schemes assume an omnidirectional reception mode.

3 Broadcasting with Network Coding and Directional Antennas

In this section, we first extend the approach developed in [33] to construct the DCDS to a dynamic version, where the constructed DCDS is source-based. We then combine

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

the network coding with the dynamic DCDS to develop the EBCD.

3.1 Dynamic Directional Connected Dominating Set (DDCDS)

In [33], the concept of using a directional network backbone for efficient broadcasting in conjunction with directional antennas was proposed. The omnidirectional transmission range of each node is divided into K sectors and each forwarding node only needs to switch on several sectors for transmission while the entire network receives the broadcast message. The DCDS is proposed for the construction of a directional network backbone, where each node determines locally not only its status of forwarding or nonforwarding but also its forwarding outgoing edges if it is a forwarding node. Note that the network is modeled as a directed graph. Then, in a broadcast initiated from any source node, the source uses ominidirectional transmission (or directional transmission if it detects a forwarding node in that direction) to send the message to a neighboring forwarding node. Then, forwarding nodes forward the message toward only their corresponding forwarding edges and the entire network receives the message. The DCDS is a directional network backbone assuming that K is infinite and each outgoing edge is a transmission sector. When K is finite, the sectors that contain selected forwarding edges are simply switched on for transmission to get a directional network backbone. Note that when K is 1, the DCDS problem turns into the CDS problem.

A minimum DCDS problem is to find a DCDS with the minimum number of forwarding edges, which is proved to be NP-complete. If the energy consumption of transmission in any direction is fixed, reducing the number of forwarding edges guarantees the smallest energy consumption in the application of broadcasting using directional antennas.

The *node/edge coverage condition* proposed in [33] constructs a DCDS for a given network locally at each node in a static manner. The constructed DCDS is for any source node in the network. After the exchange of "Hello" messages, each node makes a decision based on only local topology information in the initialization phase before the broadcast application starts. Here, we extend this method to a dynamic version, where each node makes a decision based not only on topology information but also on broadcast state information piggybacked in received broadcast messages. It decides its forwarding status and corresponding forwarding edges for each received broadcast message.

In our proposed *dynamic node/edge coverage condition*, each broadcast message piggybacks with it the information of its q most recently visited nodes (q is a small number such as 2 or 3). A *visited node* for a message is a node that has forwarded the message. Correspondingly, a *visited edge* for a message is an edge that has forwarded the message. Then, when a node applies the coverage condition to determine its status for a received message, it considers the information of visited nodes/edges of this message as well as local topology information. The dynamic version of the node and edge coverage conditions resemble the static ones [33] except that the node and edge priorities are updated based on the piggybacked broadcast state information. Note that the updated new priority is only valid for the corresponding message. Therefore, a node may have a different status

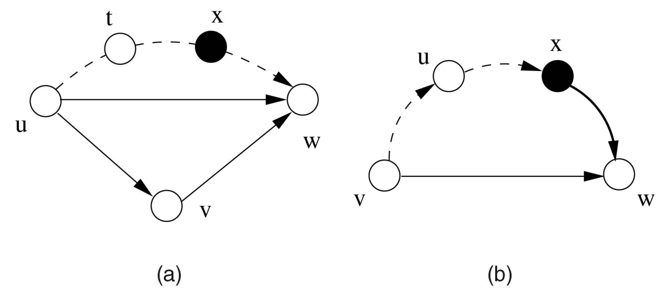


Fig. 3. Directed replacement paths in (a) node coverage and (b) edge coverage with visited nodes.

(visited or not, forwarding or not) and priorities for different messages. In the following, an unmarked status represents a nonforwarding status. A *dominating* neighbor features an incoming edge from that neighbor. An *absorbant* neighbor features an outgoing edge to that neighbor. Note that each node s has a priority $p(s)$ and such a priority is totally ordered within its h-hop neighborhood, which could be the node ID, node degree, or energy level based on different applications.

Dynamic node coverage condition. *Node v is unmarked if, for any two dominating and absorbant neighbors u and w, a directed replacement path exists connecting u to w such that 1) each intermediate node on the replacement path has a higher priority than v (including visited nodes) and 2) u has a higher priority than v if there is no intermediate node.*

Edge priority assignment. *For each edge $(v \rightarrow w)$, the priority of this edge is $P(v \rightarrow w) = (P(v), P(w))$.*

The priority of an edge is a tuple based on the lexicographic order. The first element is the priority of the start node of this edge and the second one is the priority of the end node. Therefore, there is a total order for all the edges in the graph. For example, $P(x \rightarrow y) > P(w \rightarrow v)$ if and only if $(P(x) > P(w))$ or $(P(x) = P(w)$ and $P(y) > P(v))$.

Dynamic edge coverage condition. *Edge $(v \rightarrow w)$ is unmarked if a directed replacement path exists connecting v to w via several intermediate edges with higher priorities than $(v \rightarrow w)$ or visited edges, or w is visited.*

As shown in Fig. 3, v is the current node and black nodes are visited nodes. Assume that the priority is based on the alphabetic order, i.e., $P(a) > P(b)$. Fig. 3a shows two types of directed replacement paths from u to w using the node coverage condition. When u is directly connected to w, it is required that $P(u) > P(v)$. Otherwise, when there are intermediate nodes t and x, then $P(t) > P(v)$ and $P(x) > P(v)$ since x is visited. Fig. 3b shows the directed replacement path for edge $(v \rightarrow w)$. In this case, both the intermediate edges ($(v \rightarrow u)$ and $(u \rightarrow x)$) have higher priorities than edge $(v \rightarrow w)$. Since edge $(x \rightarrow w)$ is visited, edge $(v \rightarrow w)$ can be unmarked. The difference between dynamic and static node/edge coverage conditions is that a visited node/edge has a higher priority node/edge. Note that the dynamic node/edge coverage conditions need h-hop information, which means h-hop local topology information and q-hop piggybacked visited node/edge information in each received message. For example, as in Fig. 1a, if $h = 2$, node a knows all the edges in the network except the edge between nodes c and d. Fig. 4a is a large-scale example in a 10×10-area. There

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

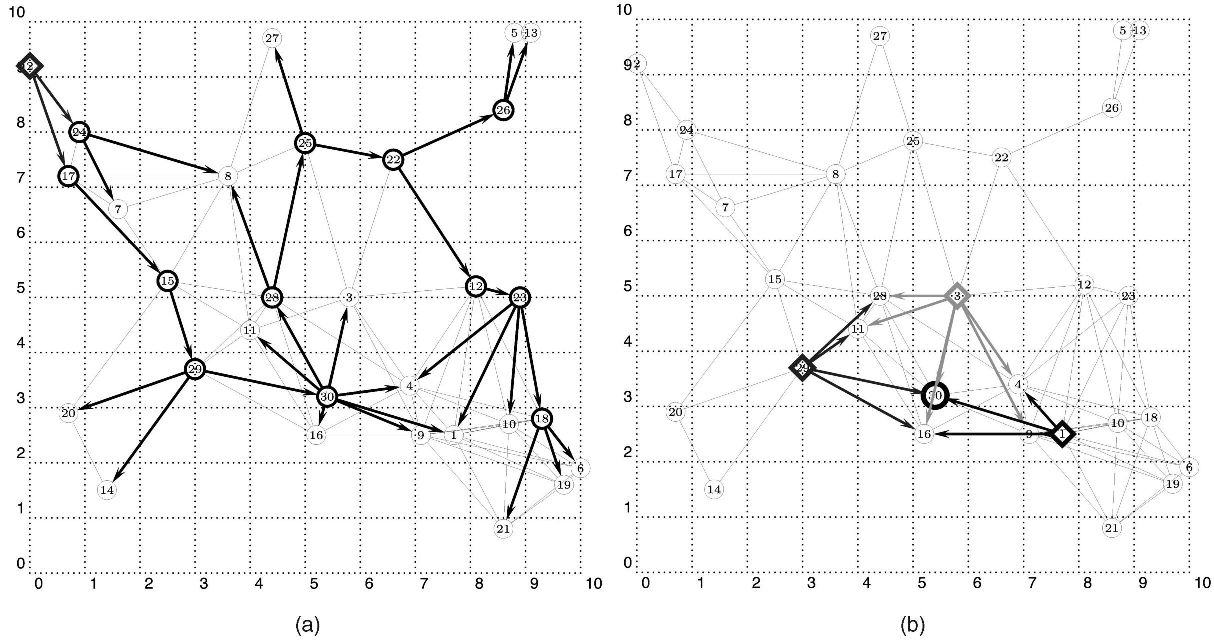


Fig. 4. Examples of (a) the dynamic edge coverage condition and (b) EBCD.

are 30 nodes with an identical transmission range of 3. Node 2 is the broadcast source (marked by red diamond). After applying the dynamic node/edge coverage condition, forwarding nodes (marked by bold circles) and their associated forwarding edges (marked by dark arrows) are determined. For example, node 15 unmarks its edges to nodes 11 and 28 since directed replacement paths connecting to nodes 11 and 28 via nodes 29 and 30, exist. Here, three-hop local information is collected.

Theorem 1. *Given a directed graph $G = (V, E), V'$ and E', generated by the dynamic node and edge coverage conditions in a broadcast, guarantee full delivery.*

Proof. Given source node s, if we can prove that for any node d in the network, there is a path with all intermediate nodes and edges designated to forward, we prove that a full delivery is achieved. We assume that all of d's neighbors form a ring as an "outer rim" of node d, like the gray area W in Fig. 5. Note that W is not empty. We assume that node u is of the highest priority in area W. If we can prove either that u is a forwarding node and $(u \rightarrow d)$ is a forwarding edge, or one of ds neighbors is a visited node and it forwards the message to d, we contradict the assumption that d cannot be reached from s. We make the two assumptions that either u is not a forwarding node or u is a forwarding node, but edge $(u \rightarrow d)$ is not a forwarding edge. We find contradictions for these two cases:

Case 1: u is not a forwarding node. Therefore, for a neighbor f of u, according to the dynamic node coverage condition, either 1) there is a replacement path connecting f to d with at least one intermediate node on it, u', or 2) f directly connects to d and the priority of f is higher than that of u. u' cannot have a higher priority than u since u is the neighbor of d with the highest priority. If u' is a visited node, d can be covered by u. For b, if f is also a neighbor of d, it cannot have higher priority than u.

Case 2: u is a forwarding node, but edge $(u \rightarrow d)$ is not a forwarding edge (an "$\times$" is on the edge). A path connecting u to d must exist, with all the edges on the path having higher priorities than $(u \rightarrow d)$ or having the visited status. Since u is the highest priority node, u'' on the path cannot be higher and has to be a visited node with a forwarding outgoing edge connecting to d. In that case, d can be covered. Note that according to the dynamic edge coverage condition, edge $(u \rightarrow d)$ is unmarked if d is visited. In this case, the assumption holds.

All of the contradictions above show that d can be reached from the source node s. $\square$

Fig. 6 shows a source-based CDS (a) in shaded nodes (s is the source), Fig. 6b is the result after applying the dynamic node/edge coverage condition. Nodes b and c are also forwarding nodes. b selects edge $(b \rightarrow d)$ as the forwarding edge and c selects edges $(c \rightarrow e)$ and $(c \rightarrow f)$. Edge $(b \rightarrow c)$ can be unmarked because a replacement path connecting b to c via s exists. s is a visited node and $(s \rightarrow b)$ is a visited edge. Therefore, the priorities of edges $(b \rightarrow s)$ and $(s \rightarrow c)$ are both higher than that of $(b \rightarrow c)$. The same is true for

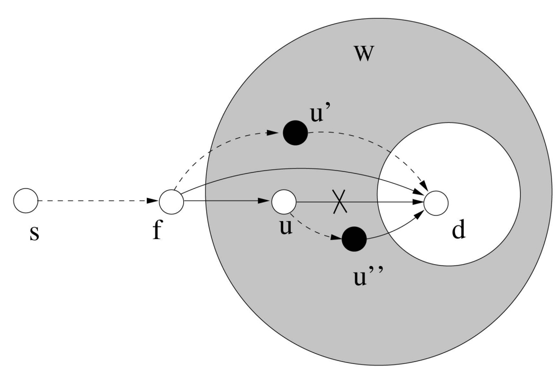


Fig. 5. Proof of dynamic node/edge coverage conditions.

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

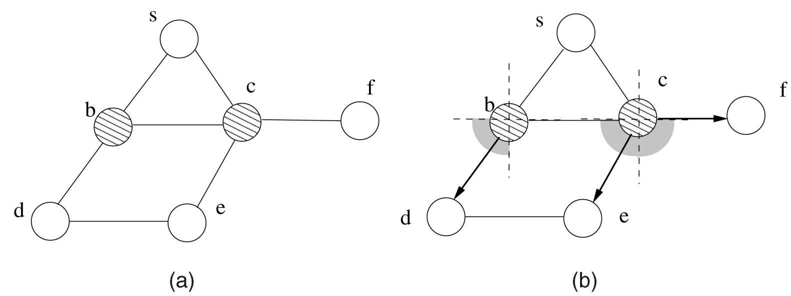


Fig. 6. (a) Forward nodes and (b) forwarding nodes and forwarding edges.

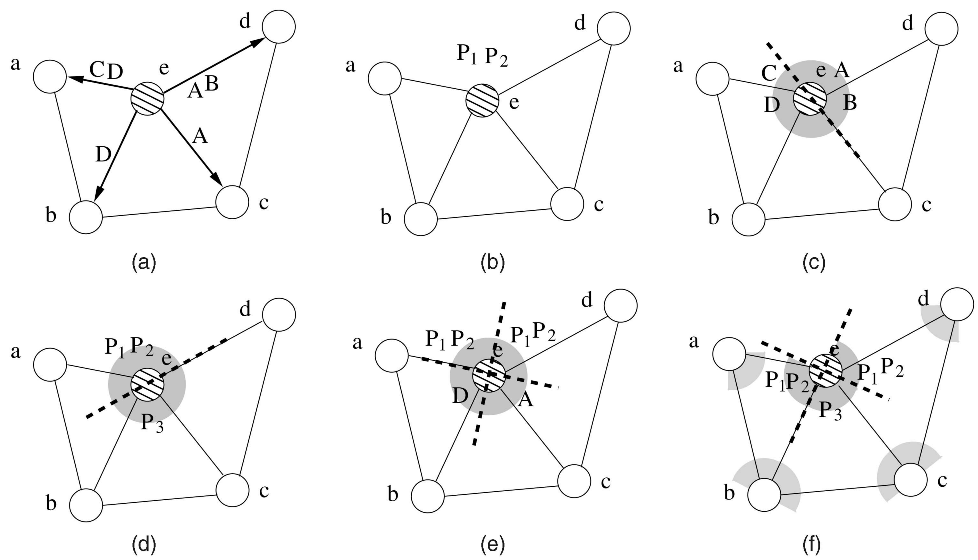


Fig. 7. (a) DDCDS, (b) coding, (c) and (d) $K = 2$, and (e) and (f) $K = 4$.

edge $(c \rightarrow b)$. Then, if K is finite, only the sectors that contain the bold arrows need to be switched on for transmission, much like the gray sectors in Fig. 6b.

3.2 Efficient Broadcasting Using Network Coding and Directional Antennas (EBCD)

In this section, we combine the network coding and directional antenna approaches into the broadcast application, exploiting the advantages of both of them.

Algorithm 1 describes EBCD executed on a node. Before the broadcast starts, each node exchanges "Hello" information with neighbors for h rounds to get the h-hop local topology information. Upon the arrival of the first message, a timer is set up and the piggybacked information in each received message is recorded to update the node priorities. When the timer expires, for each received message, the node/edge coverage conditions are applied based on the topology and broadcast state information (new priorities). The forwarding status and edges of the node are then determined. The procedure is illustrated in Fig. 7. This is the same example as in Fig. 1. Fig. 7a is the result of DDCDS after step 4 of Algorithm 1. Node e is the forwarding node for messages A, B, C, and D from nodes a, b, c, and d based on the dynamic node coverage condition. Edge $(e \rightarrow a)$ is a forwarding edge for messages C and D. Edge $(e \rightarrow b)$ is a forwarding edge for message D. Edge $(e \rightarrow c)$ is a forwarding edge for message A. Edge $(e \rightarrow d)$ is a forwarding edge for messages A and B.

Algorithm 1. EBCD algorithm at node v

Before broadcast:

1. Exchange "Hello" messages to update local topology.

Upon reception of the first message (before the timer is set up):

2. Setup the timer.
3. Update neighborhood node priorities based on each received message.
4. When the timer expires, apply dynamic node/edge coverage conditions for each message.
5. If v is a forwarding node for some messages,
   (1) align the edge of a sector to each forwarding edge,
   (2) determine coded messages in each sector using coding,
   (3) select the position with the fewest total transmissions.
6. Forward coded messages in each selected sector.

In step 5, when the timer expires, node e circumgyrates its directional antennas to let the edge of a sector align to each forwarding edge. There are at most f layouts when the number of selected forwarding edges is f. In each sector of each layout, network coding is applied to determine the final transmissions. The layout with the fewest total transmissions is then selected for use. The node then executes the forwarding. In the algorithm, we assume that steps 4, 5, and 6 can be completed before the arrival of the next message.

In EBCD, network coding is applied in each sector of a layout instead of the entire node as in [13]. We use the

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

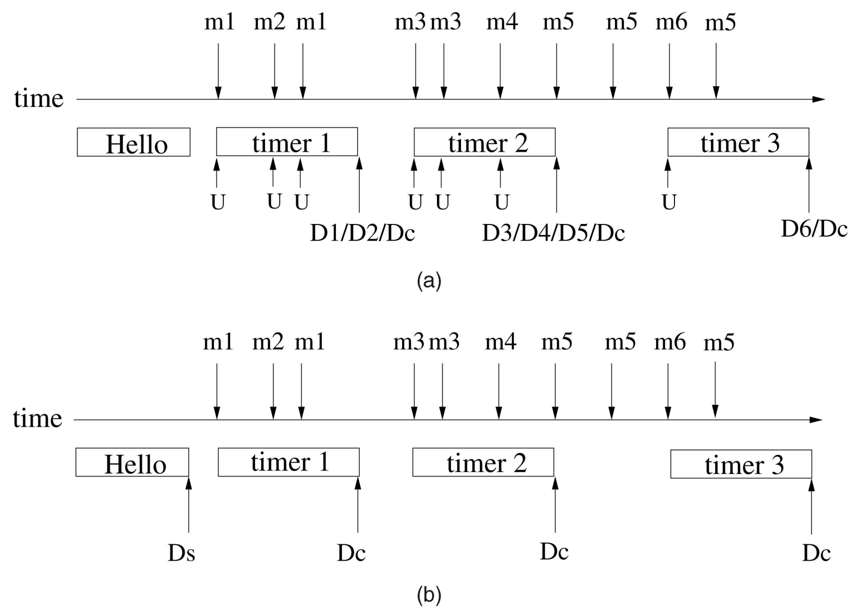


Fig. 8. Illustration of the execution procedure of dynamic/static EBCD.

XOR-based algorithm from [13]. Assuming $m_1, m_2, \ldots, m_l$ are messages received in order in this sector, $P_1, P_2, \ldots, P_t$ are the final forwarded messages (original or coded).

$$P_1 = m_1 \oplus \ldots \oplus m_{i_1},\ P_2 = m_{i_1+1} \oplus \ldots \oplus m_{i_2}, \ldots,$$
$$P_t = m_{i_t+1} \oplus \ldots \oplus m_l,$$

where each neighbor can decode from P_1 to P_t to get any missing message from m_1 to m_l. A greedy approach can be used. For the received messages in a queue, the algorithm tries to have the maximum number of messages starting from m_1 to create P_1, then to create P_2, and so on. For example, in Fig. 7b, assuming that the broadcast messages arrive in the order of A, C, B, and D at node e. P_1 is A at first, then e tries to make $P_1 = A \oplus C$. Node a needs message C and nodes d and c need message A. With P_1, all of them can decode. Therefore, $P_1 = A \oplus C$ is a correct coding. Then, e can try $P_1 = A \oplus C \oplus B$. Since node d needs message B and cannot decode P_1 to get B, this is not a correct coding. P_1 remains as $A \oplus C$. Using the same procedure, we can get $P_2 = B \oplus D$.

Fig. 7b is the result using only network coding, where e is the forwarding node forwarding the combined messages $P_1(= A \oplus C)$ and $P_2(= B \oplus D)$ omnidirectionally. Fig. 7c is one layout of EBCD using $K = 2$. Then, e needs to transmit C and D in the left sector and A and B in the right sector. Fig. 7d is another layout for $K = 2$, where e transmits P_1 and P_2 to the upper sector and $P_3(= A \oplus D)$ to the lower sector. Figs. 7e and 7f show the case, where K is 4 with different layouts. If we assume that the transmission of one message in a 90 degree sector costs one unit of energy, the energy consumption in the figures from (b) to (f) is 8, 8, 6, 6, and 5. We can see that the combination of network coding and directional antennas can improve broadcasting performance significantly in terms of energy consumption. Note that without network coding or directional antennas, the forwarding of node e costs 16.

The entire procedure can also be illustrated using Fig. 8a, where m_1-m_6 are received messages and D_1-D_6 are the corresponding forwarding nodes, edges, decisions for the received messages based on topology and priority information. U means to update the priority information based on the piggybacked information in the received message. D_c is the final transmission decision for several received messages using network coding in a valid timer. Note that the duplicated reception of a processed message is simply discarded, such as m_5, received after it has been processed and forwarded. As shown in the figure, the arrival of m_5, after timer 2 expires, will not intrigue a new timer or a new update during the timer 3 period.

The source nodes in the network can simply use omnidirectional transmission to send out the broadcast messages. In order to further reduce the total energy consumption, source nodes can only switch on sectors in which there are neighbors for transmission. In this case, the message can arrive at at least one forwarding node as well as other nonforwarding neighbors, which helps with the potential network coding conducted later on. As shown in Fig. 7f, source nodes a, b, c, and d select some sectors to switch on for transmission, shown in the light gray sectors.

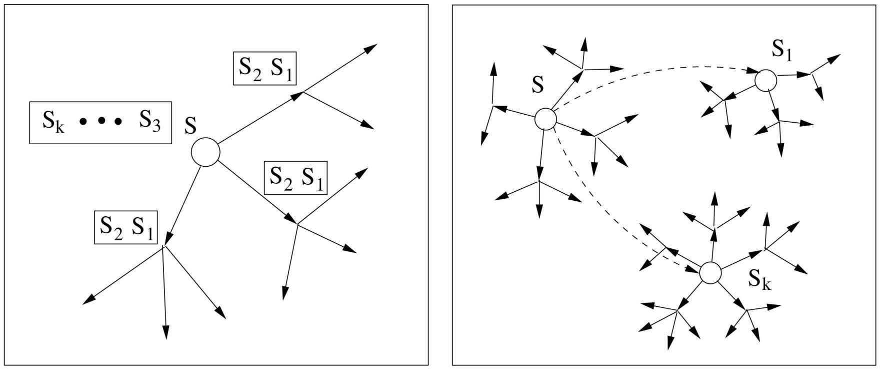
Fig. 4b is a large-scale example. We still use the sample network as in Fig. 4a. Multiple broadcast sources are necessary to apply network coding. Nodes 1, 3, and 29 are the sources in the example, with broadcast messages m_a, m_b, and m_c, respectively. We take node 30 as example. When the sources broadcasting, some of the neighbors of node 30 get a subset of the messages (shown in color arrows). Node 30 can code m_a, m_b, and m_c into two messages, $m_1 = m_a \bigoplus m_b$ and $m_2 = m_a \bigoplus m_c$, and further turn on certain sections for broadcasting based on the given K. Obviously, larger K leads to higher energy efficiency.

4 Extensions of EBCD

4.1 Static versus Dynamic Forward Node Selection

As mentioned above, Li et al. [13] applied network coding to a dynamic forwarding node selection approach, the PDP-based

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 9. Single source/single message broadcast. (a) Pipeline-based approach. (b) Spread-out approach.

approach, and stated that the coding can directly be applied to any other localized deterministic approaches for broadcasting. The previously proposed EBCD also uses a dynamic forwarding status approach. Here, we extend the proposed EBCD to a static forwarding node selection approach to analyze the overall performance of the nodes. Here, "static" means we do not use any on-the-fly information, such as whether a node has been visited or not, but only rely on the information, such as node priority and network topology, which is viewed as fixed during the broadcasting procedure.

In the static version of EBCD, we apply the coding to the static forwarding node/edge selection, and the node/edge coverage conditions in [33], as shown in Algorithm 2. We will compare the performance and trade-offs of these two algorithms in Section 6. As in Algorithm 2, in the initialization phase before the broadcast starts, local information is collected via the exchange of "Hello" messages. Then the node determines its forwarding status. This status is for all of the following received messages. Then a timer is set up when the first message arrives. When the timer expires, network coding is applied in each sector of each layout and the best one is selected for use.

Algorithm 2. Static EBCD algorithm at node v
Before broadcast:
1. Step 1 of Algorithm 1.
2. Determine forwarding status. Exit if it is nonforwarding.

Upon reception of the first message (before setup the timer):

3. Step 2 of Algorithm 1.
4. When the timer expires, follow Step 5 (1), (2), and (3) of Algorithm 1.
5. Step 6 of Algorithm 1.

The entire procedure is illustrated in Fig. 8b. After the exchange of "Hello" messages, D_s determines the status of the node and also the selected forwarding edges if it is a forwarding node for all the following received messages. Then, upon reception of the first message, a timer is set up. When the timer expires, coding is applied to all received messages to determine the final coded messages for transmission.

The size of the forwarding node set selected dynamically is smaller than or equal to the one selected statically because in the former one, the information of visited nodes helps to increase the probability of the node being a nonforwarding node. However, redundant transmissions by the extra forwarding nodes in the static manner may help to increase the potential network coding in the later phase, and hence, the overall performance. We will use simulation to verify this conjecture. The obvious advantage of the static EBCD is less overhead. The broadcast messages do not need to piggyback the broadcast state information. Also, as shown in Fig. 8b, fewer forwarding nodes, edges, decisions need to be made.

4.2 Single Source/Single Message Broadcasting

As mentioned above, broadcasting using the network coding method [13] is designed for the application of multiple sources with multiple broadcast messages, where a forwarding node has the potential of combining some of the messages to reduce the number of transmissions. In the application of a single source, only when the rate of the generation of messages is large enough, the network coding may work in nodes relatively far away from the source node.

We design two approaches for the single source/single message broadcast issue. We assume a single source in the broadcasting and the rate of the generation of messages is large enough that it can be viewed as a single message application. The basic idea is to divide the single message into several segments and treat each segment as a single broadcast message.

4.2.1 *PB Approach*

When there is only one source node in the network broadcasting one message, the source node divides the message into k segments, sends each segment as a single broadcast message, and broadcasts them one by one in the pipelined manner. In this way, the single source/single message problem turns into single source with multiple message broadcasting. In the area near the source node, the effect of network coding is insignificant since all the segments tend to come from one direction. However, in farther areas, the effect is expected to be significant. As shown in Fig. 9a, s divides the broadcast message into k segments and sends them out via k broadcasting. Therefore, the neighbors of source node s get the first broadcast message S_1, then the second broadcast message S_2 in order from s.

4.2.2 *SO Approach*

In order to enhance the effect of network coding in the single source/multiple message broadcast using the

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

message segmentation method, we can further apply the message spread method to first spread the segments out into the entire network. After the source node divides the outgoing message into k segments, it uses random walk to spread the $k-1$ of these segments. Some kind of time-to-live (TTL) control can be used to make sure the segments randomly scatter out into the network. Upon arriving at a destination, a segment is broadcast by the destination node. The source itself keeps a segment for later broadcasting. As shown in Fig. 9b, the $k-1$ segments are spread in the entire network. In this way, the application turns into a multiple source with multiple messages broadcasting. Although unicasting in the pre-processing phase costs extra overhead, the transmission reduction earned from network coding in the entire network is expected to be more significant. Note that the nodes on the unicast routes can mark themselves as visited nodes for the bypassing segments, which helps to potentially reduce transmission. Note that the PB and SO approaches may also be combined together to achieve performance trade-offs.

5 Implementation Issues

In this section, we discuss some implementation techniques of the above proposed algorithms.

5.1 Neighborhood and Piggybacked Information Collection

Note that no GPS assistance is necessary in the proposed algorithm. In Algorithm 1, each node sends out "Hello" messages K times in all K directions and accomplishes the directional neighborhood discovery. In this case, after h rounds of the message exchange, each node knows its h-hop neighborhood information, which includes both neighbors and the locations of the sectors, where these neighbors are located. According to this information, each node can create the neighbor reception table.

After a node determines its status together with the forwarding edges, it piggybacks this information in the broadcast message as part of the q most recently visited node information. The node that receives this broadcast message can extract the "visited nodes/edges" information from it.

5.2 Timer in EBCD

A timer is set for each node to collect several broadcast messages. In the static version of EBCD, it helps with the potential network coding. In the dynamic version, it also helps to collect more broadcast state information piggy-backed by these messages, to determine the status of the node. The timer selection presents the performance trade-off between energy consumption and delay. When the timer is set to 0, the effect of network coding almost reduces to 0. When the timer is large enough to counteract the difference of initial time among the broadcast messages in the network, the network coding can thoroughly be utilized. After the forwarding, the timer is reset for the next session. The value of the timer can be set in both a proactive and a reactive way. In the former method, the timer of a node can be set based on the number of neighbors of this node and the diameter of the network. In the latter one, a node can adjust the value of the timer on the fly according to the message arrival rate at this node.

5.3 Message Encoding

In step 4 of Algorithms 1 and 2, after the positions of the K sectors are determined, the forwarding node can construct K neighbor reception tables, as shown in Fig. 1, except that the neighbors in each table are only the neighbors who reside in the corresponding sector of the forwarding node. Then the forwarding node can apply any message encoding methods to determine the messages, original or coded, that need to be transmitted for each sector; such methods include the XOR-based algorithm or the Reed-Solomon-code-based optical algorithm, as proposed in [13]. As in Fig. 7f, the sector of forwarding node e that contains neighbors b and c will transmit coded message $A \oplus D$ since b needs D and has A while c needs A and has D.

5.4 Exploration for Routing

The proposed EBCD approach helps with an efficient broadcasting procedure. In the mean time, a network backbone is constructed for each broadcasting procedure. This network backbone can also be used for message routing. First, the broadcasting procedure initiated from each source sends out the "route request." Then, destination(s) of the message will send back acknowledgments. Finally, the forwarding nodes along the selected routes further reduce its selected forwarding sectors for transmission. Again, compared with the regular routing procedure, spare sectors of forwarding nodes on the routes are eliminated and network coding combines the transmitted messages potentially.

5.5 Mobility Handling

The proposed approach aims at reducing redundancy in the broadcasting procedure via restricting transmission directions and merging transmitted packages. Hence, energy consumption, as well as signal interference, can be decreased. However, when node movement is introduced in the network, the delivery ratio of the proposed approach will be smaller than that of regular broadcast approaches without directional antennas or coding. The detailed comparison is shown in the next section. One solution is to introduce a controlled transmission redundancy to improve the delivery ratio in the existence of node mobility. For instance, when the delivery ratio is lower than a threshold value, a node adds one more section for transmission, which is adjacent to the resultant transmission sections generated by EBCD, to tolerate a certain degree of node movement. The energy efficiency and delivery ratio is a trade-off. Note that this scheme efficiently improves the delivery ratio since no ACK/NACK information exchange is needed.

6 Simulation

In this section, we evaluate the proposed EBCD algorithm by comparing the total energy consumption in terms of the number of message transmissions in the network and also the size of sectors that the message transmitted. We compare EBCD with two algorithms: 1) the algorithm without network coding or directional antennas (CDS). We simply call it algorithm CDS since the forwarding node set selected by this method is a source-based CDS, which means that together with the source node, the forwarding node set forms a CDS for the network. We use a dynamic node coverage condition as used in our EBCD. 2) The algorithm with network coding, but without directional antennas (Coding). This is the approach proposed in [13], but in order

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

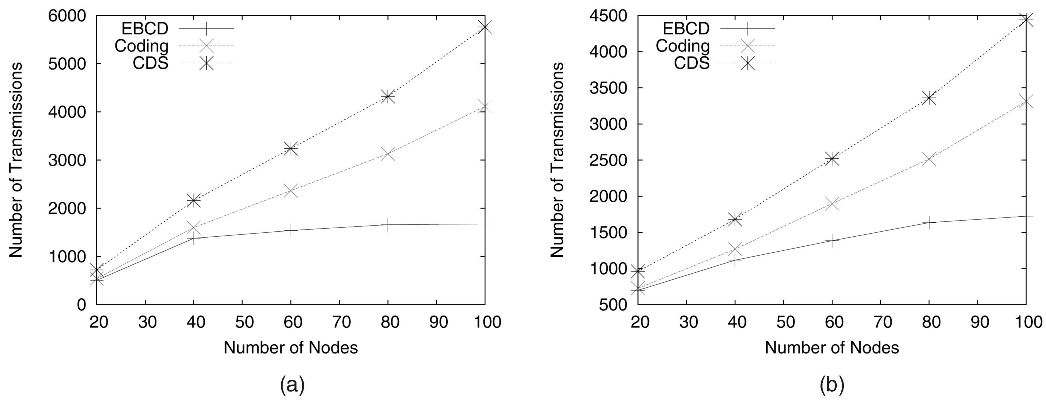


Fig. 10. Comparison of EBCD, Coding, and CDS in number of transmissions. (a) Dense network ($d = 18$). (b) Sparse network ($d = 6$).

to make a fair comparison, the underlying forwarding node selection approach we use in Coding is also the dynamic node coverage condition. We also compare EBCD with the proposed static EBCD (S-EBCD) to check the performance variation. Then, the performance of the two approaches for the single source/single message broadcast application is evaluated. These approaches are the PB approach and SO approach. We also evaluate the performance of these algorithms in a dynamic environment to determine the affects of mobility and signal interference. The number of trials for each tunable parameter is 100.

6.1 Simulation Environment

In the simulation, n nodes are randomly placed in a restricted 100×100-area and networks that cannot form a strongly connected graph are discarded. The tunable parameters in this simulation are as follows:

1. The number of nodes n: We vary the number of deployed nodes from 20 to 100 to check the scalability of the algorithms.
2. The average node degree d, which represents the density of the network: We use 6 and 18 as the values of d to generate sparse and dense networks.
3. The number of sectors of the antenna pattern K. We use 4 and 6 as the values of K.
4. The number of broadcast sessions b, i.e., the number of generated broadcasting messages: b has a fixed value of 20 in the simulation. Therefore, when n is different, we can simulate varied data loads in the network. The source nodes are randomly selected.
5. The number of segments k in the PB and SO extensions: k is 10 and 20 in the network.

In the dynamic environment, we use several additional parameters as follows:

6. The maximal forward jitter delay j, which varies from 0.01 to 100 ms.
7. The average moving speed v: When there is mobility, the average moving speed is varied from 1 to 25 m/s.

The following metrics are compared: 1) the number of transmissions in the application (we assume that the transmission of a message, original or coded, following a transmission edge, is one transmission); 2) the average energy consumption for a broadcast message (we assume that one transmission of a original broadcast message or a coded message in each sector consumes one unit of energy); and 3) the delivery ratio of a broadcast message in the dynamic environment.

6.2 Simulation Results in a Static Environment

We use a customer simulator to perform the following simulation without considering node mobility and signal interference. Fig. 10 shows the comparison of EBCD, Coding, and CDS in the number of transmissions in both dense and sparse networks. Fig. 10a is the dense network, where the average node degree is 18. We can see that Coding can reduce the number of message transmissions with a reduction rate of around 1.2. EBCD can further reduce it significantly. When the number of nodes increases, the number of transmissions in EBCD tends to be stable. Fig. 10b shows that the average node degree is 6. EBCD can still reduce the number of transmissions compared with CDS or Coding. But, the reduction rate is lower than that in the dense network.

Fig. 11 shows the comparison of EBCD, Coding, and CDS, in terms of energy consumption, when K is 4 and 6. Figs. 11a and 11b are in dense networks. We can see that EBCD can further reduce the number of switched-on sectors compared with CDS and Coding, in which all sectors of a forwarding node need to be switched on for transmission. When K is larger, the reduction rate of EBCD over CDS and Coding, is more significant since a larger forwarding area can be pruned. Figs. 11c and 11d are in the sparse networks. EBCD also reduces the number of switched-on sectors significantly. The larger the value of K, the larger the reduction rate.

Fig. 12 shows the comparison of EBCD and S-EBCD in terms of the number of forwarding nodes and transmissions in both dense and sparse networks. We can see that although the number of forwarding nodes selected in the static method should be larger than that in the dynamic one, as shown in Figs. 12a and 12b, the final numbers of transmissions in EBCD and S-EBCD are very close, especially when the network is relatively dense. This is because more forwarding nodes increase the probability of network coding, which makes up for the larger forwarding node set. The forwarding node set of S-EBCD is around 1.3 times larger than that of EBCD, while the final number of transmissions is 1.03 times higher. The advantage of S-EBCD is that it only calculates the status of each node once for any broadcast message from any source. It is also unnecessary to piggyback the broadcast state information. Therefore, if the network is dense, S-EBCD is preferred

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

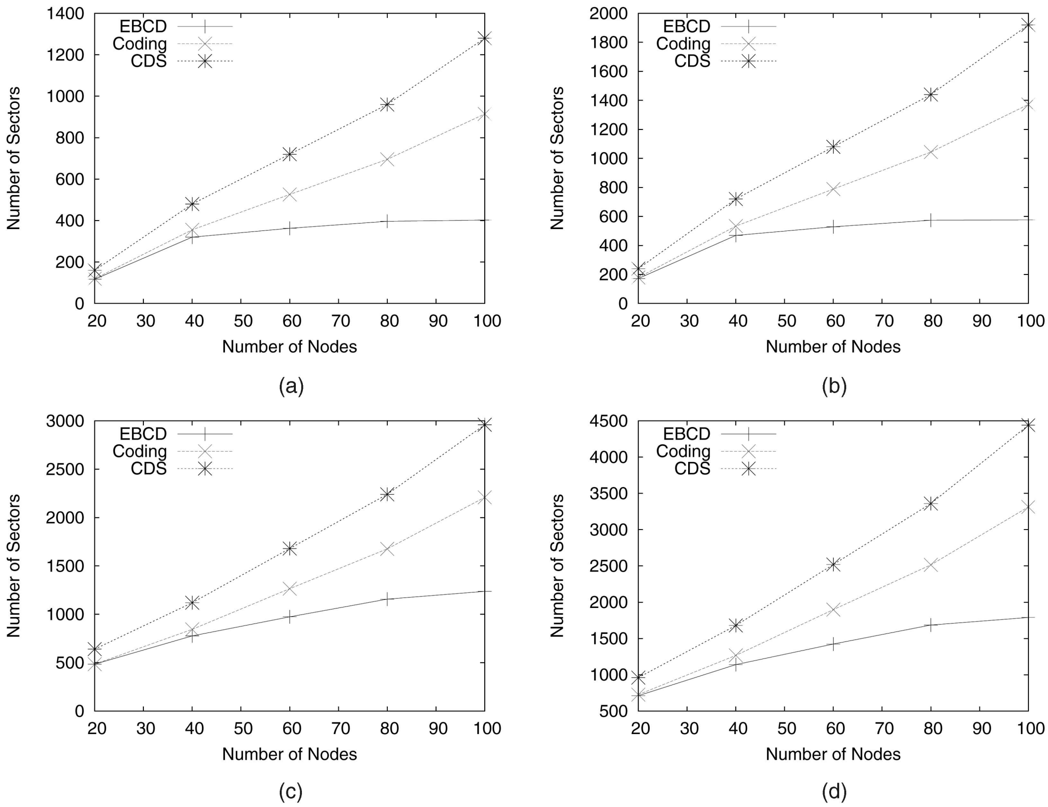


Fig. 11. Comparison of EBCD, Coding, and CDS in energy consumption. (a) Dense network ($K = 4$). (b) Dense network ($K = 6$). (c) Sparse network ($K = 4$). (d) Sparse network ($K = 6$).

since the overhead of it is smaller, while the performance is comparative.

Fig. 13 shows the performance evaluations of the two extensions of EBCD, PB, and SO with different segment numbers $k = 10, 20$. We can see that in Fig. 13a, when the network is dense, which means that the transmission range is larger, SO has better performance than PB. Smaller k makes the advantage of SO over PB more significant. When k is 20, SO is very close to PB, because the number of transmissions is larger with larger numbers of segments in the initial phase of SO. In Fig. 13b, the results in the sparse network are shown. When k is 10, SO is better than PB. When k is 20, PB is even better than SO. This is because the initial phase of SO has a lot of overhead in the case of larger k and smaller transmission range, which leads to more hops spreading out the segments.

6.3 Simulation Results in a Dynamic Environment

In this section, we simulate EBCD, Coding, and CDS in a dynamic environment to determine the affect of node mobility and signal interference on these algorithms. We use *ns2.1b9*, the directional antenna model, and an enhanced IEEE 802.11 MAC layer provided by the enhanced network simulator (TeNs) [22]. The nodes share a single 2 MB channel and the traffic load is 1-10 packets per second (pps) with a packet size of 64 bytes. The "Hello" message interval is 1 s. We use the random waypoint mobility model [18].

Fig. 14 shows the simulation results in a dynamic environment ($K = 4, n = 100$). Figs. 14a and 14b are the number of sectors and the delivery ratio when there is signal interference in the network ($v = 0$). In Fig. 14a, the number of switched-on sectors of EBCD is the smallest among the three approaches. The advantage of EBCD over the other two approaches is more significant than in the static network. This is because in EBCD, the directional transmission and decreased number of transmissions reduce the signal interference. In order to reduce the collisions caused by the directional hidden terminal problem, we can use a random forward jitter delay in the simulation, which is within range $[0, j_{max}]$. We can see that with the increasing of the jitter delay, the number of switched-on sectors of CDS decreases. However, the numbers of switched-on sectors of EBCD and Coding increase as the jitter delay increases, because the delay in the network coding procedure makes the EBCD and Coding approaches less effective.

Fig. 14b is the delivery ratio in the presence of interference. We can see that as the jitter delay increases, so does the delivery ratio of EBCD, Coding, and CDS. When the jitter delay is large enough, the delivery ratio of CDS is larger than that of Coding, but still smaller than EBCD. This is because directional transmission in EBCD helps to reduce the collision in transmission.

Figs. 14c and 14d are the delivery ratios when there is node movement in the network in dense and sparse networks, respectively, ($j = 0.1$). In Fig. 14c, the delivery ratio of the three approaches decreases as the average moving speed increases. EBCD has the worst performance when the speed is large enough, although it has the best performance in the static network. The directional transmission has small redundancy, which can help with

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

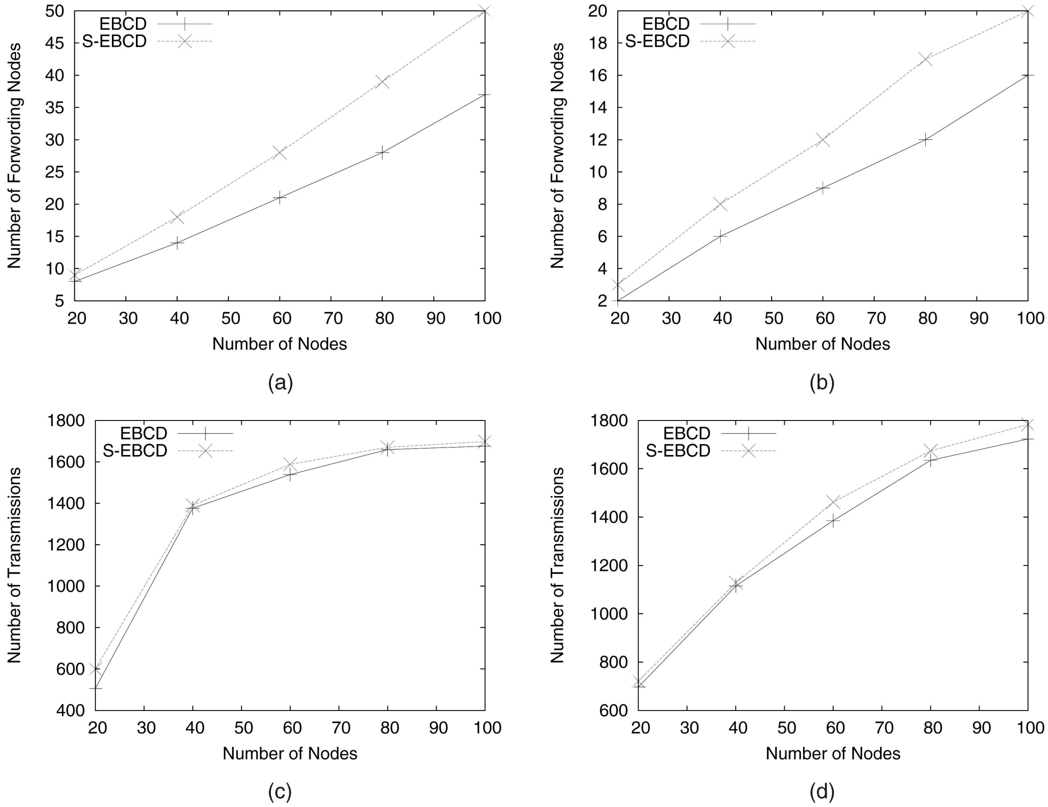


Fig. 12. Comparison of EBCD and S-EBCD. (a) Forwarding nodes ($d = 18$). (b) Forwarding nodes ($d = 6$). (c) Number of transmissions ($d = 18$). (d) Number of transmissions ($d = 6$).

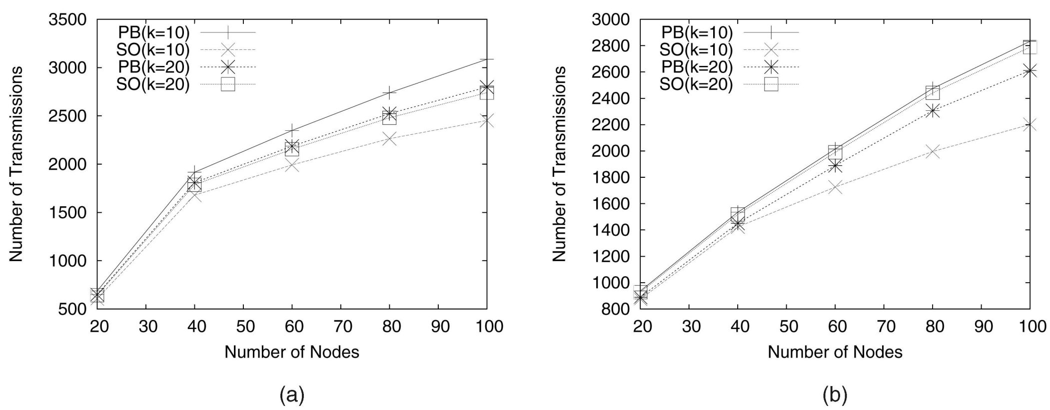


Fig. 13. Comparison of PB and SO in number of transmissions. (a) Dense network ($d = 18$). (b) Sparse network ($d = 6$).

the broadcast procedure in the presence of node mobility. Fig. 14d has curves that are similar to Fig. 14c, but with a larger delivery ratio. This is because in sparse networks, the signal interference is less significant.

6.4 Simulation Summary

In this section, we use simulations to verify the performance of the proposed EBCD approach by comparing with two other solutions for broadcast backbone construction in MANETs. We use the number of total transmissions, the average energy consumption for a broadcast, and the delivery ratio of the broadcast message as the measurement metrics, which represent the effectiveness and efficiency of the constructed backbone. We also simulate both the static and dynamic environments to test the robustness of the approaches.

The simulation results can be summarized as follows:

1. EBCD has significant performance improvement in terms of the number of transmissions compared with CDS and Coding, especially in relatively dense networks.
2. EBCD has better performance than CDS and Coding in terms of the number of switched-on sectors, which corresponds to the energy consumption. The larger the value of K, the larger the reduction rate of EBCD over the other two methods.
3. S-EBCD has very close performance to EBCD, especially when the network is relatively dense. Therefore, due to its other advantage, such as less overhead, S-EBCD is another option.

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

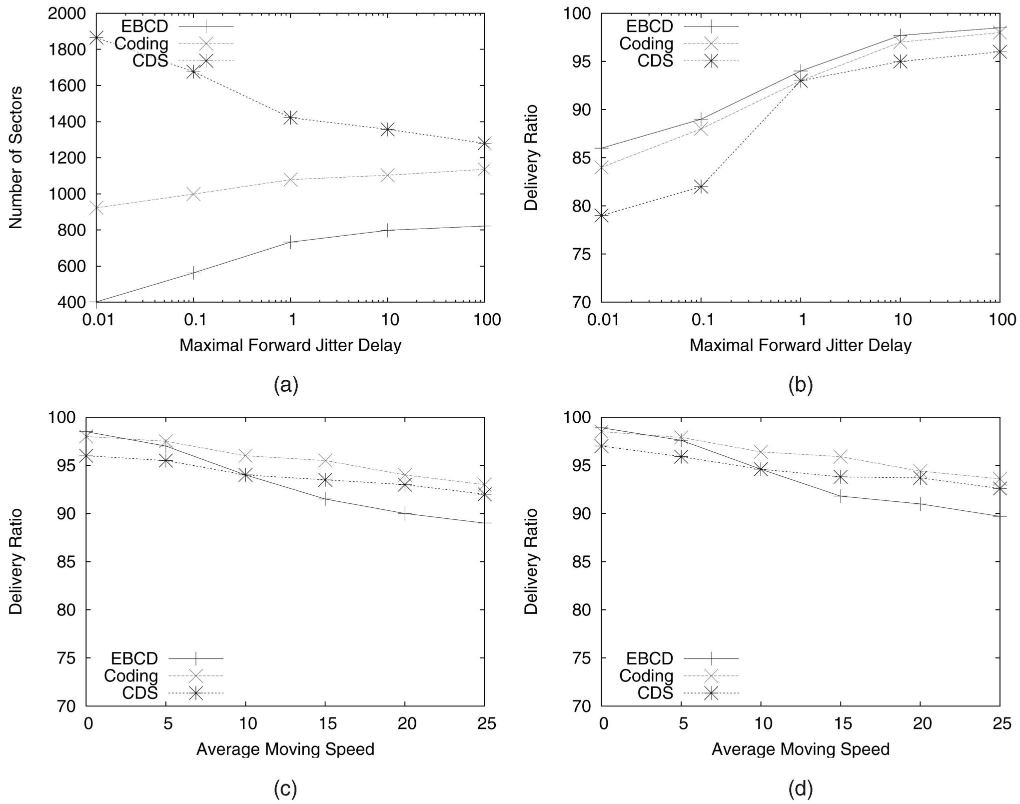


Fig. 14. Performance in a dynamic environment with collision and mobility ($K = 4, n = 100$). (a) Number of sectors ($v = 0$). (b) Delivery ratio ($v = 0$). (c) Delivery ratio in dense network. (d) Delivery ratio in a sparse network.

4. SO has better performance than PB when the network is relatively dense and the number of segments a message is divided into is small. When in sparse networks with large k, PB performs even better than SO.
5. In dynamic environments, the directional transmission of EBCD helps to reduce the effect of signal collision, but harms the performance in the presence of node mobility due to the fact that there is less redundancy.

7 Conclusions

Network coding has been exploited for efficient broadcasting to reduce the number of transmissions in the multiple source broadcast application. We combine the network coding-based broadcast approach with broadcasting using directional antennas for a more efficient broadcast strategy, and develop efficient broadcasting using network coding and directional antenna algorithm (EBCD). We extend existing broadcasting using the directional antenna approach to a dynamic mode. Although the coding-based approach is independent of the underlying forwarding node selection procedure, we show that different forwarding node selections affect the overall performance significantly. We also discuss the single source/single message application and design two approaches for it. Performance analysis is conducted through simulations. The proposed EBCD approach has better performance than traditional CDS-based broadcasting and the existing network coding-based broadcasting in terms of energy consumption. Also, the static version of EBCD has comparative performance in energy conservation with smaller overhead. In a dynamic environment with signal interference and node mobility, EBCD has better performance against collision, but worse performance when it comes to movement. In the future, we will improve the robustness of mobility for the proposed approaches.

Acknowledgments

This work was supported in part by the US National Science Foundation grants CNS 0422762, CNS 0434533, CNS 0531410, CNS 0626240, CNS 0948184, CCF 0830289, and CCF 0943452.

References

[1] R. Ahlswede, N. Cai, S.R. Li, and R.W. Yeung, "Network Information Flow," *IEEE Trans. Information Theory,* vol. 46, no. 4, pp. 1204-1216, July 2000.

[2] D. Bein and S.Q. Zheng, "An Effective Algorithm for Computing Energy-Efficient Broadcasting Trees in All-Wireless Networks," *Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS),* 2008.

[3] P. Chaporkar and A. Proutiere, "Adaptive Network Coding and Scheduling for Maximizing Throughput in Wireless Networks," *Proc. ACM MobiCom,* 2007.

[4] F. Dai and J. Wu, "Efficient Broadcasting in Ad Hoc Wireless Networks Using Directional Antennas," *IEEE Trans. Parallel and Distributed Systems,* vol. 17, no. 4, pp. 335-347, Apr. 2006.

[5] F. Delgosha, E. Ayday, K. Chan, and F. Fekri, "Security Services in Wireless Sensor Networks Using Sparse Random Coding," *Proc. IEEE INFOCOM,* 2007.

[6] A. Dimakis, B. Godfrey, M. Wainwright, and K. Ramchandran, "Network Coding for Distributed Storage in Peer-to-Peer Networks," *Proc. IEEE INFOCOM,* 2007.

[7] C. Fragouli, J. Widmer, and J.-Y.L. Boudec, "A Network Coding Approach to Energy Efficient Broadcasting: From Theory to Practice," *Proc. IEEE INFOCOM,* 2006.

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.

[8] S. Gollakota and D. Katabi, "ZigZag Decoding: Combining Hidden Terminals in Wireless Networks," *Proc. ACM SIGCOMM,* 2008.

[9] P. Gupta and P.R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. Information Theory,* vol. 46, no. 2, pp. 388-404, Mar. 2000.

[10] C. Hu, Y. Hong, and J. Hou, "On Mitigating the Broadcast Storm Problem with Directional Antennas," *Proc. IEEE Int'l Conf. Comm. (ICC),* 2003.

[11] S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, and M. Médard, "Resilient Network Coding in the Presence of Byzantine Adversaries," *Proc. IEEE INFOCOM,* 2007.

[12] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the Air: Practical Wireless Network Coding," *Proc. ACM SIGCOMM,* 2006.

[13] L. Li, R. Ramjee, M. Buddhikot, and S. Miller, "Network Coding-Based Broadcast in Mobile Ad Hoc Networks," *Proc. IEEE INFOCOM,* 2007.

[14] X.Y. Li, Y. Wu, S.J. Tang, and X.H. Xu, "Broadcast Capacity for Wireless Ad Hoc Networks," *Proc. IEEE Int'l Conf. Mobile Ad-Hoc and Sensor Systems (MASS),* 2008.

[15] J. Liu, D. Goeckel, and D. Towsley, "Bounds on the Gain of Network Coding and Broadcasting in Wireless Networks," *Proc. IEEE INFOCOM,* 2007.

[16] W. Lou and J. Wu, "On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks," *IEEE Trans. Mobile Computing,* vol. 1, no. 2, pp. 111-122, Apr.-June 2002.

[17] A. Mohammed, M. Ould-Khaoua, L.M. Mackenzie, and J. Abdulai, "An Adjusted Counter-Based Broadcast Scheme for Mobile Ad Hoc Networks," *Proc. 10th Int'l Conf. Computer Modeling and Simulation (UKSIM),* 2008.

[18] W. Navidi and T. Camp, "Stationary Distributions for the Random Waypoint Mobility Model," *IEEE Trans. Mobile Computing,* vol. 3, no. 1, pp. 99-108, Jan./Feb. 2004.

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

[20] S. Pleisch, M. Balakrishnan, K. Birman, and R. Renesse, "MISTRAL: Efficient Flooding in Mobile Ad-Hoc Networks," *Proc. ACM MobiHoc,* 2006.

[21] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint Relaying for Flooding Broadcast Message in Mobile Wireless Networks," *Proc. 35th Hawaii Int'l Conf. System Sciences (HICSS-35),* 2002.

[22] S. Roy and A. Kumar, "Realistic Support for IEEE 802.11b MAC in NS," bachelor thesis, Indian Inst. of Technology, Kanpur, 2004.

[23] S. Sengupta, S. Rayanchu, and S. Banerjee, "An Analysis of Wireless Network Coding for Unicast Sessions: The Case for Coding-Aware Routing," *Proc. IEEE INFOCOM,* 2007.

[24] C.C. Shen, Z. Huang, and C. Jaikaeo, "Directional Broadcast for Ad Hoc Networks with Percolation Theory," technical report, Computer and Information Sciences, Univ. of Delaware, 2004.

[25] D. Simplot-Ryl, J. Cartigny, and I. Stojmenovic, "An Adaptive Localized Scheme for Energy Efficient Broadcasting in Ad Hoc Networks with Directional Antennas," *Proc. Ninth IFIP Conf. Personal Wireless Comm. (PWC),* 2004.

[26] L. Song and D. Hatzinakos, "Broadcasting Energy Efficiency Limits in Wireless Networks," *IEEE Trans. Wireless Comm.,* vol. 7, no. 7, pp. 2502-2511, July 2008.

[27] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating Sets and Neighbor Elimination Based Broadcasting Algorithms in Wireless Networks," *IEEE Trans. Parallel and Distributed Systems,* vol. 13, no. 1, pp. 14-25, Jan. 2002.

[28] 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.

[29] Y.C. Tseng, S.Y. Ni, Y.S. Chen, and J.P. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Wireless Networks,* vol. 8, nos. 2/3, pp. 153-167, 2002.

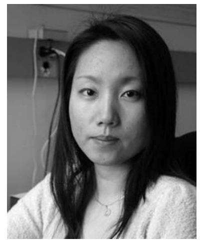
[30] M. Wang and B. Li, "Lava: A Reality Check of Network Coding in Peer-To-Peer Live Streaming," *Proc. IEEE INFOCOM*, 2007.

[31] J. Wu and F. Dai, "A Generic Distributed Broadcast Scheme in Ad Hoc Wireless Networks," *IEEE Trans. Computers*, vol. 53, no. 10, pp. 1343-1354, Oct. 2004.

[32] J. Wu and H. Li, "On Calculating Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks," *Proc. ACM Workshop Discrete Algorithms and Methods for Mobile Computing and Comm. (DIALM)*, 1999.

[33] S. Yang, J. Wu, and F. Dai, "Efficient Backbone Construction Methods in Manets Using Directional Antennas," *Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS),* 2007.

[34] D. Yuan, J. Bauer, and D. Haugland, "Minimum-Energy Broadcast and Multicast in Wireless Networks: An Integer Programming Approach and Improved Heuristics Algorithms," *Ad Hoc Networks,* vol. 6, no. 5, pp. 696-717, 2008.



Shuhui Yang received the BS and MS degrees in 2000 and 2003, respectively, from Jiangsu University, Zhenjiang and Nanjing University, Nanjing, China, and the PhD degree in computer science from Florida Atlantic University in 2007. She is an assistant professor in the Department of Math, Computer Science, and Statistics, Purdue University Calumet. Her current research focuses on the design of localized routing algorithms in wireless ad hoc and sensor networks and distributed systems. She is a member of the IEEE and the IEEE Computer Society.

Jie Wu is chair and professor in the Department of Computer and Information Sciences, Temple University. Prior to joining Temple University, he was a program director at the US National Science Foundation. His research interests include wireless networks and mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. He has published more than 450 papers in various journals and conference proceedings. He serves on the editorial board of the *IEEE Transactions on Mobile Computing*. Dr. Wu was also general cochair for IEEE MASS 2006, IEEE IPDPS 2008, and DCOSS 2009. He has served as an IEEE Computer Society distinguished visitor and is the chairman of the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a fellow of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: Temple University. Downloaded on August 11,2010 at 15:52:34 UTC from IEEE Xplore. Restrictions apply.