Scalable Computing: Practice and Experience
Volume 11, Number 1, pp. 43–52. http://www.scpe.org
ISSN 1895-1767

EFFICIENT BROADCASTING IN MANETS BY SELECTIVE FORWARDING

DOINA BEIN[*], AJOY K. DATTA[†] AND BALAJI ASHOK SATHYANARAYANAN[†]

Abstract. A major challenge faced in mobile ad hoc networks (MANETs) is locating devices for communication, especially in the case of high node mobility and sparse node density. Present solutions provided by the ad hoc routing protocols range from flooding the entire network with route requests, to deploying a separate location management scheme to maintain a device location database. Many applications as well as various unicast routing protocols such as Dynamic Source Routing (DSR), Ad Hoc On Demand Distance Vector (AODV), Zone Routing Protocol (ZRP), and Location Aided Routing (LAR) use broadcasting or a derivation of it. Flooding is expensive in terms of overhead and wastes valuable resources such as bandwidth and power. We propose to develop a strategy to reduce the redundant transmission of packets in normal flooding used in broadcasting and we describe strategies for choosing only a minimal set of nodes to re-broadcast in grid networks. Our strategies reduce the redundant transmission of packets, thus packets are forwarded with a minimal number of transmissions. To determine the minimal set of nodes we propose a new algorithm called Efficient Broadcasting by Selective Forwarding (EBSF) that uses a distance-based approach in selecting the nodes among all the nodes in a grid network. The distance-based approach is implemented for broadcast and rebroadcast to a set of nodes with the help of a threshold value that reduces the number of redundant transmission. This threshold value can be tuned to show the performance enhancement.

Key words: broadcast, distance-based approach, grid network, mobile network, routing

1. Introduction. A mobile ad hoc network (MANET) is an autonomous system of mobile routers, connected by wireless links, the union of which forming an arbitrary graph (see Figure 1.1). The routers are free to move and organize themselves arbitrarily. Thus, the topology may change rapidly and unpredictably. Such a network may operate in a stand-alone fashion or may be connected to the larger network. In mobile ad hoc networks, it is often necessary to broadcast control information to all constituent nodes in the network. Blind flooding [14] is often deployed to achieve the above objective. Its advantages, the simplicity and the guarantee that every destination in the network is reached, are downsized by the fact that it is expensive in terms of overhead and wastes valuable resources such as bandwidth and power:

- Some routers receive a packet multiple times.
- It leads to transmission of redundant packets.
- Packets can go in a loop forever.
- For dense networks, it causes significant contention and collisions—the so-called *broadcast storm problem*.

Broadcasting, the process by which one node needs to communicate a packet to all other nodes in the network, is a critical mechanism for information diffusion and maintaining consistent global network information. Additionally, it is an energy intensive function. Broadcasting or a derivation of it is used by routing protocols such as Dynamic Source Routing (DSR) [11], Ad Hoc On Demand Distance Vector (AODV) [21], Location Aided Routing (LAR) [13, 15], and Zone Routing Protocol (ZRP) [9], for establishing routing paths. Currently all these protocols rely on a simple form of broadcasting called *blind flooding*. In blind flooding, each node re-broadcast a packet whenever it receives it for the first time: Every incoming packet is sent out on every outgoing line except the one it arrived on. Blind flooding generates many redundant transmissions, which may cause a more serious broadcast storm problem [17]. Given the expensive and limited nature of wireless resources such as bandwidth and battery power, minimizing the control message overhead for route discovery is a high priority in protocol design. Recently, a number of research groups have proposed more efficient broadcasting techniques. Centralized broadcasting schemes are presented in [1, 4, 8]. The algorithms in [16, 25, 19, 20, 22, 23] utilize neighborhood information to reduce the number of redundant messages.

In this paper we present a new protocol called *Efficient Broadcasting by Selective Forwarding* (EBSF) that minimizes the number of transmissions or retransmissions needed for broadcasting using a distance-based approach [2]. In our proposed protocol, only a set of selected nodes are allowed to do the broadcast; these nodes are selected using a so called *threshold distance* [18, 3]. We set the threshold distance to be

$threshold = n \times transmission\_radius$, where n is a real number. This threshold value can be tuned to show performance enhancement, thus minimizing the number of transmissions or retransmissions needed. If

[*]Applied Research Laboratory, The Pennsylvania State University, University Park, PA 16802, USA (siona@psu.edu).
[†]School of Computer Science, University of Nevada, Las Vegas, NV 89154, USA (datta@cs.unlv.edu, ashoksathyan@yahoo.com).

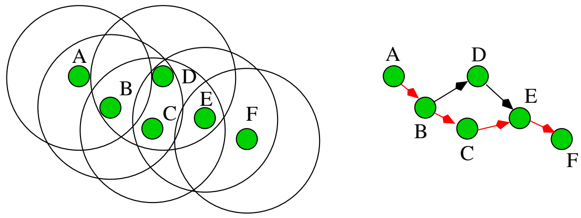

Fig. 1.1. *Example of MANET*

the threshold distance is set to 0, selective broadcasting will be exactly blind flooding. To check for message duplication, thus to reduce the number of redundant messages to be delivered, we implement a new data structure called *message cache*.

We have simulated the communication network using a network simulator called GLOMOSIM [5, 6, 7]. GLOMOSIM (GLObal MObile System SIMulator) is a library-based sequential and parallel simulator for wireless networks. The library of GLOMOSIM is based on the parallel discrete-event simulation capability provided by Parsec, the compiler used in GLOMOSIM. Using GLOMOSIM we show the performance variation of our algorithm using different threshold factors.

In Section 2 we present an overview of various broadcasting schemes followed by the motivation of why blind flooding is not efficient for broadcasting in MANETs. In Section 3 we define efficient broadcasting, followed by a description of our protocol and its advantages. A discussion regarding the simulation results and the performance analysis is given in Section 4. We finish with concluding remarks and present concepts for future research in Section 5.

2. Preliminaries. Broadcasting is a communication paradigm that allows sending data packets from source to multiple receivers. Broadcasting in wireless ad-hoc networks is a critical mechanism for applications such as information diffusion, wireless networks and also for maintaining consistent global network information. Broadcasting or a derivation of it is often necessary in MANET routing protocols such as Dynamic source routing (DSR), Ad Hoc On Demand Distance Vector (AODV), Location Aided Routing (LAR), Zone Routing Protocol (ZRP) for establishing routes. Currently all these protocols rely on a simplistic form of broadcasting called *flooding*. Flooding is a process in which each node (or all nodes in a localized area) retransmits each received unique packet exactly one time. In flooding, every incoming packet is sent out on every outgoing line except the one it arrived on. Whenever a device connected to the LAN (Local Area Network) switch sends a packet to an address that is not in the LAN switchs table or whenever the device sends a broadcast or multicast packet, the switch sends the packet out to all ports. This is referred to as *blind flooding*. The routers forward packets to all ports except the ingress port. There are two advantages to blind flooding: simplicity and the fact that every node in the network is eventually reached. But there are many disadvantages: receiving the same packet multiple times that leads to redundant transmissions, packets can go in a loop forever, inefficient use of bandwidth, high power consumption.

An improvement to blind flooding is to choose only a subset of nodes to re-broadcast and in this manner to reduce the number of data transmissions. Several alternatives are presented next. The probabilistic scheme [23, 24, 10, 12] is similar to blind flooding, except that nodes only re-broadcast with a predefined probability. Since some nodes do not re-broadcast, node and network resources are saved without having delivery effectiveness. In sparse networks, nodes will not receive all broadcast packets unless the probability parameter is high. When the probability is 100%, this scheme is identical to blind flooding.

There is an inverse relationship between the number of times a packet is received at a node and the probability of that node being able to reach some additional area on a broadcast. This result is the basis of the counter-based scheme [17, 23, 24]. Upon reception of a packet never received before, the node initiates a counter with value 1 and sets a random assessment delay (RAD) timeout. During the RAD, the counter is incremented for each redundant packet received. If the counter is less than a threshold value when the RAD expires, then the packet is re-broadcast, otherwise the packet is simply dropped.

The probabilistic and counter-based schemes are simple and inherently adaptive to the local topologies. Their disadvantage is that the delivery is not guaranteed to all nodes even if the ideal Media Access Control (MAC) layer is provided. (The MAC layer at a node provides addressing and access control mechanisms of the

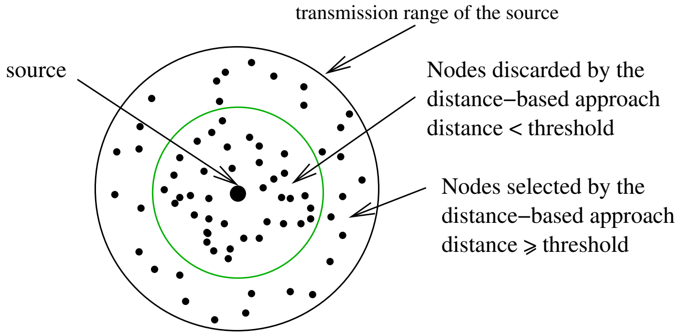

Fig. 3.1. *Efficient Broadcasting by Selective Forwarding*

communication channels of the node that allows the node to send and receive packets in a network.) In other words, both schemes are unreliable.

Area-based methods [24] only consider the coverage area of transmission and do not consider whether there are nodes within that area. A node decides whether to re-broadcast purely based on its own information. There are two coverage area-based methods [23]: distance-based and location-based schemes. In this paper we have developed a new approach based on the physical distance [23] by which the data transmitted is reduced considerably to a greater extent, while ensuring that all data is received by all nodes in the network. In our approach, broadcasting is done by some selected nodes that are privileged to broadcast based on so-called *threshold distance*. If the threshold distance is set to 0 then the network broadcast becomes blind flooding.

A uniform distribution of nodes is an effective way to organize a large network. Flooding provides important control and route establishment functionality for a number of unicast and multicast protocols in ad hoc networks. Considering its wide use as a building layer for other network layer protocols, the flooding method should deliver a packet from one node to all other network nodes using as few messages as possible, including control information. This makes the network wide broadcasting an energy intensive function. An improvement to flooding is to choose only a subset of nodes to rebroadcast and thus reduce data transmissions. To this end, we have devised a mechanism where an optimal node selection is done to determine the minimal set of nodes for broadcast in mobile ad hoc networks.

3. Efficient Broadcasting by Selective Forwarding. EBSF reduces the redundant transmission of packets used in broadcasting. An optimal selection is done to determine the minimal set of nodes to re-broadcast. To determine this minimal set of nodes we employ the distance-based approach [23]. We assume that the nodes are placed on a grid and all nodes have the same transmission range. We also assume that the transmission range is constant. The distance-based approach is implemented with the help of a threshold value that is taken to be $threshold = N \times transmission\_radius$, where N has values $0 < N < 1$.

In blind flooding, location discovery is achieved by allowing all nodes that receive some message to retransmit it. In our case we prevent any node within a distance factor which is the threshold to re-transmit. Thus a few selected nodes will be selected for retransmission, and subsequently we reduce the number of redundant transmissions (see Figure 3.1). This threshold value can be tuned as per requirements. It can be increased and decreased to show the performance variation. Higher the threshold, lesser is the number of packets transmitted by each node in the network and vice versa. If threshold is set to 0, then all packets are transmitted like in blind flooding (no enhancement). The advantages of EBSF are: the selection of optimal number of nodes to retransmit, an effective utilization of bandwidth, minimizing the number of unnecessary transmission and therefore reducing the redundant packets, and minimizing the power consumption. We made the following assumptions:

1. The nodes have uniform transmission power (they have the same transmission radius).
2. The nodes are placed in a grid-like topology.
3. The nodes are almost uniformly distributed in the network.

We show next where our proposed EBSF protocol is executed within the OSI layers.

In the OSI model, upper layers deal with the application's issues and generally are implemented only in software. The highest layer, the application layer, is closest to the end user. The lower layers of the OSI model handle data transport issues, namely communications between network devices. When a packet is to be

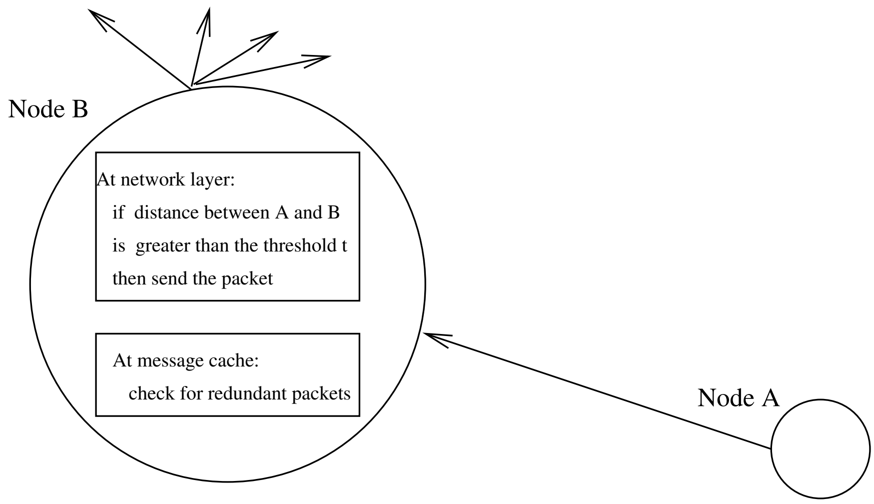

Fig. 3.2. *EBSF protocol*

transmitted from a source node to a destination node, the network layer (Layer 3) is responsible for end-to-end (source to destination) packet delivery, whereas the data link layer (Layer 2) is responsible for node-to-node (hop-to-hop) packet delivery. When a node receives a packet, the MAC layer of the data link layer sends the packet to the network layer for transmission. Efficient Broadcasting by Selective Forwarding (EBSF) is implemented in the network layer.

When a packet is forwarded to the network layer for transmission, the IP header in the Internet Protocol has the source node and the destination node coordinates. Based on these coordinates, the distance between the source and destination nodes cam be easily computed. In blind flooding, whenever a message is received and needs to be sent further, at the network layer the packet is broadcast to all the nodes within its transmission range once it is received from the MAC layer.

In EBSF protocol, when a packet reaches the network layer of a node for broadcasting some computation is performed as follows (see Figure 3.2). Based on the information about the source and the destination taken from the IP header, EBSF computes the distance between the source and the destination. When this distance is less than the threshold distance, the node discards the message. If this distance is greater than the threshold distance, then EBSF broadcasts the message to the nodes at a distance equal to or greater than the threshold distance.

When a node A broadcasts p packets to other nodes in the network, each node has a message cache, called MSGCACHE, which stores information about messages the node has received. When a node receives a packet, it checks with the MSGCACHE whether the packet has already been received. In affirmative, the packet is discarded; otherwise, EBSF forwards it to the network layer for transmission. In this manner, the redundant packets are discarded. MSGCACHE is implemented as a linked-list data structure. In EBSF, at any node every packet is checked for message redundancy using MSGCACHE and then send it to the network layer for broadcast. The network layer will decide, based on the threshold distance between the source node and destination nodes, whether to broadcast or not, as follows. Based on the information about the source and the destination in the IP header, the distance between the source node and the destination node is calculated. When this distance is less than the threshold distance, the node discards the message. If this distance is greater than the threshold distance, the node will broadcast the message to the nodes at a distance equal or greater than the threshold distance. This process is repeated until all the nodes in the network receive all the packets originated from the source node.

For further enhancement, we partition the neighbors of a node into two types, *communicational neighbors* and *geographical neighbors*. Communicational neighbors are the ones within the node's transmission range.Geographical neighbors of a node in a grid network are the nodes located at a distance of one grid unit: a corner node has two geographical neighbors and a border node has three geographical neighbors; all other nodes have four geographical neighbors. For example, in Figure 3.3, the geographical neighbors of node E are nodes B, D, F, H. The corner node G has two geographical neighbors, nodes D and H. The border node H has three geographical neighbors G, E, I.

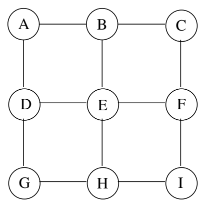

Fig. 3.3. *Geographical Nodes*

Table 4.1
Layers of GLOMOSIM

Layers	**Protocols**
Mobility	Random waypoint, Random drunken, Trace based
Radio Propagation	Two ray and Free space
Radio Model	Noise Accumulating
Packet Reception Models	SNR bounded, BER based with BPSK / QPSK modulation
Data Link (MAC)	CSMA, IEEE 802.11 and MACA
Networking (Routing)	IP with AODV, Bellman-Ford, DSR, Fisheye, LAR scheme 1, ODMRP
Transport	TCP and UDP
Application	CBR, FTP, HTTP and Telnet

We have enhanced EBSF to exclude to broadcast to the geographical neighbors of a node situated within the threshold distance, and thereby decreasing the number of transmissions. For example, given a sender node S, if some node R is at a distance greater than or equal to the threshold distance of node S, then more likely the geographical nodes of node R are also at threshold distance from node S. In this case, in EBSF node R and the geographical neighbors of R are allowed to broadcast the message received from node S. When node R and its geographical neighbors may have common nodes which are at a threshold distance from them, these nodes will receive the same messages from node R and its geographical neighbors, which would lead to redundant transmissions. Hence, with the enhanced EBSF algorithm we restrict the geographical neighbors of a node to not broadcast and thereby reduce redundant transmission, which is evident from the simulation results.

4. Simulations. The tool we have used for the simulation is GLOMOSIM. GLOMOSIM is a scalable simulation environment for wireless and wired systems, designed using the parallel discrete-event simulation capability provided by Parsec. Parsec compiler is similar to the C compiler with some added functions. The simulation statistics are stored in the “bin” directory of GLOMOSIM as “glomo.stat” file. The layers in GLOMOSIM are shown in Table 1.

GLOMOSIM contains the following directories: `application` contains code for the application layer, `bin` for executable and input or output files, `doc` contains the documentation, `include` contains common include files, `java gui` contains the visual tool, `Mac` contains the code for the MAC layer, `main` contains the basic framework design, `network` contains the code for the network layer, `radio` contains the code for the radio layer, `scenarios` contains some example scenarios, `tcplib` contains libraries for TCP, and `transport` contains the code for the transport layer.

In Figure 4.1 we present the files that we have been modified in GLOMOSIM.

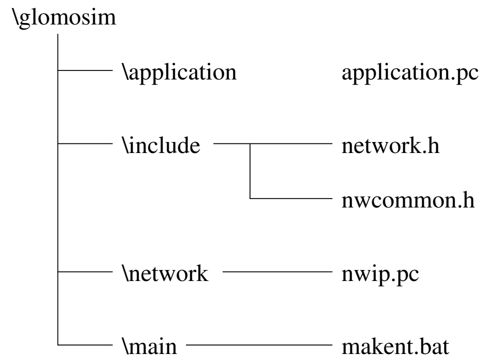

Fig. 4.1. *Files modified in GLOMOSIM*

All routing decisions are taken in the network layer, so we embed our protocol EBSF in the network layer of each node of the network. Whenever the packet enters the network layer, the packet will be handled by EBSF. For our protocol EBSF to work in the network layer of GLOMOSIM, we have modified and included the files EBSF.H and EBSF.PC in the directory `network` of GLOMOSIM. The program written in the file EBSF.PC contains the code for threshold computation and the code to determine whether to broadcast or to discard a message.

Whenever the packet reaches the network layer, it will be handled by protocol EBSF as follows (see Figure 4.2). At the beginning, the initialization function $RoutingEBSFInit()$ is called and defines a structure called *GlomoRoutingEBSF* and allocates memory to it. Also the stats (statistics to be printed to determine the enhancements) are initialized, the sequence table for the node, and message cache. The top value is initialized to 0.

A routing function $RoutingEBSFRouter()$ determines the routing action to be taken and handles the packet accordingly (if it is from UDP or MAC). If the packet is from UDP, then the node is the source node and the data is sent, i. e., the function $RoutingEBSFSendData()$ is called. If the data comes from the MAC layer, then the decision is made as whether to send it to UDP or drop the packet. This decision is made in the function $RoutingEBSFHandleData()$.

The function $Finalize()$ initializes the statistics part of the protocol. When this function is called, it collects the statistics from the file "glomo.stat" in the "bin" directory and formats the statistics such that number of data transmitted, the number of data originated, and the number of data received for each node are printed.

The function $RoutingEBSFHandleData()$ is called whenever the node receives the packet from the MAC layer. This function checks with the message cache and it decides whether to transmit or discard the packet. The nodes that are with in the transmission range will receive the packet and only the nodes that are greater than threshold distance and lesser than transmission range will transmit the packet. If the node is at a distance lesser than a threshold value from a transmitting node, then the packet is discarded.

The message cache is implemented as a linked-list, to which an element is inserted whenever a new message arrives at a node; the element contains the source address and the message's sequence number. The function $LookupMessageCache$ searches the message cache to see whether the message already exists, using its sequence number. The function $InsertMessageCache$ inserts a message into the cache if it is not already present there. Other packet functions are $GLOMO\_MsgAlloc$, $GLOMO\_MsgAddHeader$, $GLOMO\_MsgRemoveHeader$, and *GLOMO_MsgPack-etAlloc*.

The PARSEC compiler, called `pcc`, accepts all the options supported by the C compiler, and also supports separate compilation. C programs (files with .c suffix) and object files (files with .o suffix) can also be compiled and linked with PARSEC programs. PARSEC programs are usually given an extension `.pc`. PARSEC supports separate compilation of entities.

Figure 4.3 shows the GUI of the GLOMOSIM simulator for a network of 49 nodes arranged as a grid of 7x7 and in which node 35 transmits messages to all other nodes.

The threshold distance is defined as $N \times transmission\_range$ of a node, where N is a real number between 0 and 1.

We have implemented and tested our protocol for a network with $n = 49$ nodes (a perfect grid) and various values of the threshold (see Figure 4.4).

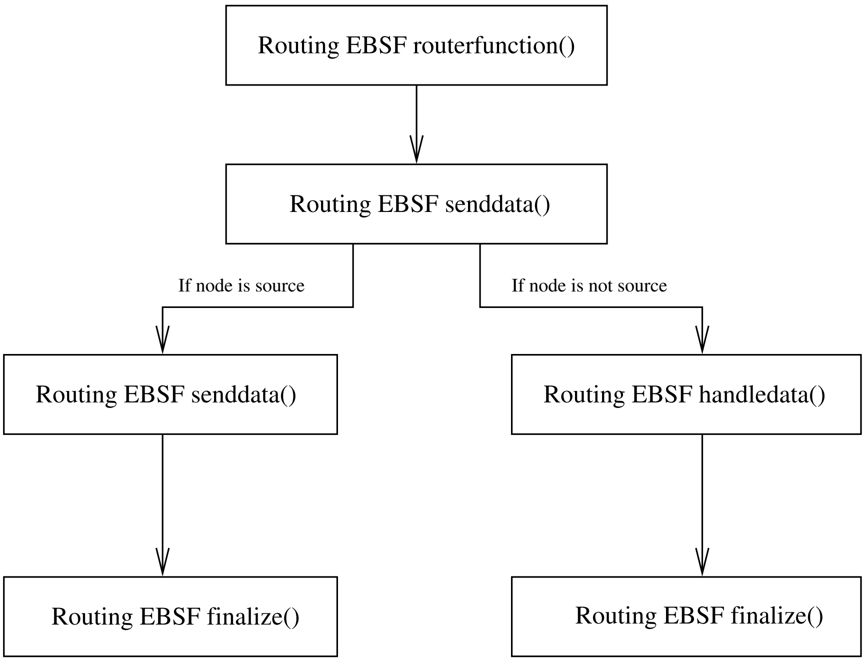

Fig. 4.2. *Module Diagram of EBSF*

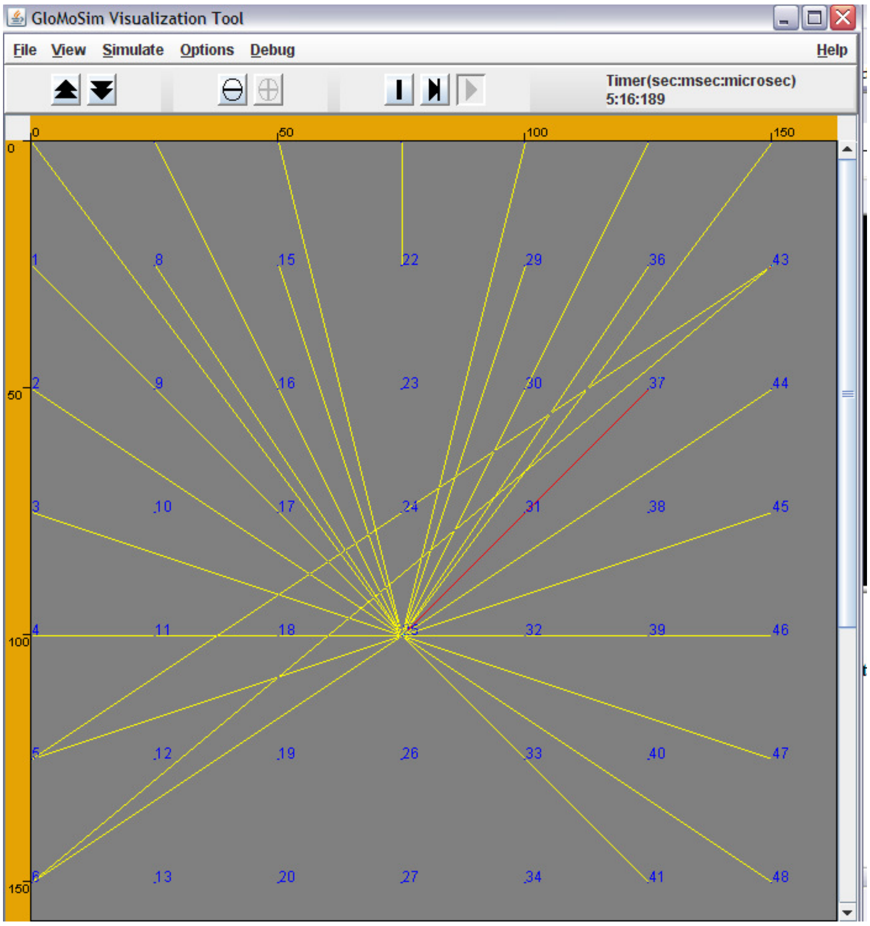

Fig. 4.3. *GUI of GLOMOSIM for a 49-node network; node 35 is the sender*

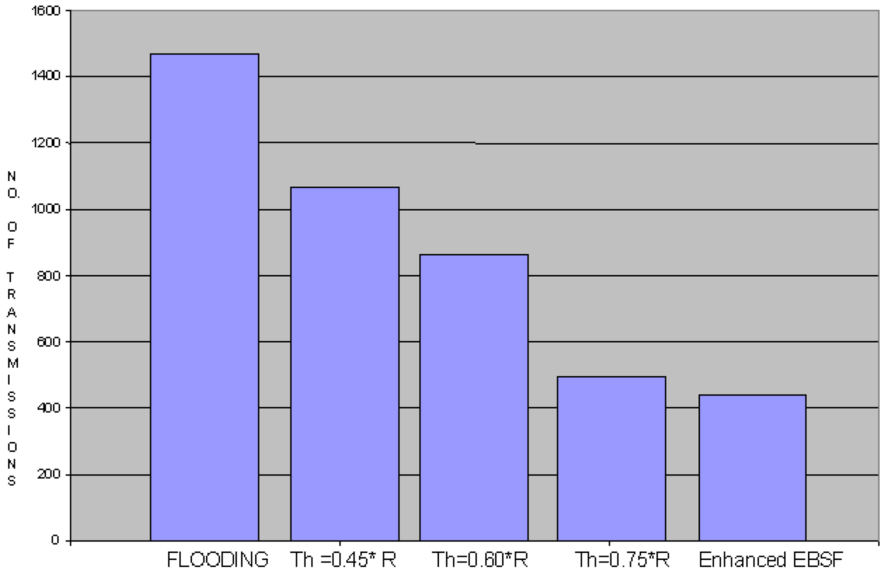

Fig. 4.4. *Performance Analysis with Different Threshold Values*

The performance statistics are presented in Table 4.2.

5. Conclusion. Building efficient protocols for ad hoc networks is a challenging problem due to the dynamic nature of the nodes. Unlike traditional fixed-infrastructure networks, in MANETs there is no centralized control over the network or over the size of the network. The proposed Efficient Broadcasting by Selective Forwarding has a number of advantages over other approaches considered in the literature. First of all, it selects a minimal number of nodes to retransmit, thus it utilizes the bandwidth efficiently. Secondly, it minimizes the number of unnecessary transmissions and therefore it reduces the number of redundant packets. Thirdly, it minimizes the overall power consumption since only the selected nodes use their power for re-transmission. The selected nodes are chosen based on the threshold value; the threshold value can be tuned to show the performance enhancements. Higher the threshold value, more optimized results are obtained. EBSF does not impose any bandwidth overhead and reduces the power consumption drastically. The efficiency of EBSF remains very high even in large networks. Overall, the proposed protocol shows that broadcasting can be enhanced greatly by choosing only an optimal set of nodes for transmission and thus avoiding redundant transmissions and at the same time ensuring data delivery to all the nodes in the network.

This protocol could be integrated with any routing protocol for finding a route in mobile ad-hoc networks with minimal power consumption and without imposing any bandwidth overhead.

Current research in wireless networks focuses on networks where nodes themselves are responsible for building and maintaining proper routing (self-configure, self-managing). Our algorithm does not adapt to topology changes; this is a topic of future research. If nodes are missing from the grid, the threshold value needs to be decreased; in case new nodes are added, the threshold value needs to increase to keep the performance of the protocol. This increasing or decreasing has to be done dynamically, and better in a non-centralized manner. If new nodes are added to the network, then the set of selected nodes may need to be recomputed. It is desirable to minimize the number of such computations while stile preserving the properties of EBSF.

Acknowledgments. The authors would like to thank the anonymous reviewers for the valuable comments that have improved the quality of the paper.

TABLE 4.2
Performance Statistics

Node	Flooding	EBSF threshold=0.45*R	EBSF threshold=0.6*R	EBSF threshold=0.75*R
0	30	0	0	0
1	30	30	30	30
2	30	0	0	0
3	30	0	0	0
4	30	30	30	0
5	30	30	30	30
6	30	21	7	8
7	30	0	0	0
8	30	0	0	0
9	30	0	0	0
10	30	30	0	0
11	30	30	30	0
12	30	30	30	30
13	30	20	8	10
14	30	30	0	0
15	30	0	0	0
16	30	30	0	0
17	30	30	0	0
18	30	30	30	30
19	30	30	30	30
20	30	22	9	8
21	30	30	30	0
22	30	30	30	0
23	30	30	30	0
24	30	30	30	30
25	30	30	30	30
26	30	18	14	4
27	30	22	16	2
28	30	30	30	30
29	30	30	30	30
30	30	30	30	30
31	30	30	30	30
32	30	22	16	10
33	30	22	16	9
34	30	25	18	8
35	30	20	13	12
36	30	22	10	9
37	30	26	9	6
38	30	27	12	6
39	30	21	16	8
40	30	24	17	7
41	30	27	19	12
42	30	25	17	13
43	30	25	19	13
44	30	28	21	10
45	30	26	20	9
46	30	22	22	12
47	30	25	21	12
48	30	27	22	22

REFERENCES

[1] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg. A lower bound for radio broadcast. *J. Comput. Syst. Sci.*, 43:290–298, October 1991.

[2] X. Chen, M. Faloutsos, and S. Krishnamurthy. Distance Adaptive (DAD) broadcasting for ad hoc networks. In *Proceedings of MILCOM*, vol. 2, October 2002.

[3] S.-J. Choi, K.-H. Kwon, and S.-J. Yoo. An efficient cross-layer based flooding algorithm with retransmission node selection for wireless sensor networks. In *22nd International Conference on Advanced Information Networking and Applications - Workshops (AINA Workshops 2008)*, pp.941-948, March 25-28, 2008.

[4] I. Gaber and Y. Mansour. Broadcast in radio networks. In *Proceedings of SODA*, pages 577–585, January 1995.

[5] GLOMOSIM. Installation of GLOMOSIM 2.03 in windows XP. `http://www.cs.ndsu.edu/~ahmed/GLOMOSIM.html`

[6] GLOMOSIM. `http://pcl.cs.ucla.edu/projects/GLOMOSIM/`

[7] GLOMOSIM. `http://www.sm.luth.se/csee/courses/smd/161\_wireless/glomoman.pdf`

[8] S. Guha and S. Khuller. Approximation algorithms for connected dominating sets. In *Proceedings of ESA*, 1996.

[9] J. Haas. A new routing protocol for the reconfigurable wireless networks. In *Proceedings of 6th IEEE International Conference on Universal Personal Communications (ICUPC)*, pages 562–566, October 1997.

[10] A. M. Hanashia, A. Siddique, I. Awan, and M. Woodward. Performance evaluation of dynamic probabilistic broadcasting for flooding in mobile ad hoc networks. Simulation Modelling Practice and Theory, 17(2):364–375, February 2009.

[11] D. B. Johnson. Routing in ad hoc networks of mobile hosts. In *Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 158–163, December 1994.

[12] N. Karthikeyan, V. Palanisamy, and K. Duraiswamy. Optimum density based model for probabilistic flooding protocol in mobile ad hoc network. European Journal of Scientific Research, 39(4):577–588, 2010.

[13] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *Proceedings of ACM/IEEE Mobicom*, pages 66–75, October 1998.

[14] Y. B. Ko and N. H. Vaidya. Location-aided routing in mobile ad hoc networks. *Technical report* 98-012, Texas A&M University, 1998.

[15] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. *Wireless Networks*, 6:307–321, 2000.

[16] H. Lim and C. Kim. Multicast tree construction and flooding in wireless ad hoc networks. In *Proceedings of the ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, 2000.

[17] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. The broadcast storm problem in a mobile ad hoc network. In *Proceedings of ACM MOBICOM*, pages 151–162, August 1999.

[18] V. K. Paruchuri, A. Durresi, and R. Jain. Optimized flooding protocol for ad hoc networks. IEEE Transactions on Parallel and Distributed Systems, 15(11):1027–1040, November 2004.

[19] W. Peng and X. Lu. On the reduction of broadcast redundancy in mobile ad hoc networks. In *Proceedings of MOBIHOC*, 2000.

[20] W. Peng and X. Lu. Ahbp: an efficient broadcast protocol for mobile ad hoc networks. *Journal of Science and Technology*, 2002.

[21] C. Perkins and S. Das. Ad hoc on-demand distance vector (aodv) routing. http://tools.ietf.org/html/rfc3561, July 2003.

[22] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying: an efficient technique for flooding in mobile wireless networks. Technical Report Technical Report 3898, INRIA - Rapport de recherche, 2000.

[23] J. Sucec and I. Marsic. An efficient distributed networkwide broadcast algorithm for mobile ad hoc networks. Technical Report CAIP Technical Report 248, Rutgers University, September 2000.

[24] B. Williams and T. Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *Proceedings of the ACM MOBIHOC*, pages 194–205, 2002.

[25] J. Wu and F. Dai. Broadcasting in ad hoc networks based on self-pruning. In *Proceedings of IEEE INFOCOM*, 2003.

Edited by: Costin Bădică, Adina Magda Florea
Received: March 4, 2010
Accepted: March 31, 2010