Pushing the Limits of Multicast in Ad Hoc Networks

Katia Obraczka, Gene Tsudik, Kumar Viswanath
USC Information Sciences Institute
4676 Admiralty Way s. 1001, Marina del Rey, CA 90292, USA
E-mail contact: katia@isi.edu
Tel: (310) 822-1511 Fax: (310) 823-6714

September 15, 2000

Abstract

This work focuses on the requirements of "better than best effort" (high hop-by-hop delivery guarantee) broadcast in highly dynamic mobile multi-hop ad hoc networks (MANETs). Our work is motivated by mission-critical applications such as disaster relief and military operations. This class of applications is characterized by: (1) high delivery guarantee requirements even in the presence of high mobility, and (2) broadcast-style of communication where all nodes are receivers.

Extensive simulations conducted on two different platforms show that, as node speeds, network traffic load, and number of senders increase, the performance of existing multicast protocols (exemplified by ODMRP and MAODV) degrade in terms of packet delivery and overhead. In contrast, simple flooding, while clearly not a panacea, performs comparatively well and shows promise as a foundation for more specialized protocols for highly dynamic MANETs of the future.

1 Introduction

Mobile multi-hop ad hoc networks, or MANETs, are characterized by the lack of any fixed network infrastructure. All network components of a MANET can be mobile. Moreover, there is no real distinction between a host and a router since all nodes can be sources as well as forwarders of traffic.

MANETs differ from traditional, fixed-infrastructure mobile networks, where mobility occurs only at the last hop. In such networks, issues such as address management arise; however, these do not affect core network functions, most importantly, routing. In contrast, MANETs require fundamental changes to conventional routing and packet forwarding protocols for both unicast and multicast communication. Conventional routing mechanisms are based on routers maintaining distributed state about the network topology. These mechanisms were designed for wired networks and work well in fixed-infrastructure mobile networks. However, topology changes in MANETs can be very frequent making conventional routing mechanisms both ineffective and expensive.

A number of MANET-oriented multicast routing protocols have been recently proposed. They can be grouped into two main categories: proactive and reactive. Protocols of the former variety maintain routing state, while the latter reduce the impact of frequent topology changes due to mobility by acquiring routes on demand. However, a common feature of all heretofore proposed protocols is that they address unreliable, best-effort multicast following the IP multicast model.

One of the main challenges presented by multicast routing in MANETs is the need to achieve robustness in the presence of universal mobility and frequent node outages and failures. We observe that key features of MANETs make them attractive for deployment in critical environments, such as military or civilian emergency operations. Since in almost any critical environment, robustness and high quality of service are very important, multicast routing and packet forwarding algorithms (which may be attractive otherwise) that cannot provide high delivery guarantees may be inadequate for mission-critical, highly dynamic MANETs.

2 Focus

The focus of our work is to probe the need for new protocols that: (1) specifically target broadcast in highly dynamic MANETs and (2) can provide high delivery guarantees. These features are crucial when addressing the needs of mission-critical applications like emergency rescue and military operations. For instance, in a battlefield scenario, it is imperative that all communication from a battalion commander coordinating an attack reaches all soldiers.

In particular this paper compares the performance of mesh-based and tree-based multicast protocols with plain flooding. The On-Demand Multicast Routing Protocol (ODMRP) [7] was chosen to represent mesh-based protocols since it has been shown to the best performer among a recently proposed protocol pool [11]. Multicast Ad hoc On-Demand Distance Vector (MAODV) [14] was chosen to represent tree-based protocols. Both protocols belong to the reactive category. Flooding is arguably the simplest and oldest routing technique. Despite the hefty overhead, it provides the best delivery guarantees for unicast, multicast and broadcast in wired networks. However, it is not clear that it will perform the same way in wireless, radio networks due to the collision effect.

We use simulations to study the reliability of the different protocols under a wide range of mobility conditions, traffic source populations, and network traffic loads. Although we are mainly interested in how multicast protocols perform in broadcast communication, we also evaluate them for different multicast scenarios. We evaluate the tradeoff between robustness and efficiency by measuring the overhead incurred by each protocol. We also measure packet delivery delay as another performance metric. One unique feature of this work is our parallel use of two different simulation platforms: (`ns`) [8], and GloMoSim [16]. The motivation for using two simulation platforms is not only to validate the simulations study by comparing results obtained on two simulators but also to evaluate the consistency of the simulators themselves.

Our simulation results demonstrate that – as node speed, numbers of senders, and network traffic load increase – flooding outperforms ODMRP and MAODV with respect to reliable delivery for both broadcast and multicast communication. This is not surprising since ODMRP and MAODV , like other multicast protocols, were not specifically designed for high speed. In particular, most simulations and experiments with MANET-oriented multicast (and unicast) protocols found in recent literature involve node speeds on the order of 20 *m/sec* (72 *km/hr*). While such speeds might accurately reflect many current MANETs, we believe that future MANETs will be made up of nodes traveling at much higher speeds, in excess of 200 *km/hr*, i.e., roughly the high end of today's automobile speeds. Our simulations also show that flooding exhibits good performance in terms of overhead generated and packet propagation delay.

The rest of this paper is organized as follows. In the next section, we review ODMRP and MAODV and briefly describe our implementation of flooding. Section 4 describes the simulation environment used, including a detailed description of the simulation parameters. In Section 5, we present and discuss simulation results. We also highlight the lessons we learned from using two simulation platforms in parallel. Finally, Section 7 presents some concluding remarks as well as our future work plans.

3 ODMRP, MAODV and Flooding Overview

In this section we review the operation of ODMRP and MAODV, as well as highlight the main features of our flooding implementation. It is worth pointing out that for all three protocols we perform broadcast, a special case of multicast. In other words, we always have the set of receivers be identical to the set of network nodes.

3.1 On Demand Multicast Routing Protocol (ODMRP)

The On-Demand Multicast Routing Protocol (ODMRP) [7] falls into the category of on-demand protocols since group membership and multicast routes are established and updated by the source whenever it has data to send. Unlike conventional multicast protocols which build a multicast tree (either source-specific or shared by the group), ODMRP is mesh-based. It uses a subset of nodes, or *forwarding group*, to forward packets via scoped flooding.

Similar to other on-demand protocols, ODMRP consists of a request phase and a reply phase. When a multicast source has data to send but no route or group membership information is known it broadcasts a JOIN DATA packet.

When a neighbor node receives a unique JOIN DATA packet it records the upstream node ID in its *message cache*, the node's routing table, and rebroadcasts the packet. This process' side effect is to build the reverse path to the source. When a JOIN DATA packet reaches the multicast receiver, it generates a JOIN TABLE packet that is broadcast to its neighbors. The JOIN TABLE packet contains the multicast group address, sequence of (source address, next hop address) pairs, and a count of the number of pairs. When a node receives a JOIN TABLE, it checks if the next node address of one of its routing table entries matches its own address. If it does, the node realizes that it is on the path to the source and thus becomes a part of the forwarding group for that source by setting its *forwarding group flag*. It then broadcasts its own JOIN TABLE built upon matched entries. The next hop IP address can be obtained from the message cache. This process constructs (or updates) the routes from sources to receivers and builds the forwarding group. Membership and route information is updated by periodically (every JOIN-DATA-REFRESH interval) sending JOIN DATA packets. Nodes only forward data if they belong to the forwarding group and they receive a non-duplicate data packet. The forwarding group helps to set up redundant paths when the primary path breaks due to node mobility.

3.2 Multicast Ad hoc On-Demand Distance Vector (MAODV)

MAODV, like ODMRP, is also an on-demand multicast routing protocol. However unlike ODMRP it is tree based. Route discovery is based on a route request (RREQ) and route reply (RREP) cycle. When a multicast source requires a route to a multicast group, it broadcasts a RREQ packet with the join flag set and the destination address set to the multicast group address. A member of the multicast tree with a current route to the destination responds to the request with a RREP packet. Non members rebroadcast the RREQ packet. Each node on receiving the RREQ updates it route table and records the sequence number and next hop information for the source node. This information is used to unicast the RREP back to the source. If the source node receives multiple replies for its route request it chooses the route having the least hop count or the freshest sequence number. It then sends a multicast activation message (MACT). The MACT is used to activate the path from the source node to the node sending the reply. If a source node does not receive a MACT message within a certain period it broadcasts another RREQ. After a a certain number of retries ($RREQ\_RETRIES$), the source assumes that there are no other members of the tree that can be reached and declares itself the *group leader*. The group leader has the function of periodically broadcasting group hello (GRP_HELLO) messages to maintain group connectivity. Nodes also periodically broadcast hello (HELLO) messages with a TTL of one to maintain local connectivity.

3.3 Flooding

Our implementation of routing by flooding is quite standard: when a node receives a packet, it broadcasts the packet except if it has seen that packet before. Nodes keep a cache of recently received packets; older packets are replaced by newly-received ones. A node only re-broadcasts a packet if that packet is not in the node's cache.

We use a well-known randomization technique to avoid collisions: when a node receives a packet it waits a random time interval between 0 and *flooding interval* before it rebroadcasts the packet.

4 Methodology

One of the contributions of our study is that we use two different simulation platforms as a way to cross-validate our results. We ran simulations on both the Global Mobile System Simulator (GloMoSim) and the Network Simulator (ns).

GloMoSim [16] is a discrete-event network simulator developed at UCLA. It was written in PARSEC, a C-based parallel simulation language [1], and thus can run in sequential or parallel mode. GloMoSim was originally developed to simulate wireless environments and is now being extended to support wired network simulations.

The ns simulator was originally developed at Lawrence Berkeley National Laboratory (LBNL) [12]. Currently it is being extended as part of the VINT project [8] involving USC/ISI, Xerox PARC, LBLN, and UC Berkeley. Like GloMoSim, ns is a discrete-event simulator. However, ns started as a simulation environment for wired networks and has been extended to simulate mobile wireless environments. In particular, we use the CMU Monarch group extensions that enable ns version 2 (ns-2) to simulate MANETs [4].

4.1 Simulation Environment

In our experiments we used GloMoSim 1.1.1 and ns 2.1b6. Since our goal is to validate our simulations by comparing results obtained in GloMoSim and ns, we tried to make the two simulation environments equivalent to one another. We accomplish that by driving both simulators with the same set of parameters. Below we describe those parameters, including the mobility and traffic models we used.

The total simulation time for both simulators was set to 500 seconds. This simulation time ensures that the transmitted packets have a chance to reach the multicast receivers before the simulation halts. A total of 50 nodes were randomly placed in a 1000x1000 meter field. Each sender generated 100 packets using a 2 Mbit/sec channel with a power range of 225 meters. Free space propagation was used to determine whether the nodes that are in the transmitters range are able to receive the data.

Parameter	Value	Description
number-of-nodes	50	network nodes
num-packets	100	messages sent by a node
field-range-x	1000 m	X-dimension of motion
field-range-y	1000 m	Y-dimension of motion
power-range	225 m	node's power range
bandwidth	2 Mbit/s	node's bandwidth
simulation-time	500 s	simulation duration
node-placement	random	node placement policy
propagation-func	FREE-SPACE	propagation function
radio-type	RADIO-CAPTURE	capture effect
mac-protocol	CSMA (GloMoSim) 802.11 (ns)	MAC layer
transport-protocol	UDP protocol	transport layer

Table 1: Simulation Parameters.

Parameter	Value
flooding-interval	25 ms

Table 2: Flooding Parameter Values

Parameter	Value
Join Data refresh interval	3 secs
Forwarding Group Timeout	3 secs
Route Timeout	5 secs
Data Rebroadcast interval	25 ms

Table 3: ODMRP Parameter Values

Parameter	Value
Group Hello Interval	5 secs
Hello Interval	1 sec
Mtree Build	3 secs
Route Discovery Timeout	3 secs

Table 4: MAODV Parameter Values

Mobility Model

The mobility model used is a modified version of the random-waypoint model also known as the bouncing ball model. In this model the nodes start off at random positions within the field. Each node then chooses a random direction and keeps moving in that direction till it hits the terrain boundary. Once the node reaches the boundary it chooses another random direction and keeps moving in that direction till it hits the boundary again. Although we use the same model in both simulators there are some implementation differences. The mobility model in GlomoSim causes the nodes to oscillate at the boundary before they head back towards the center.

Traffic Model

A constant bit rate (CBR) traffic generator was used for both simulators. GlomoSim1.1.1 does not have a CBR application and we developed a traffic generator to simulate CBR sources. The network traffic for the different senders was maintained constant by adjusting the inter-packet interval for the CBR sources.

Other Parameters

Table 1 summarizes generic simulation parameters, while Tables 2, 3, and 4 list protocol-specific parameters.

5 Results

We ran each simulation (keeping all parameters constant) five times, each time using a different seed value. Seeds varied from 1000 to 5000 in steps of 1000. Each data point in the graphs below represents the average across all five runs. In our simulations the senders are chosen randomly from the multicast group members. All the nodes join as members at the start of the simulations and remain members throughout the duration of the simulation.

Recall that our objective is to evaluate multicast routing protocols in highly mobile MANETs. Therefore, in our simulations we subject the protocols to node speeds much higher than the ones normally used in MANET routing protocol evaluation (up to 300 km/h). Although we are mainly interested in broadcast communication (all network nodes are receivers), we also evaluate the protocols for multicast scenarios, where the number of receivers is different than the number of network nodes.

We measure both sides of the protocol reliability (i.e., delivery) versus efficiency trade-off. Packet delivery ratio measures protocol robustness associated to its ability to deliver packets to all receivers. Overhead measures protocol efficiency in terms of the amount of additional information (both data and control) the protocol generates. The third performance metric we use is packet delivery delay.

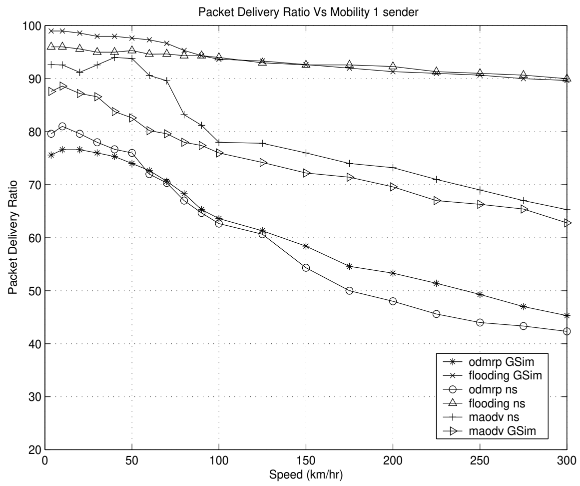
5.1 Reliability in Broadcast Scenarios

We compute packet delivery ratio as the ratio of total number of packets received by the nodes to the total number of packets transmitted times the number of receivers. The graphs in Figure 1 show how protocol reliability varies with mobility (node speed) and the number of traffic sources. We vary node speed from 0 all the way to 300 km/h. While 300 km/h speeds may seem unrealistic for most current MANET environments, they may become reality for future MANETs. We vary the number of traffic sources from 20 to 50 in steps of 10. All other parameters are kept constant including a 20 pkt/sec network traffic load.

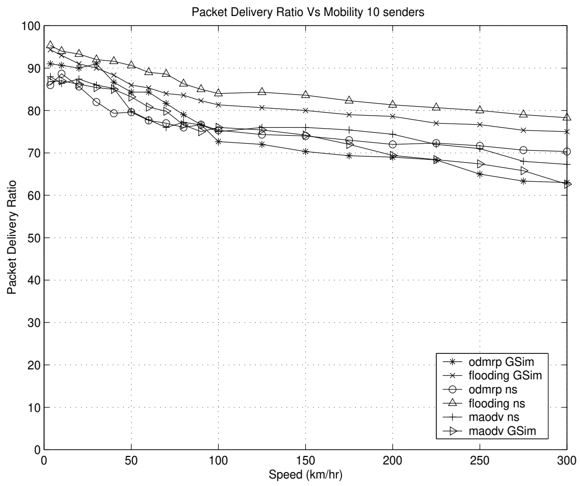
We should point out that in the performance study reported in [11], speeds were limited to 72 km/h and the number of sources- to 20 in a network of 50 nodes. Except when varying multicast group size from 5 to 40 receivers, all other simulations performed in the above mentioned study used 20-node groups.

The study in [11] does not include MAODV in the pool of multicast protocols they evaluated.

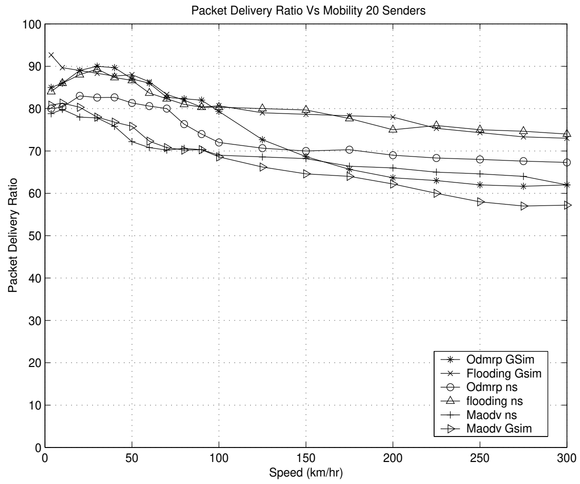
It is interesting to observe that for the 1-sender case, flooding performs considerably better than ODMRP and MAODV as speed increases. However MAODV performs better than ODMRP even at high speeds. This is because



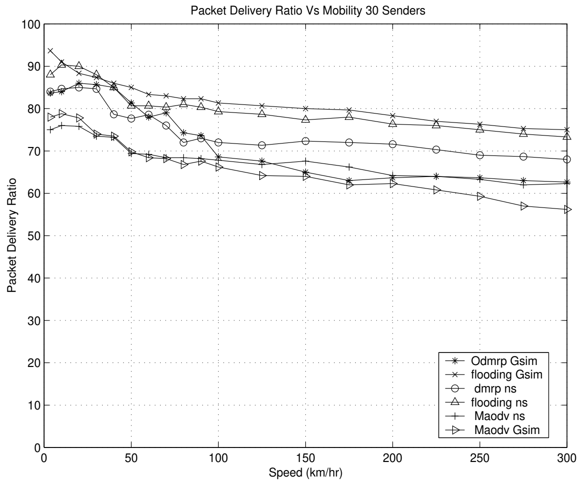
(a) 1 sender



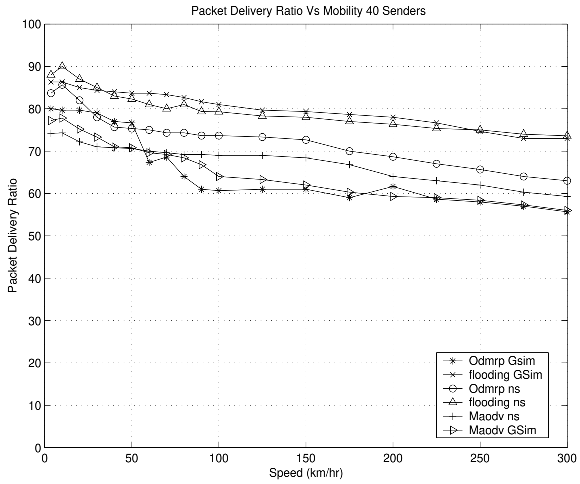
(b) 10 senders



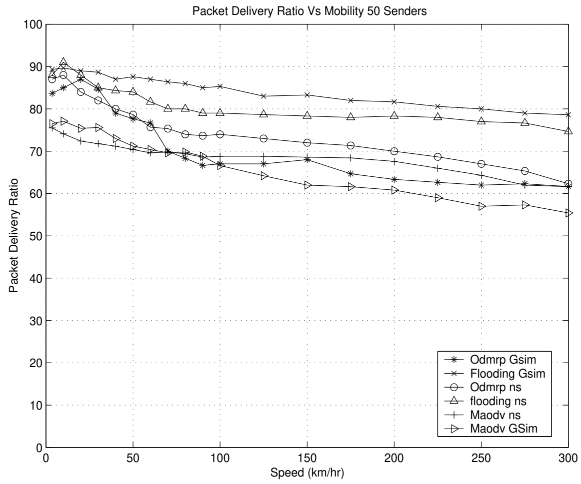
(c) 20 senders



(d) 30 senders



(e) 40 senders



(f) 50 senders

Figure 1: Packet delivery ratio as a function of node speed and number of senders.

we are considering a broadcast scenario with all nodes as receivers. In the case of MAODV this implies that each node is likely to have one or more active group members as neighbors. Hence almost all the nodes rebroadcast the data packet resulting in greater path redundancy. In the case of ODMRP, when the ratio of the number of senders to number of receivers is small (in particular, 1 sender to 50 receivers), the number of forwarding nodes is small when compared to the number of receivers. Therefore, the number of alternate routes is also small, which affects the ODMRP's reliability at higher speeds. To verify this fact we ran some simulations where the number of receivers was smaller than the total number of nodes in the network. In this case MAODV's performance degraded considerably at very high speeds.

The general trend we observe from both simulators is that for scenarios with moderate ratios between number of senders and number of receivers, flooding performs better than ODMRP which in turn performs better than MAODV.

Comparing flooding to ODMRP we notice that at lower speeds the difference in packet delivery ratio is only within 10%. This indeed agrees with what was observed in [11]. As the number of senders increases and for higher speeds, the difference in packet delivery ratio starts widening again. For instance, in the case of 50 senders we observe packet delivery ratio differences of up to 25% in favor of flooding. This is not because ODMRP's performance degrades: it stays reasonably constant as the number of senders increases since network traffic load is kept constant (we illustrate the impact of traffic load on both protocols in Figure 2 below). However, flooding's performance improves because there are fewer collisions. As discussed in Section 5.2 below, as the number of senders increases, ODMRP's overhead can become prohibitive. Flooding's overhead stays constant with the number of senders.

ODMRP outperforms MAODV because of the higher number of control packets MAODV generates in order to maintain the multicast tree. Since we are considering a broadcast scenario, MAODV's performance is hampered by collisions. MAODV's performance however does not vary considerably with the number of senders. As explained above, MAODV's performance is related to the number of receivers rather than the number of senders.

Although both simulators show similar performance trends, we notice some differences in the results they report. These discrepancies are due to inherited differences between the simulators as well as different ways we implemented certain features and models. These issues are discussed in more detail in Section 5.6 below.

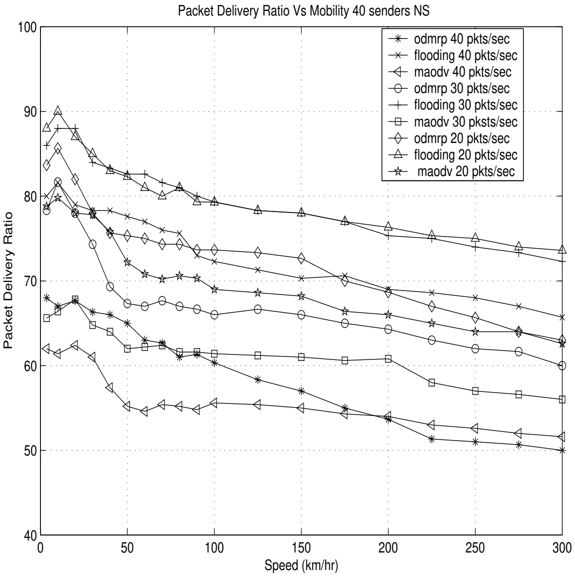
Network Traffic Load

The goal of this next set of experiments is to evaluate the behavior of both protocols for different network traffic loads. The graphs in Figure 2 plot packet delivery ratio as a function of node speed for workloads of 20, 30, and 40 pkt/sec. We used 40 and 50 senders. We show ns and GloMoSim results in separate graphs for readability purpose.

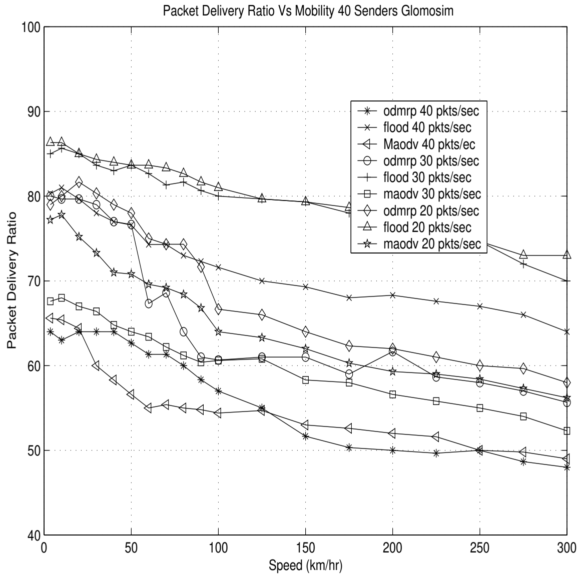
Note that all protocols present lower delivery ratio in the 40-sender case. This is because, for the same traffic load with lower number of senders, the inter-packet interval is smaller and thus the chance of losses due to collisions is higher.

We observe that for both 40 and 50 senders and as the traffic load increases flooding maintains higher delivery ratio than ODMRP and MAODV. Take the 50-sender, 40 pkt/sec case: at 0 speed, flooding's delivery ratio is approximately 85%,ODMRP's is 75% and MAODV's is 60% (referring to results from ns simulations); flooding delivers about 68% of the packets at 300 km/h, while ODMRP 55% and MAODV 52%. Similar values are obtained from the GlomoSim simulations as well. This is because ODMRP's delivery ratio is dependent on the relation between the *inter_packet* and $JOIN\_DATA\_REFRESH$ intervals. For higher loads, $inter\_packet\_interval < JOIN\_TABLE\_REFRESH$. This means that when packets are sent, routes can be stale and packets can be lost. Clearly, in higher network traffic scenarios, ODMRP's $JOIN\_TABLE\_REFRESH$ frequency can be increased but that will increase overhead. It will also increase the probability of losses due to collisions as more control packets will be competing with data packets for the shared medium.

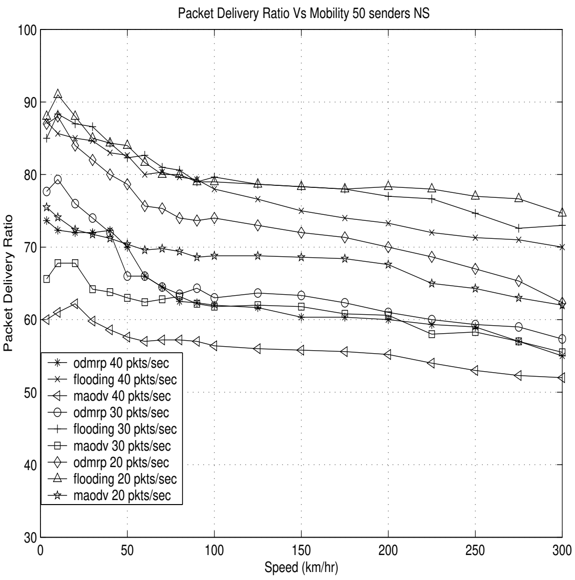
For the different network load conditions ODMRP performs better than MAODV. As previously mentioned, MAODV requires more control packets than ODMRP (although lower number of control bytes) to maintain the multicast tree. However, as we previously pointed out, MAODV's performance is not affected by speed as much as ODMRP's. This is validated by simulation results from both platforms.



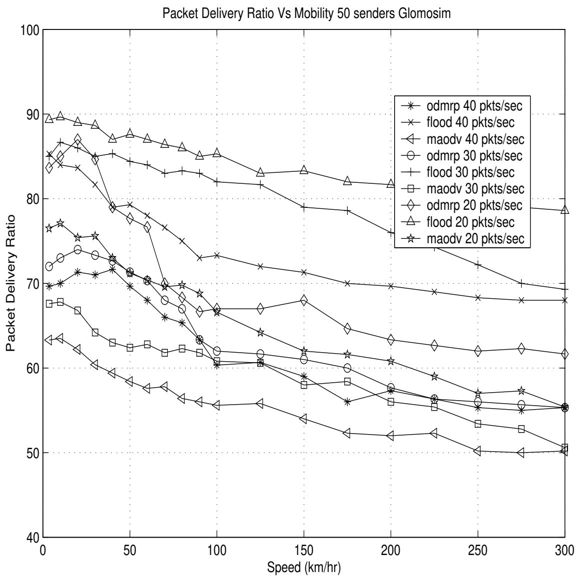
(a) 40 senders - ns



(b) 40 senders - GloMoSim



(c) 50 senders - ns



(d) 50 senders - GloMoSim

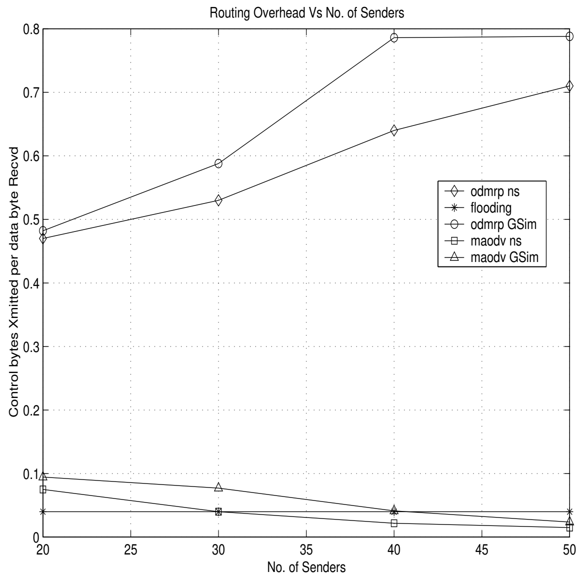
Figure 2: Packet delivery ratio as a function of network traffic load and node speed for 40 and 50 senders.

5.2 Overhead

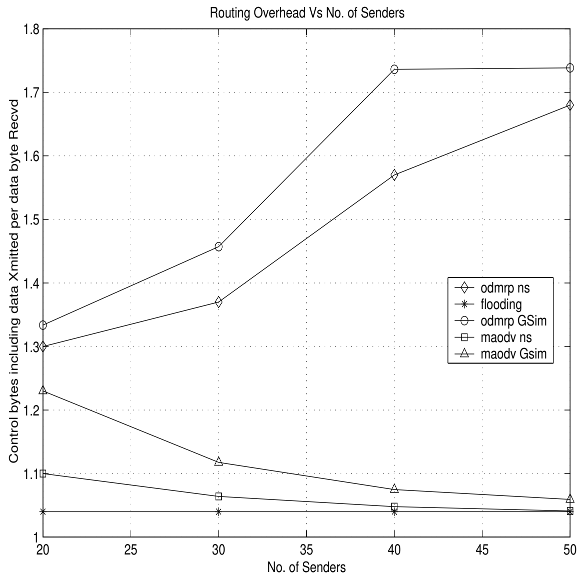
We use two metrics for protocol overhead: while one accounts only for routing overhead, the other measures both routing and data overhead. The reason for having two different overhead metrics is fairness. If we accounted for routing overhead only, we would be unfair to ODMRP since most of the overhead generated by flooding is due to data duplication.

The control overhead is computed as the ratio between the number of control bytes to the number of data bytes received. In ODMRP, control bytes account for JOIN DATA and JOIN TABLE packets. It also includes data packet header bytes forwarded by the forwarding group members. In MAODV control bytes account for the RREQ, RREP, MACT, HELLO and GRP_HELLO packets. It also includes the data packet headers forwarded by the intermediate nodes. In flooding, control bytes include all data header bytes forwarded by network nodes.

In addition to the control bytes, the second overhead metric also includes data payload bytes for all data packets forwarded by intermediate nodes. It should be noted that this ratio will be smaller for ODMRP in the case of smaller number of senders because of fewer rebroadcasts of the data packets .



(a) Pure routing overhead



(b) Routing and data overhead

Figure 3: Protocol overhead as a function of number of senders.

The graphs in Figure 3 plot pure routing and routing plus data overhead as a function of number of traffic sources. In these simulations, we keep traffic load constant at 20 pkt/sec. As expected, flooding's overhead stays constant with the number of traffic sources. ODMRP's overhead is considerably higher than flooding or MAODV and grows with the number of senders. This is because it builds per-source meshes. Although ODMRP is a on-demand protocol it has to periodically flood JOIN DATA packets to maintain the mesh connectivity. Another reason is that the JOIN-TABLE replies are expensive in terms of the number of bytes. Therefore, the more senders there are, the more control and data overhead gets generated. The same behavior was observed in [11], but only for up to 20 senders.

In the case of MAODV, the routing overhead required to maintain the tree is independent of the number of senders. Hence the routing overhead per data byte transmitted is higher for a smaller number of senders and decreases as the number of senders increases.

In both graphs, we observe that the difference in ODMRP overhead between the 40- and the 50-sender case is quite small. We speculate that, as the number of senders nears the total number of network nodes, so does the set of forwarding nodes. Therefore, the overhead stays roughly the same in these two cases.

When comparing both graphs, we also notice that, in ODMRP, the variation of control and data overhead is higher than the variation of pure control overhead. This is because the amount of data being flooded increases in tandem with the number of senders (because of the commensurate increase in the number of forwarding nodes). Note that we have only plotted the flooding overhead for GloMoSim since the overhead in ns is also the same.

5.3 Delay

We also measure the time it takes for packets to arrive at receiver nodes. The graphs in Figure 4 plot average packet delivery delay as a function of the number of senders. Network traffic load was kept constant at 20 pkt/sec and node mobility at 10 km/hr. We average the delay it takes a packet to get to all nodes over all nodes. We then take the average over all packets sent. From the graph it can be seen that the delay statistics obtained from both simulators show the same trend.

Flooding provides a lower bound on average delivery delay since it almost always sends packets over the shortest paths to the receivers. For instance, in the 20-sender case, ODMRP's delay is almost three times higher. As expected, flooding's average delay increases with the number of senders. Notice that the behavior of ODMRP is the inverse: as the number of senders increases, the delivery delay decreases. Although this may sound surprising at first, it is because of the fact that that ODMRP "degenerates" to flooding as the number of senders and receivers equals the total nodes in the network.

We observe that MAODV's delay is higher than ODMRP's due to the collision effect previously described. Note that it stays almost constant with the number of senders.

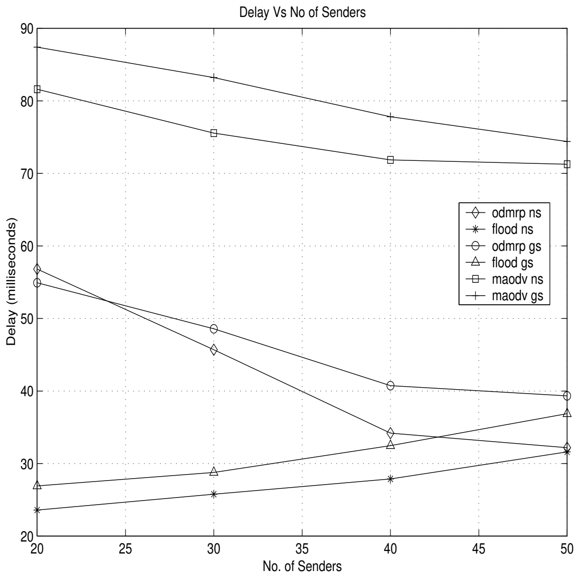
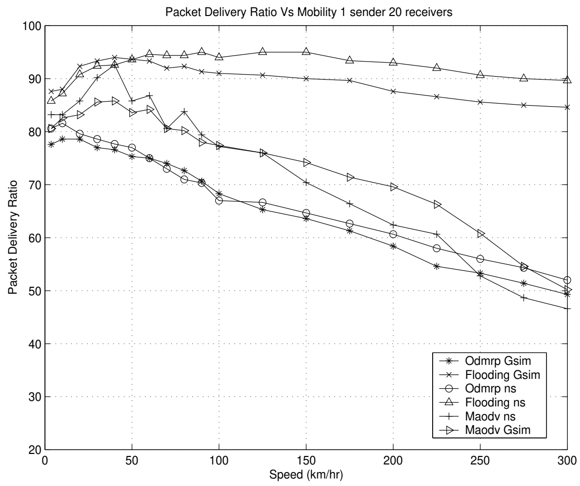


Figure 4: Delay as a function of number of senders.

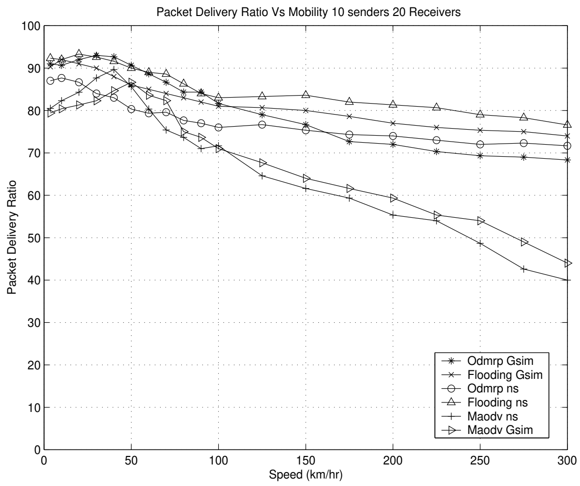
5.4 Reliability in Multicast Scenarios

In this section we present the results for those scenarios where the number of receivers is less than the total number of nodes. The number of senders and receivers range in the sets {*1, 10, 20*} and {*20, 30*}, respectively. The total number of network nodes is set at 50 and the network traffic load is fixed at 20 pkts/sec.

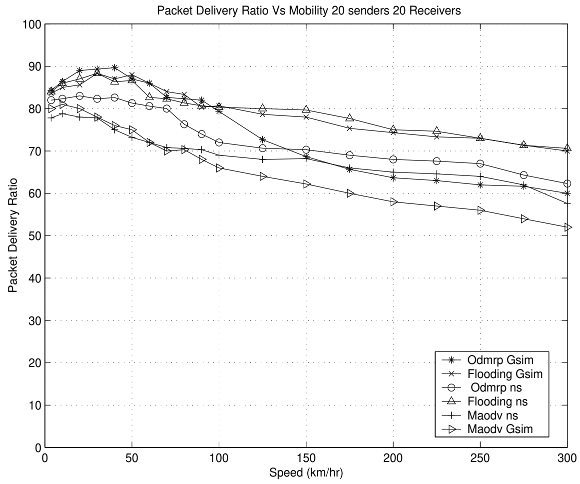
The graphs in Figure 5 show that packet delivery ratios for flooding in multicast are lower when compared to broadcast (see Figure 1. This is because every unique packet received and forwarded, in broadcast, is counted toward packets succesfully delivered since all nodes are receivers. However, in multicast, every unique packet received and



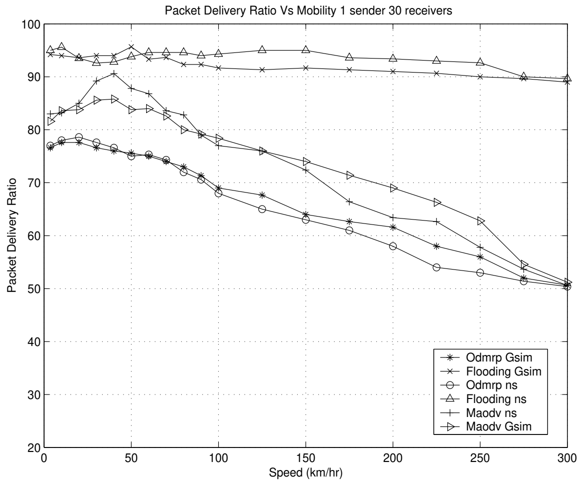
(a) 1 sender 20 Receivers



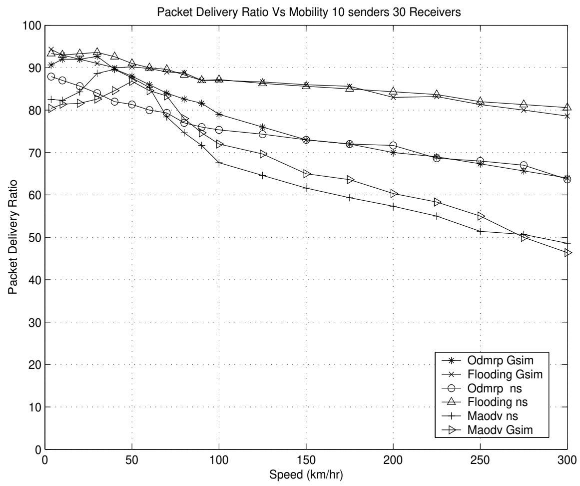
(b) 10 senders 20 Receivers



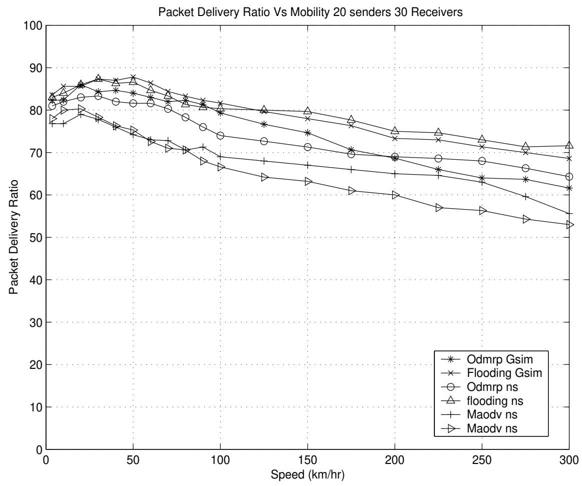
(c) 20 senders 20 Receivers



(d) 1 sender 30 Receivers



(e) 10 senders 30 Receivers



(f) 20 senders 30 Receivers

Figure 5: Packet delivery ratio as a function of node speed.

forwarded does not necessarily count as succesfully delivered as the forwarding node may not be a member of the multicast group.

In the 1-sender case, flooding performs considerably better than ODMRP and MAODV as the speed increases (This is explained in detail in Section 5.1). When compared to the 1-sender broadcast scenario ODMRP performs slightly better because of the lower control traffic messages and collisions.

Surprisingly, MAODV's performance is slightly degraded as compared to the broadcast scenario at high speeds. This is because in the broadcast scenario each node is highly likely to have one or more active group members. Hence, almost all the nodes re-broadcast the packet resulting in greater path redundancy.

As the number of senders increases the performance of flooding degrades slightly as more packets are lost by collision and channel contention. ODMRP's performance approaches that of flooding as the number of senders is increased because of the increase in the number of forwarding groups and greater path redundancy. This can be observed in Figure 5(c) for the 20 senders 20 receivers scenario. However, flooding still outperforms ODMRP for increased mobility.

5.5 Overhead

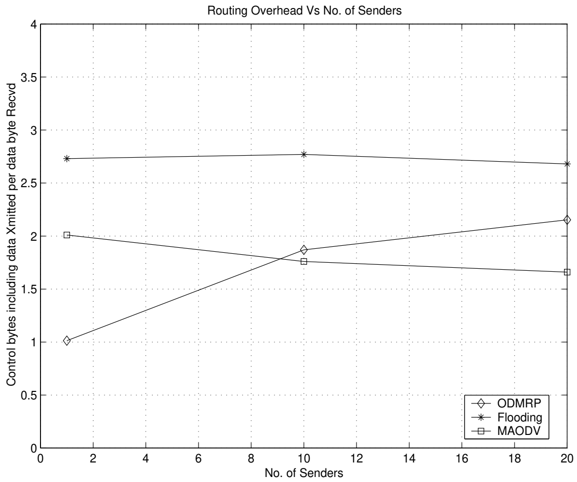


Figure 6: Routing and Data Overhead

The graphs in Figure 6 plot routing plus data overhead as a function of number of traffic sources. In these simulations, we keep traffic load constant at 20 pkts/sec and the multicast group size was fixed at 20 nodes.

The routing overhead for flooding is higher than ODMRP and MAODV. This is expected because of the large number of redundant broadcasts. However the overhead remains fairly constant with the increase in the number of senders and is due to the fact that the network traffic is maintained constant at 20 pkts/sec even with increase in number of traffic sources. The routing overhead of MAODV is independent of the number of senders as explained in Section 5.2. Hence the *Control overhead/Data byte received* decreases as the number of senders increases. In the case of ODMRP the routing overhead increases with the increase in the number of traffic sources. As the number of senders increases, more JOIN DATA packets are propagated and the control overhead increases accordingly.

5.6 Summary and Discussion

In this section, we summarize our results presented above and highlight the lessons learned from using two different simulation platforms in parallel.

- Flooding performs better than ODMRP and MAODV both in broadcast and multicast communication for a wide range of node speeds, different number of senders and network traffic loads.

- In terms of reliability, ODMRP performs better than MAODV given that there is not a large gap between numbers of senders and receivers.
- However MAODV's performance does not degrade as much as ODMRP's at very high speeds.
- At high traffic loads, ODMRP's performance degrades as node speed increases. This can be explained by the relationship between the inter-packet interval and ODMRP's JOIN-TABLE-REFRESH timer. Making the JOIN-TABLE-REFRESH timer smaller will attenuate the problem at the expense of increased overhead.
- ODMRP's overhead increases drastically with the number of senders, while flooding's stays constant. When we consider control and data overhead, the discrepancy increases since ODMRP uses flooding among the forwarding nodes.
- MAODV's routing overhead is independent of the number of senders. Hence the number of control bytes required per data byte transmitted decreases as the number of senders increases.
- In general, flooding does better than ODMRP and MAODV in terms of packet delivery delay. However, it was interesting to observe that ODMRP's delay nears flooding's as the number of senders approaches the number of receivers. MAODV's delay does not vary much with an increase in the number of senders.

Using Different Simulators: Lessons Learned

The motivation for using both GloMoSim and ns is to cross-validate the results obtained from our simulations. Getting consistent results from both simulators makes our study more credible. A by-product of using two experimental platforms in parallel and trying to obtain equivalent simulation environments from both of them was an in-depth understanding of the simulators themselves.

We tried to drive both simulators with the same set of parameters. However, inherent differences between GloMoSim and ns probably account for the minor variations we observed. As mentioned before, though we use the same (*bouncing ball*) mobility model, due to implementation differences in GloMoSim, nodes tend to oscillate around the boundary before bouncing away. In ns, nodes immediately bounce away when they reach a field boundary.

The original propagation model in ns was the two-ray propagation model which we modified to match GloMoSim's free space propagation model.

Another difference between the simulators is the threshold used by the radio capture model to determine if a packet was correctly received.

A point to note is that we used CSMA as the MAC protocol in GlomoSim and 802.11 in the case of ns. Since both ODMRP and MAODV employ unicast control messages, using 802.11 (and its collision avoidance mechanism) should have a positive impact on the performance of the two protocols. Although we do not expect this effect to be considerable, it is another factor which explains the differences in the results obtained from the two simulators.

6 Related Work

MANET Multicast Routing Protocols

MANET multicast protocols can be classified according to how they propagate data as tree-based or mesh-based. While tree-based protocols propagate data over a tree spanning all multicast group members, in meshed-based protocols a subset of network nodes (the mesh) is responsible to forward data to all multicast receivers. MANET protocols can also be classified according to how they acquire/maintain routes. Reactive (or on-demand) protocols acquire routes on demand and proactive protocols maintain routing state.

Examples of meshed-based protocols include ODMRP, CAMP [6], and flooding. ODMRP is reactive, while CAMP is proactive. Flooding of course does not require routing state maintenance. AMRoute [2] and AMRIS [15] are examples of proactive, tree-based protocols. MAODV exemplifies a reactive, tree-based protocol.

Performance of MANET Multicast

Several studies have evaluated the performance of **unicast** routing protocols for MANETs [3, 5, 10, 9]. The only performance study of MANET **multicast** routing protocols was reported in [11]. This comparative study analyzed five different protocols including ODMRP and flooding. It concluded that mesh-based protocols in general, and ODMRP in particular, perform better than tree-based approaches.

As we previously pointed out, this work focused on a different portion of the routing protocol design space. It evaluated protocols for lower mobility scenarios and communication involving a smaller set of traffic sources. For these scenarios, ODMRP outperformed all the other protocols evaluated, except for flooding. As we confirmed in our study, for these scenarios, the performance difference between flooding and ODMRP is reasonably small.

Our study targets a different segment of the design space. We confine our interest to "better than best effort" broadcast in highly dynamic MANETs. Our results show that, for a wide range of node speeds, different number of traffic sources and network traffic load, flooding performs better than ODMRP and MAODV with respect to delivery reliability and delay.

Another contribution of the present work is our parallel usage of two different simulation platforms to cross-validate the results. Except for some minor differences in absolute results, both simulators reported very similar performance trends for the protocols studied.

7 Conclusions, Ongoing, and Future Work

In summary, this paper reported on simulation-driven experiments evaluating three approaches to broadcast and multicast communication with an emphasis on high delivery guarantees in highly dynamic mobile ad hoc networks. The results demonstrate that as both network traffic load and numbers of senders increase, multicast protocols (exemplified by ODMRP and MAODV) degrade in terms of delivery ratios and overhead. Whereas, plain flooding performs relatively well and shows promise as a foundation for more specialized protocols for highly dynamic MANETs of the future.

We are currently investigating different flooding variations to increase reliability, lower redundant broadcasts (recall that the packet delivery ratio in flooding is still in the mid-70% when number of senders is over 30 for higher traffic loads and higher node speeds) and address bandwidth overhead issues. Some variations of flooding like *hyper-* and *scoped-flooding* have been proposed [13].

We have implemented two different scoped-flooding strategies. The first one uses the received signal power associated with an incoming packet to decide whether a node should re-broadcast the packet. Transmission power is fixed for all nodes and free space attenuation is used to determine the received signal power. If the received power is greater than a pre-specified threshold, it implies that the receiving node is quite close to the node that broadcast the packet. Hence the coverage region of a re-broadcast would considerably overlap the first broadcast and retransmitting the packet is unnecessary. We have simulations results for different thresholds corresponding to distances of 35 and 50 meters. The packet delivery ratios for the 35 meter scenario is similar to that obtained for pure flooding. However the number of redundant broadcasts is significantly lowered in this case resulting in a lower byte overhead per data byte delivered.

Another scoped-flooding variation we are investigating is based on neighbor discovery: each node periodically transmits hello messages along with its neighbor list. When a node receives a transmission it compares the neighbor list of the transmitting node with its own neighbor list. If the receiving node's neighbor list is a subset of the transmitting node's neighbors then it does not have to retranmit the packet. We are currently running simulations to study the performance of this form of scoped flooding. Prelimnary results indicate that delivery ratios similar to pure flooding can be obtained for certain scenarios with lower redundant broadcasts.

References

[1] R. Bagrodia and R. Meyerr. PARSEC: A parallel simulation environment for complex system. Technical report, UCLA, 1997.

[2] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade. AMRoute: Adhoc multicast routing protocol. IETF manet (draft-talpade-manet-amroute-00.txt), August 1998.

[3] J. Broch, D.A. Maltz, D.B. Johnson, Y.C Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. *In Proceedings of ACM/IEEE MOBICOM'98*, pages 85–97.

[4] CMU Monarch Project. *Mobility Extensions to ns-2*, 1999. Available from http://www.monarch.cs.cmu.edu/.

[5] S.R. Das, R. Castaneda, J. Yan, and R. Sengupta. Comparative performance evaluation of routing protocols for mobile, ad hoc networks. *In Proceedings of IEEE IC3N'98*, pages 153–161.

[6] J. Garcia-Luna-Aceves and E. Madruga. A multicast routing protocol for ad-hoc networks. In *Proc. of INFOCOM'99*, pages 784–792, March 1999.

[7] M. Gerla and S.J. Lee. On-demand multicast routing protocol for mobile ad-hoc networks. Available from http://www.cs.ucla.edu/NRL/wireless/.

[8] The VINT Group. VINT:virtual internet testbed. http://netweb.usc.edu/vint, 1996.

[9] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. *Proceedings of IEEE/ACM MOBICOM'99*, pages 195–206.

[10] S.J. Lee, , M. Gerla, and C.K. Toh. A simulation study of table-driven and on-demand routing protocols for mobile ad-hoc networks. *IEEE Network*, 13(4):48–54.

[11] S.J. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia. A performance comparison study of ad hoc wireless multicast protocols. *To appear in Proceedings of the IEEE Infocom 2000*, March 2000.

[12] S. McCanne. ns - LBNL network simulator. Available from http://www-nrg.ee.lbl.gov/ns/.

[13] Katia Obraczka and Gene Tsudik. Multicast routing issues in ad-hoc networks. *Proceedings of IEEE/ICUPC'98.*

[14] E. Royer and C. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. *Proceedings of the ACM Mobicom'99*, pages 207–218, August 1999.

[15] C. Wu, Y. Tay, and C. Toh. Ad hoc Multicast Routing protocol utilizing Increasing id-numberS (AMRIS). IETF manet (draft-ietf-manet-amris-spec-00.txt), 1998.

[16] X. Zeng, R. Bagrodia, and M. Gerla. GloMoSim: a library for parallel simulation of large-scale wireless networks. In *Proc. of PADS'98*, 1998. Software available from http://pcl.cs.ucla.edu/projects/domains/glomosim.html.