Parameterized Neighborhood-Based Flooding For Ad Hoc Wireless Networks

Vijay Dheap, Mohammad Ahmad Munawar, Sagar Naik, Paul A.S. Ward
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo Ontario N2L 3G1, Canada
{vdheap,mamunawa,knaik,pasward}@engmail.uwaterloo.ca

Abstract

Flooding is a simple routing technique that can be used to transmit data from one node to every other node in a network. The focus of this paper is to investigate improvements to flooding techniques used in ad hoc wireless networks. Recent work has focused on using topological information to reduce the number of broadcasts. The number of broadcasts necessary to flood the network was the major performance metric used to compare previous neighborhood-based flooding algorithms. We build upon this foundation by first presenting a Parameterized Neighborhood-Based Flooding (PNBF) algorithm, which provides a single platform for the performance comparison of various multi-hop neighborhood-based flooding algorithms. We also introduce and motivate the use of additional performance metrics, including total number of collisions and percentage of nodes that receive the message, for comparing flooding algorithms. An analysis is given of how different network properties, such as average node degree, communication patterns, affect the performance of the different neighborhood-based flooding algorithms. Our simulation results demonstrate that our algorithm is capable of handling a wide variety of situations where properties of ad hoc networks along with the relative importance of the performance criteria are taken into consideration.

I. Introduction

An ad hoc wireless network is a self-configuring communication network, established by mobile devices using the wireless medium. The configurations of an ad hoc wireless network are transient, since the nodes that comprise the network may enter or leave dynamically. Centralized administration is absent, and each node behaves as a router as well as a host [3].

In this paper, we study the problem of flooding in ad hoc networks. Flooding is required by a number of protocols in ad hoc networks, including route discovery [10] and self-configuration [6], [5]. In an ad hoc wireless network flooding usually occurs as follows: the source node transmits its data packets and every node in the network that receives the data packets retransmits them until the termination criteria is met.

In wireless transmission when a network node transmits any data packets without channel interference, every node that is located within the transmitting range can receive the data. This transmission is referred to as broadcasting. The amount of network resources used is effectively independent of the number of nodes that receive a broadcast message from the transmitting node. As a result, the total number of broadcasts used is the cost criterion for flooding techniques. Redundant broadcasts in a wireless network can potentially degrade the performance of an ad hoc network by causing contention and collisions. In the extreme case, the broadcast storm problem may occur [8]. Blind flooding in an ad hoc wireless network results in a large number of redundant broadcasts. Finding the least number of broadcasts necessary to flood a network is an NP-complete problem [3], though not one worth solving, since some redundancy is desirable. Specifically, it increases the probability that all nodes in the network receive the message because wireless communication is prone to high transmission error rates and the shared medium leads to collisions.

The objective of our work is to develop a parameterizable algorithm that requires fewer broadcasts than does blind flooding, but provides effective flooding in the presence of packet loss. In particular, we require that the algorithm be configurable to the characteristics of a network and needs of the application. In this paper, we motivate the development of our parameterized flooding algorithm and provide simulation results to demonstrate the performance benefits that our algorithm offers.

II. Background and Related Work

Given the importance of effective flooding in ad hoc networks, recent research in this area has focused on exploiting topological information to reduce the number of redundant broadcasts. The idea is to balance the transmission and computational overhead, with a reduction in the number of redundant broadcasts. Each node in the network shares local topological information with neighboring nodes so that these nodes can use the information to make "smarter" broadcasting decisions.

An ad hoc network can be modeled as a graph where the vertices represent the devices in the network. An edge exists between any two vertices when their corresponding devices are within wireless transmission range of each other. A broadcast from a device reaches all vertices which are adjacent to that device's vertex. A device that receives a broadcast can choose to forward that broadcast or not, according to the particular flooding algorithm that it is executing. Minimal flooding is defined as the least number of broadcasts required to reach all vertices in the graph [3]. Determining a minimal flood is NP complete.

Flooding techniques can be categorized into probability-based, area-based, and neighborhood-based schemes [10]. In a probability-based algorithm, a node retransmits a received message based on a predetermined probability. In area-based algorithms a device evaluates its additional coverage area based on the transmissions it receives to determine whether to retransmit the message. For this paper we will focus on the neighborhood-based flooding algorithms.

A. *Self-Pruning*

In self-pruning [3] every node in the network is assumed to know its immediate neighbours. When a node receives a data packet to be forwarded, it also receives the one hop neighborhood information of the sender. The node then determines whether it has any neighbours which are not included in the sender's neighbourhood list. If it determines that all its neighbours are included in the neighbourhood list of the sender then it does not forward the data packet. Otherwise it does. As this process continues, all the nodes in the network will eventually receive the data packet. The authors of this algorithm assume that nodes in the network are relatively static in terms of their location during the flooding process.

B. *Dominant Pruning*

Dominant pruning extends self-pruning by using two-hop neighbourhood information [3]. The proposition is that the additional information used to make forwarding decisions will decrease the overall number of broadcasts. The other significant difference from self-pruning is the point at which the forwarding decision is taken. Dominant pruning gives the sender the right to decide which of its neighbours will forward the data packet. This list of forward nodes is piggybacked with the data packet. A node that receives the data packet and is not in the forward list does nothing, but if the node is on the forward list then it computes its own forward list and forwards the data packet to its neighbours. This process is iterated until all the nodes in the network have received the data packet. A node determines its forward list so that every node two hops away from the node will receive the data packet.

C. *Total Dominant Pruning*

Lou and Wu improved dominant pruning by reducing the number of forward nodes generated at each step [4]. They suggest that by incurring the extra overhead of transmitting the 2-hop neighbourhood information of the sender with the data packet will enable the forwarding receiver to reduce the number of nodes it selects as forwarding nodes in the next iteration. Overall, this results in a smaller set of forward nodes than in dominant pruning.

D. *Partial Dominant Pruning*

Partial dominant pruning [4], like total dominant pruning, is a derivative of dominant pruning. However, unlike total dominant pruning, the extra cost of partial dominant pruning relative to dominant pruning is incurred not through transmission overhead, but by computational overhead at the broadcasting node. To reduce the number of forwarding nodes, redundant broadcasts in the common two-hop neighbourhood of two neighbour nodes in a network are pruned.

III. The PNBF Algorithm

The family of neighbourhood-based flooding algorithms is incomplete. We arrive at this proposition due to two reasons. Firstly, we recognize that the algorithms presented thus far, limit themselves to 2-hop neighbourhood information to make forwarding decisions. The second reason is that minimal flooding is not ideal when more realistic characteristics of ad hoc networks are incorporated into the model. The shared communication medium in ad hoc wireless networks leads to collisions resulting in corrupted data. It would be desirable to maintain a level of redundancy in the flooding procedure to improve the probability that all nodes in the network receive the data packets.

The first objective was to determine whether there was a sensible trade off for using additional neighbourhood information, to reduce the number of broadcasts in light of the additional transmission/computational overhead. Parameterizing the level of hop information that a node would gather in making the forwarding decision enabled us to achieve this objective.

The second objective was to ascertain the additional metrics under which optimality of a flooding algorithm could be analyzed. We removed the major assumption of an ideal MAC-layer environment in which collisions were not considered, since it does not match wireless transmission behaviour. This assumption was present in the earlier neighbourhood-based algorithms. If an ideal MAC layer is not assumed, then the broadcasts of acknowledgments and retransmissions must also be incorporated into the model which was not done so in previous studies. A more realistic view of ad hoc networks is when nodes have to consider transmission delays and collisions when participating in the flooding procedure. Some of the metrics we employed to assess the optimality of neighbourhood-based flooding algorithms were the number of collisions, total time of the flooding procedure, as well as the metric used by previous papers, which is the total number of broadcasts necessary to flood the network.

We developed the Parameterized Neighbourhood-Based Flooding algorithm (PNBF). A formal description of PNBF is given in the following subsection. This algorithm is capable of using h-hop neighbourhood information to make the broadcast decision where h is greater than or equal to two. When considering two-hop neighbourhood information the algorithm is equivalent to TDP. PNBF provides a single platform for making performance comparisons when the level of neighbourhood information taken into account is varied. The first two performance metrics used were the total number of broadcasts and the average transmission overhead.

The next logical step was to introduce transmission delay/time and collisions into the model to assess performance in a more realistic environment. Collisions occur when q node receives data from two or more nodes during overlapping time intervals. As the number of collisions increases the number of nodes that receive the message during flooding decreases dramatically. In order to increase the probability that nodes receive the message, a random time delay was introduced. The random time delay is a function of the transmission time of the data. This enables new metrics of interest such as total flooding time, total number of collisions, and average number of nodes that encounter collisions.

Finally, it should be noted that in an ad hoc network, mobility of the nodes is an issue. During the course of the flooding procedure we assumed that the positions of the nodes are relatively static. To justify this assumption, consider an 802.11 ad hoc wireless network. In order to transmit 1KB of data 300 meters over an 11 Mbps channel 100 times, it would take 73 milliseconds. Within this very short period of time, it is assumed that the topology of the network does not change significantly to influence flooding. Another assumption is that transmission range between any two nodes in the ad hoc network is symmetric. Finally, we assume (as do previous neighbourhood-based flooding algorithms) that nodes exchange neighbourhood information to determine all the nodes in their h-hop neighbourhood. This information can be updated through knowledge acquired from normal traffic.

A. PNBF Algorithm

Let x and y be two nodes in the ad hoc network. Assume that node x is the originator of a message. Assume that node y receives the message when node x performs a broadcast.

Definition 1 (Immediate Neighbours: $I(x) \subseteq V$): The immediate neighbours of node x are those nodes that are within transmission range of node x.

Definition 2 (Neighbourhood: $N^h(x) \subseteq V$): The h-hop neighbourhood of a node x is the set of nodes that are reachable within h-hops from x. Formally:

$$N^h(x) = \begin{cases} x & \text{if } h = 0 \\ N^{h-1}(x) \cup \left\{ \bigcup I(t) \mid t \in N^{h-1}(x) \right\} & \text{if } h > 0 \end{cases}$$

Definition 3 (Universe: $U^h(x, y) \subseteq V$): $U^h(x, y)$ is defined as the set of nodes that node y must aim to reach after h hops from y since receiving the message from node x. Formally:

$$U^h(x, y) = N^h(y) - N^h(x)$$

Definition 4 (Collection of Forward Nodes Sets): F^y is the collection of sets of nodes that are designated by node y to rebroadcast the message when they receive it. (Assume that x has designated y as a forward node).
$F^y = \{F^i(x, y) \mid 1 \leq i \leq h - 1\}$ where h is the number of hops and x is the node from which node y received the message.
$F^i(x, y) \subseteq U^i(x, y)$ such that $\{\bigcup (N^1(t) \cap U^{i+1}(x, y)) \mid t \in F^i(x, y)\} = U^{i+1}(x, y)$

1) Greedy Set Cover Algorithm to Initialize a Set of Forward Nodes (GSCFN): This modified set cover algorithm will pick a minimum number of nodes from a designated set (O) such that the neighbourhoods of these nodes include all the nodes contained in another set (S). It is greedy because it selects a node whose neighbourhood intersects the most number of nodes from the set to cover (S). This process is continued until S is covered.

Input: the set to cover (S), the set to select nodes from (O).
$T = \emptyset$.
$F = \emptyset$ (the set of forward nodes).
While (T != S)
    Pick e from O such that $N(e) \cap S$ is max.
    $T \longleftarrow T \cup (N(e) \cap S)$
    $F \longleftarrow F \cup \{e\}$
    $O \longleftarrow O - \{e\}$
end
Output: F

2) The PNBF Algorithm: If node is initiator of the flood

Compute F^{node}

Piggyback F^{node} and node's h-level neighbourhood information with the message and perform a broadcast.

When an intermediate node receives a message

If node is designated as a forward node.

compute F^{node}

Merge F^{node} with the received F set.

Piggyback F^{node} and node's h-level neighbourhood information with the message and perform a broadcast.

Else do nothing

3) Explanation of Specific Steps: We will refer back to the original assumptions concerning nodes x and y. The algorithm parameter h defines the level of neighbourhood information used in flooding.

1. Computation of F^y

$F^{h-1}(x,y) = \text{GSCFN}[U^h(x,y), U^{h-1}(x,y)]$. [Note: $U^h(x,y)$ corresponds to the set to cover (S) and $U^{h-1}(x,y)$ corresponds to the set to select nodes from (O)].

$h' = h - 2$.

Loop while $h' \geq 1$:

1. $F'(y) = \text{GSCFN}[F^{h'+1}(x,y), U^{h'}(x,y)]$
2. $h' = h' - 1$

2. Merging F^y with F^x

$F^x = \{F^i(\_,x) \mid 1 \leq i \leq h-1\}$, where $\_$ is an anonymous node in the network from which x received the message.

$F^y = \left\{\bigcup_{i=1}^{i=h-2}[F^{i+1}(x,y) = F^i(\_,x) \cup F^{i+1}(x,y)]\right\} \cup F^1(x,y)$

3. Special Cases and Clarification

Node y has thus far been assumed to be a forward node. Node y checks F^x to determine if it is a forward node. Specifically, if y is member of $F^1(\_,x)$ [the first element in F^x] then it is designated as a forward node. Otherwise y does not retransmit the message.

If node $x = \emptyset$, which implies that y is the source of the flood, then its h-level neighbourhood is its universe $[U^h(\emptyset,y) = N^(y)]$.

4. Termination Condition

Every node designated as a forward node retransmits the data at most once. When all the forward nodes have finished transmitting the data the flooding process ends.

IV. Algorithm Analysis

We evaluated our algorithm through simulation using the SimJava package [9]. In order to simulate a wireless ad hoc network we generated random connected graphs using Stojmenovic's algorithm [7]. Each node in a graph represents a device in the ad hoc network and each edge represents the wireless connection among nodes. Since SimJava does not directly support wireless networks, we adapted it by defining a MAC-layer broadcast as a transmission on all edges extending from a node in the graph.

Our analysis is divided into two phases. Initially we used the same assumptions as previous neighbourhood-based algorithms so as to demonstrate that our approach is a true generalization of that prior work. We then incorporated transmission delay into the simulation, which prior approaches had not, to determine if this had a significant effect on the behaviour of our, and previous, algorithms.

A. Ideal MAC Layer

Prior neighbourhood-based flooding algorithms use an ideal MAC-layer assumption for their simulation and/or analysis. Specifically, they presume that the transmission time is negligible, and can therefore be ignored. We used this assumption in our initial evaluation so as to provide a basis for comparison with the previous methods. The relevant performance criteria are the total number of broadcasts and the average transmission overhead. We simulated the PNBF algorithm using 2-, 3-, 4-, and 5-hop neighbourhood-based flooding. Recall that the 2-hop version of the PNBF algorithm is equivalent to the TDP algorithm, and has identical performance to that of TDP.

Each h-hop PNBF algorithm was simulated with graphs whose number of vertices ranged from 20 to 100. Every algorithm was simulated one hundred times for each graph size to justify a certain level of performance over a large test space. For every simulation, a different graph of the same size was generated and the performances of all the algorithms were measured on the same graph. The average number of total broadcasts and the average transmission overhead was calculated for the different algorithms for each graph size. This process was repeated twice, once for average node degree 6 and once for average node degree 16. An approximation algorithm for minimal flooding (Approximate Minimum Connected Dominating Set (AMCDS)) was also incorporated into the simulation as an approximation of the lower bound.

The parameters of interest in this phase were average node degrees, number of nodes in the network, and hop-level neighbourhood information used by the PNBF algorithm. We observed that taking into account additional neighbourhood information, while slightly better than TDP, has no signifcant benefit in graphs with average node degree 6. However, as can be seen in Fig. 1, when the node degree is 16 the additional neighbourhood information dramatically reduces the number of broadcasts necessary for flooding. The 5-hop version requires one-third the

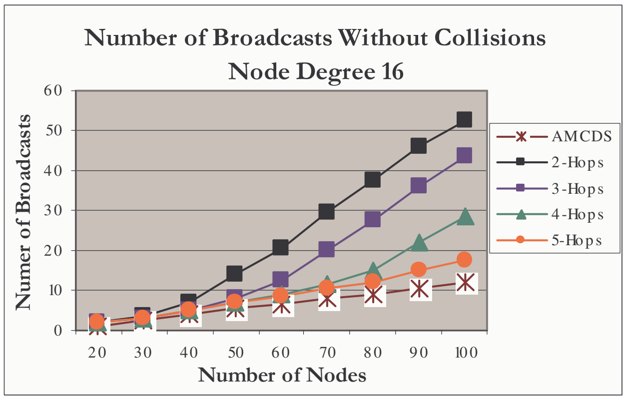

Fig. 1. Results for Ideal MAC-Layer Assumption

number of broadcasts than does the 2-hop version (*i.e.* TDP) for networks with up to 100 nodes. This was because redundant broadcasts are more effectively identified and removed when nodes have more interconnections. Significant increases in performances could be observed as both the average node degree and the overall number of nodes in the network was increased.

The average transmission overhead increased as additional neighbourhood information was considered in making forwarding decisions. However, this tradeoff between reducing the number of broadcasts and the increase in transmission overhead was to be expected.

B. *Effect of Transmission Duration*

The second set of simulations removed the ideal-MAC-layer assumption and introduced transmission delays. All other parameters and procedures for these simulations were unaltered from the ideal-MAC-layer experiment. Since transmissions take a finite time, the possibility of packet collision, and resulting loss, must be taken into account. In this experiment a bandwidth of 1 Mbps was assumed to determine the transmission delays. In addition to the prior measurement criteria, we now also determine the total number of collisions and the percentage of nodes that receive the message. We call the latter metric the "coverage" of the algorithm.

In the ideal-MAC-layer simulations there was consistently 100% coverage during flooding and the transmission was assumed to be instantaneous. When delay is taken into account, coverage dropped significantly, and was never greater than 90% when there were more than 20 nodes. Coverage declined as the number of nodes increased. The 2-hop approach (TDP) achieved slightly less than 90% coverage in networks of average node degree 6. However, when network denisty increased, 2-hop coverage dropped to 80–85%. The 5-hop approach was worse, achieving 60–80% coverage. This is likely because the 5-hop approach is more efficient, requiring less broadcasts, and thus more vulnerable to failure in the event of packet collision.

Overall we see that neighbourhood-based flooding does not work in the absence of an ideal MAC layer. We therefore explored mechanisms to increase the probability of achieving 100% coverage. We observed that the cause of the lower coverage is broadcast storms, and so we took the simple approach of incorporating a random time delay at each node to mitigate this effect. To evaluate this approach, we simulated the 2-, 3-, 4-, and 5-hop methods with a graph size of 100 nodes, since this was the worse case for coverage, and with the random time delay selected from a range that varied from 0 to 5 times the transmission delay of a packet. In this experiment it was seen that as the random-time-delay range was increased, coverage approached 100%, exceeding 98% with a range of 4.5 times the packet transmission time.

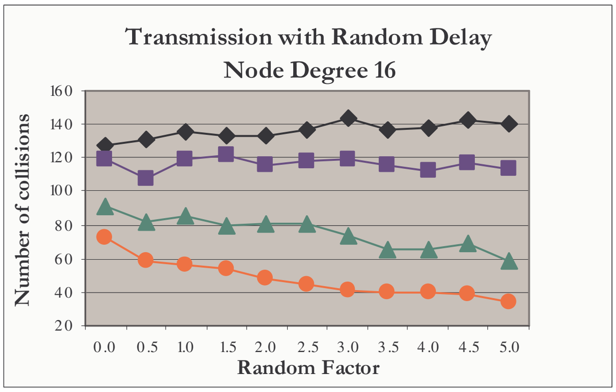

Fig. 2. Resolving the Transmission Collision Problem

For networks with average node degree 6, the number of collisions declines as the random-delay range is increased. For denser networks, as can be seen in Fig. 2, the total number of collisions only declines when more neighbourhood information is included in the algorithm. It must be noted, however, that even though the number of collisions increased in the 2-hop case shown in the figure, the coverage also increased, to more than 99%. This apparent contradiction is possible because the 2-hop case has so much redundancy, as we saw in the ideal-MAC-layer results.

We now recall that the original motivation for the neighbourhood-based approach was to reduce the total number of MAC-layer broadcasts required for flooding. We therefore measured this number under our modified algorithm. Fig. 3 shows that not only have we still met this motivation, but in fact we achieved superior results through this random delay factor. First note that performance relative to the number of broadcasts is better for higher multi-hop versions of the PNBF algorithm. Second,

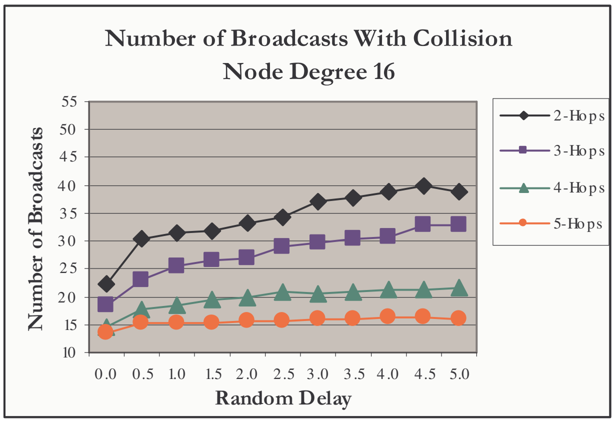

Fig. 3. Broadcasts vs. Delay

if we compare these results to those of the ideal-MAC-layer assumption (see Fig. 1), we can see that we are now achieving the required coverage with less total broadcasts than were required if there were no collisions. This is because remaining collisions simply remove some of the redundancy within the algorithm.

The versions of the PNBF algorithm need to be discussed with regard to two more performance criteria. It was stated earlier that coverage improves as the random time delay is increased. However, this improvement begins to level off beyond the 5 times transmission-delay mark. In general, the 2-hop version of PNBF achieved higher probabilities of coverage with a smaller time delay. This is to be expected, as it maintained a higher level of redundancy in broadcasts. The transmission overhead incurred was also minimal in the 2-hop version of PNBF since the level of neighbourhood information to be piggybacked with a message was less than that of other versions of the PNBF algorithm. However, these two performance criteria were the tradeoff for reducing the traffic in the communication channel due to flooding. Reduction of traffic minimized the number of collisions as well as the number of nodes that were affected by it. The incorporation of the random time delay provided effective coverage. When considering an ad hoc network, communication other than flooding will be taking place simultaneously. Therefore, reducing the traffic on the channel is vital to guarantee that network resources are used effectively and collisions avoided. The appropriate version of the PNBF algorithm could be selected, based on common network activity. If communication among nodes while flooding is high, then selecting 5-hop version of PNBF will be a better choice than the 2-hop version.

To summarize the results, increasing use of topological information leads to a reduction in the total number of broadcasts. As a network becomes increasingly dense, there is significant reduction in the number of broadcast packets and the number of collisions when additional topological information is considered. A random delay is required to achieve effective coverage. Our PNBF algorithm enables an ad hoc network to configure its flooding procedure based on network properties, and the performance criteria it deems to be most important to the functioning of the network.

V. Conclusion

In this paper we have provided a generalized neighbourhood-based flooding algorithm, and demonstrated its efficacy through simulation. Prior neighbourhood-based algorithms assumed an ideal-MAC layer, and focused on the total number of MAC-layer broadcasts as the performance metric. We have demonstrated that these algorithms cannot provide effective coverage in the presence of collisions. Our method solves this problem, while keeping the benefits of the previous neighbourhood-based approaches. Since our algorithm is parameterized, it can be easily adapted to specific network conditions. We are currently exploring techniques to reduce the overhead of neighbourhood-based algorithms, and to more effectively deal with node mobility. We also plan to explore the tradeoff between the random delay and the total time required for the flooding operation.

References

[1] Bevan Das, Raghupathy Shivkumar, and Vaduvur Bharghavan. Routing in ad hoc networks using a virtual backbone. In *Proceedings of the International Conference on Computers and Communication Networks (IC3N)*, pages 1–20, September 1997.

[2] Sudipto Guha and Samir Khuller. Approximation algorithms for connected dominating sets. In *European Symposium on Algorithms*, pages 179–193, 1996.

[3] Hyojun Lim Chongkwon Kim. Flooding in wireless ad hoc networks. *Computer Communications*, 24(3–4):353–363, February 2001.

[4] Wei Lou and Jie Wu. On reducing broadcast redundancy in ad hoc wireless networks. *IEEE Transactions on Mobile Computing*, 1(2):111–122, April–June 2002.

[5] A. J. McAuley and K. Manousakis. Self-configuring networks. In *21st Century Military Communications Conference Proceedings*, volume 1, pages 315–319, 2000.

[6] Sanket Nesargi and Ravi Prakash. MANETconf: Configuration of hosts in a mobile ad hoc network. In *Infocom 2002*. IEEE, 2002.

[7] Ivan Stojmenovic, Mahtab Seddigh, and Jovisa Zunic. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 13(1):14–25, January 2002.

[8] Yu-Chee Tseng, Sze-Yao Ni, Yuh-Shyan Chen, and Jang-Ping Sheu. The broadcast storm problem in a mobile ad hoc network. *Wireless Networks*, 8(2/3):153–167, 2002.

[9] University of Edinburgh. Simjava. Available at: http://www.dcs.ed.ac.uk/home/hase/simjava/.

[10] Brad Williams and Tracy Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pages 194–205, 2002.