An Adaptive Approach to Group Communications in Multi Hop Ad Hoc Networks

Kumar Viswanath and Katia Obraczka
University of California, Santa Cruz
Santa Cruz, CA 95064
Tel:(831) 459-4308, Fax: (831) 459-4829
E-mail contact: katia@cse.ucsc.edu

Abstract

The diverse nature of MANETs makes it almost impossible for a single routing protocol to perform well under a wide range of operating conditions. Therefore the solution may be to adopt an adaptive strategy to routing and the co-existence and interoperability of different routing protocols. Considering that MANET's are generally deployed in mission critical applications the adaptive protocol should be capable of providing high reliability and timeliness guarantees in the presence of mobility. To this end we develop an adaptive flooding protocol in which nodes can dynamically switch routing mechanisms based on their perspective of network conditions. We use relative velocity as the switching criterion. Each node periodically computes its velocity relative to that of its neighbor set and based on its computation switches to one of the three modes, i.e. scoped flooding, plain flooding or hyper flooding modes . Simulations using our adaptive protocol under various realistic scenarios have shown that such protocols provide impressive benefits and can be used as the basis for developing adaptive, integrated routing techniques for MANET's of the future.

1 Introduction

In the past five years, routing in MANETs has received considerable attention from the network research community. As a result, several unicast routing protocols have been proposed. Examples include Dynamic Source Routing (DSR) [4], Ad hoc On Demand Distance Vector (AODV) [11], Destination Sequenced Distance Vector (DSDV) [10] etc. More recently, as it became clear that group-oriented services are one of the primary classes of applications targeted by MANETs, a number of multicast routing protocols for MANETs have been developed. The On Demand Multicast Routing protocol (ODMRP) [5] and Multicast-Ad hoc On Demand Distance Vector (MAODV) [13] are examples of *on demand* routing protocols, where routes are established when a source has data to send. Although these protocols perform well in constrained mobility MANETs, our analysis show that their performance degrades under more stringent network conditions such as high mobility and traffic load [9].

In general, we believe that no single multicast protocol is optimal for all MANET scenarios. The diverse nature of MANETs makes it almost impossible for a single routing protocol to perform well under a wide range of operating conditions. We envision that future internetworks will consist of a wired backbone and a collection of wired, fixed-infrastructure mobile, and ad hoc networks as leaves. We argue that a “global” multicast solution for the future internet will include specialized solutions for each type of network, as well as mechanisms for integrating these solutions. Wired multicast protocols will be used in the fixed portion of the network, while MANETs will use different multicast routing mechanisms depending on their reliability requirements and typical operating conditions (mobility, traffic load, number of sources and receivers).

Our long-term goal is to provide seamless integrated multicast service whereby a single multicast group can span all network types (fixed, fixed mobile, and different types of MANETs). This would let a given host to partake in multicast communication regardless of the currently underlying network type. To this end, hosts will have to dynamically switch among different multicast routing mechanisms as they move from one network to another. To our knowledge, there is little or no experience in the network research community in multicast protocol inter-operation (albeit, some proposals have been recently floated in the IETF [2, 14, 7]) or adaptation.

2 Focus

In this paper we concentrate on the problem of providing adaptive multicast for different types of MANET scenarios. Important MANET applications, including battlefield, disaster and emergency rescue operations, are mission critical in nature and require MANET protocols to provide high delivery and timeliness guarantees in the presence of mobility and permanent or temporary outages. The primary motivation to use flooding as the basis for our adaptive multicast routing framework, is that flooding and its variations perform considerably better than other protocols such as MAODV and ODMRP [9] over a wide range of mobility and traffic load conditions. Another factor is that flooding and its variations interoperate easily. However, we note that adaptive flooding may not necessarily be most suited for all MANET scenarios. We propose flooding variations as the first step in investigating the merits of adaptive- over non-adaptive routing mechanisms. We are also currently investigating other adaptive protocols which are not based on flooding.

The routing protocol we propose in this paper integrates scoped flooding, plain flooding, and hyper flooding into a single adaptive protocol. Nodes can switch among flooding variations based on their own perception of the current network conditions. This paper analyses the performance of the integrated routing protocol and compares it with plain flooding, and also stand alone versions of scoped and hyper flooding. We study the performance of the protocol for multicast and broadcast communication. Our study considers typical MANET scenarios such as rescue operations and conferencing. Simulations results show that the adaptive protocol performs consistently well, with reliability gains on the order of 20% for the rescue and conference scenarios. For higher mobility scenarios the adaptive protocol performed better than flooding and similar to hyper flooding with much lower routing overhead than hyper flooding.

The rest of the paper is organized as follows. In the next section, we overview the adaptive routing protocol and also describe the switching criterion used by the nodes to switch from one variation to another. Section 4 describes the simulation environment used, including a detailed description of the simulation parameters. In section 5 we present simulation results for both multicast and broadcast in artificial mobility scenarios. We also present the performance of the adaptive protocol under more realistic ad hoc scenarios in section 5.3. Finally in section 6 we present some concluding remarks as well as our future work and directions.

3 Protocol Overview

We developed an adaptive multicast protocol based on flooding and its variations. The resulting adaptive flooding protocol has three modes of operation namely, *Scoped Flooding*, *Plain Flooding*, and *Hyper Flooding* [9]. Each node is capable of operating in any of the three modes and can switch modes based on its own perspective of the network conditions. The criterion for switching modes is explained in detail later.

3.1 Scoped Flooding Mode

The basic principle behind Scoped Flooding is to reduce re-broadcasts to avoid collisions and minimize overhead. Scoped Flooding is suitable for constrained mobility environments (as in the case of conference scenario) where nodes do not move much and thus plain flooding will likely yield unnecessary redundant re-broadcasts. The work in [8] showed that a re-broadcast can provide between 0–61% additional coverage over what was already covered by a previous transmission. The coverage area of subsequent retransmission reduces drastically and drops down to 0.05% when the number of retransmissions is greater than four.

Different heuristics can be used in deciding whether to re-broadcast a packet. In our scoped flooding implementation, each node periodically transmits *hello* messages which also contain the node's neighbor list. Nodes use hello messages to update their own neighbor list and add received lists to their neighbor list table. When a node receives a broadcast, 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 neighbor list, then it does not re-broadcast the packet. In our simulations we did not require the neighbor lists to be strict subsets of one another; an 85% overlap was considered sufficient to prevent re-broadcasts.

3.2 Hyper Flooding Mode

Recall that due to the mission-critical nature of typical MANET applications, we try to provide high delivery guarantees. Hyper flooding is suitable for high mobility scenarios where high reliability is required. The price hyper flooding pays is its high overhead.

Nodes in hyper flooding mode periodically transmit hello messages. When a neighbor receives a hello message, it adds the hello message originator to its neighbor list (if the list does not already contain that node). Similarly to plain flooding, when a node receives a data packet, it simply re-broadcasts the packet and also queues it in its packet cache. Additionally, re-broadcasts are triggered by two other events: receiving a packet from a node which is not in the current neighbor list or receiving a hello message from a new node. In these cases, nodes transmit all packets in their cache. The rationale behind re-broadcasts is that "newly acquired" nodes could have possibly missed the original flooding wave on account of its mobility. This increases overall reliability by ensuring that new nodes entering the transmission region of a node receive data packets which they otherwise would have missed. Nodes periodically purge their packet cache to prevent excess re-flooding of the packets.

Parameter	Value	Description
Hold_Table_Purge_Interval	5 secs	Duration after which the Packet cache is purged
Min_Hello_Interval	3 secs	Minimum value of the Hello Interval
Max_Hello_Interval	4 secs	Maximum value of the Hello Interval
Neighbor_Table_Purge_Interval	5 secs	Duration after which neighbor entries are purged
Flooding_Interval	25 msecs	Jitter period for re-broadcasts

Table 1: Protocol Specific Parameters.

3.3 Switching Among Protocols

One fundamental issue in the design of integrated multicast is deciding when a node should switch protocols and which protocol to switch to. Every node needs to make its own decision based on its perception of current network conditions.

For the current version of adaptive flooding, we chose *relative velocity* as the preliminary criterion nodes use to switch among the different flooding variations. The rationale for using relative velocity as the switching criterion is based on our comparative performance study of plain, scoped, and hyper flooding [9]. Our simulation results indicate that protocol performance (e.g., packet delivery ratio) is highly dependent on mobility.

The proposed relative-velocity based switching criterion works as follows. Nodes send velocity (speed and direction) information as part of hello messages. Each node is then able to compute its velocity relative to its neighbors. We use only immediate neighbor information to calculate a node's relative velocity. This way a node does not have to acquire global knowledge in order to build its perception of the network current conditions. If a node's relative velocity is higher than a pre-defined threshold, the node switches to hyper flooding mode. If the relative velocity is below a lower threshold, scoped flooding is used. Otherwise, the node switches to plain flooding. Clearly, if a node detects no changes in its relative velocity, it will keep running the current flooding variant.

All nodes start off in plain flooding mode. Currently, hello messages are sent every *Hello_Interval* seconds which is also the frequency at which nodes recompute relative velocity. The upper switching threshold is set to 25 m/s, while the lower threshold is 10 m/s. These thresholds are particular to the scenarios considered and were arrived at by experimenting with different threshold values for our different scenarios. One point to be noted is that nodes acquire neighbor velocity information once every *Hello_Interval* seconds. This ensures that each node runs the same routing mechanism for a minimum period of 3 seconds before it can potentially switch to another mechanism, preventing oscillations.

Since mobility is one of the most important parameters that affect the performance of the routing protocol we chose relative velocity as the switching methodology for the adaptive protocol. We are currently investigating other switching criteria as well. Depending upon the requirements of the network, nodes can switch routing mechanisms based on the network condition parameters, such as traffic load, or multicast group characteristics, such as number of senders or receivers. Nodes can periodically monitor the network for the traffic load and switch protocols when the traffic exceeds certain thresholds. In our analysis of multicast protocols we observed that certain protocols (eg. ODMRP) performed well when the sender to receiver ratio was small. If the nodes have information about the number of receivers belonging to the multicast group they can use this information to switch routing mechanisms.

4 Methodology

We used the network simulator `ns-2` for our simulations. `ns` was originally developed at Lawrence Berkeley National Laboratory (LBNL) [6]. Currently it is being extended as part of the VINT project [3] involving USC/ISI, Xerox PARC,LBLN, and UC Berkeley. Ns is a discrete-event simulator which 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 [1]. Some of the scenarios were generated using a scenario generator for ad hoc networks [12].

4.1 Simulation Environment

All simulations consist of 50 nodes placed in a 1000x1000 meter field. Each node transmits 250 packets (256 bytes each) at various times during the simulations. We use a CBR traffic generator for some of the scenarios. Node channel bandwidth is set to 2 Mbit/sec and their transmission range is 225 meters. The total simulation duration is set to 400 seconds to ensure that senders have sufficient time to finish transmitting all data packets.

Mobility Model

Except for the conference and disaster scenarios, the mobility model used is a modified version of the random-waypoint model (also known as the bouncing ball model). In this model 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.

In the case of the conference and disaster scenarios, besides original random waypoint, we use the random motion model. The mobility patterns for these scenarios is explained in greater detail in 5.3.

Traffic Model

A constant bit rate (CBR) traffic generator was used for all the scenarios. The senders start transmitting the data packets at random times within the first 25 seconds of the simulation. For the the conference scenario, the speaker is attached to a CBR source transmitting 1 packet/sec for the first 150 seconds of the simulation and at 0.8 packets/sec for the remainder of the simulation.

Overall network traffic is maintained constant at 20 packets/sec even in the simulations where we vary the number of senders. This is accomplished by adjusting the inter-packet interval.

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 reported data point represents the average across all five runs. In our simulations the senders are chosen randomly from among all the nodes. For the multicast scenarios, receivers are also chosen at random among all nodes. The receivers join as members of the multicast group and remain as members throughout the simulation.

For the scenarios (excepting the conference and disaster scenarios) we have three different mobility groups consisting of 20, 15 and 15 members each. Each group is assigned a particular velocity and all nodes in a mobility group start so that they are positioned adjacent to other nodes of the same mobility group. In these scenarios the velocity of each group is varied across different simulation runs resulting in different values of the overall relative velocity. A node belonging to one mobility group can traverse other mobility groups changing the relative velocity of its immediate neighborhood. This may trigger the node to switch routing protocols.

We investigated both sides of the protocol reliability (i.e., delivery ratio) versus efficiency (overhead) trade-off. Packet delivery ratio measures protocol reliability 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.

5.1 Multicast Results

5.1.1 Packet Delivery

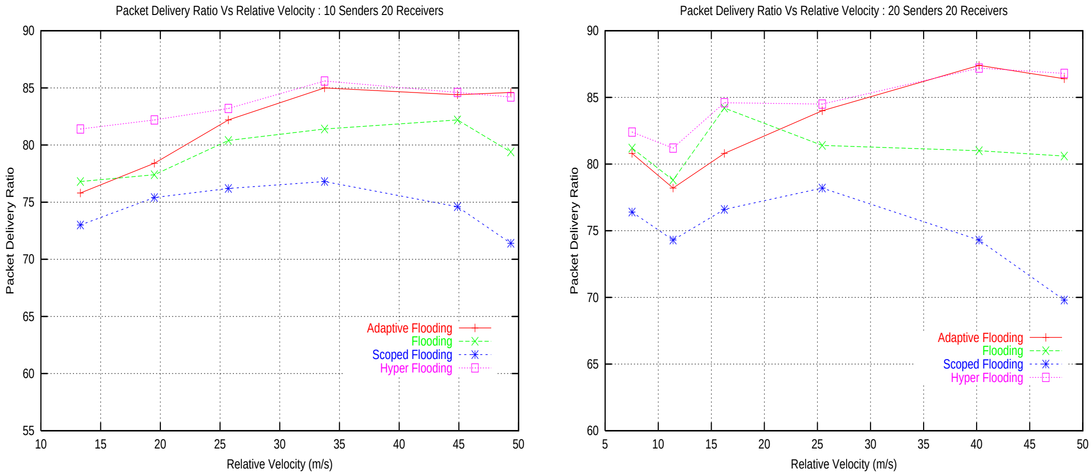
We compute packet delivery ratio as the ratio of total number of packets received by the nodes to the total number of packets transmitted times the number of receivers. For the multicast scenario, we use 10 and 20 senders and the number of receivers is set at 20 nodes. We simulate two multicast groups each with ten receivers. The receivers can be members of both multicast groups.

The graphs in figure 1 show how protocol reliability varies with node mobility which is expressed in terms of average relative velocity. The average relative velocity is computed as follows. The relative velocity of each node with respect to its neighbors is calculated throughout the duration of the simulation and is then averaged over all nodes.

It can be observed from figure 1 that initially the delivery ratio increases as the group mobility increases (which in turn increases the relative velocity) but starts degrading at higher values of the relative velocity. In our scenarios members of the same mobility group are started off at adjacent positions in the topology. This results in packet drops due to collisions and the hidden terminal problem at lower speeds. As the mobility of the nodes increases they start moving outside the transmission region of other nodes and this causes lesser packet collisions which results in an increase in the packet delivery ratio. However as the relative velocity increases beyond 40 m/s (∼145 kms/hr), increased mobility of the nodes causes them to move outside the radio range of their neighbors more frequently resulting in lower packet reception. In case of the adaptive flooding protocol, nodes rely on neighbor information to decide if they retransmit packets. Neighbor information may become stale as the mobility of the nodes increases resulting in lower packet delivery ratio at higher speeds. It can be seen from the graphs that hyper flooding performs better than the adaptive protocol at lower speeds. This is because at lower speeds adaptive flooding switches to scoped flooding mode in an attempt to reduce redundant retransmissions. As the relative velocity increases it switches to flooding and hyper flooding modes resulting in higher packet delivery ratios similar to stand alone hyper flooding. As expected at higher speeds the performance of scoped flooding starts to degrade in comparison with the other variations.

5.1.2 Routing Overhead

Routing overhead is computed as the ratio between the number of control bytes to the number of data bytes received. In adaptive flooding the control overhead accounts for the *hello messages* and also the overhead gen-



(a) 10 Senders 20 Receivers

(b) 20 Senders 20 Receivers

Figure 1: Packet Delivery Ratio as a function of Node Mobility

erated by retransmits in hyper flooding. In the case of plain flooding the overhead consists of the flooding header information required to forward the packets.

It can be observed that at low mobility (< 20 m/s) the routing overhead of adaptive flooding is lower than that of flooding but it increases as the relative velocity increases and approaches that of hyper flooding. This is a similar to the behavior observed in the delivery ratios. At low speeds the nodes use scoped flooding to forward the packets which reduces the overhead but at high relative velocities the nodes switch to flooding and hyper flooding modes which results in greater number of re-broadcasts thus increasing the routing overhead. The routing overhead of flooding and scoped flooding remain almost constant with scoped flooding having the lowest overhead.

From the results outlined above we can see that the adaptive protocol performs comparably to flooding even outperforming flooding when the relative velocity is greater than 20 m/s. For higher values of the relative velocity its performance approaches that of hyper flooding. The interesting fact is that it achieves this performance at considerably lower routing overhead than hyper flooding. Note that in case of adaptive flooding the overhead includes the cost incurred to obtain neighbor position and velocity information. At lower speeds the difference in packet delivery ratios between flooding and adaptive flooding is around 2-3% whereas the routing overhead of adaptive flooding is lower by about 10%.

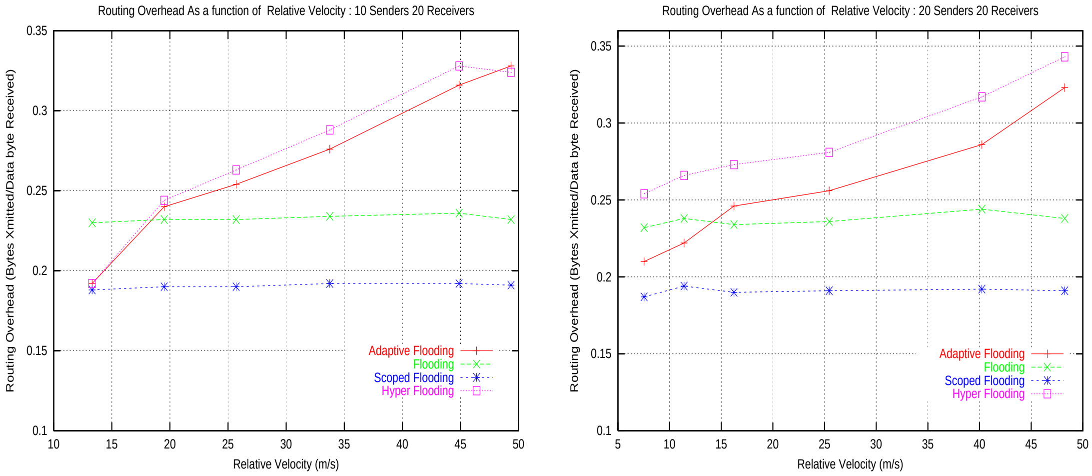
5.2 Broadcast Results

The results obtained for broadcast scenarios show similar trends to that observed in multicast scenarios. The only interesting observation from the results is that the overhead for broadcast is lower than multicast for the same number of senders. In multicast every packet which is received by the forwarding nodes does not count towards a unique data packet received. Only those data packets received by multicast group members are used in calculating the total unique packets received. However the total number of re-broadcasts remains almost the same for multicast and broadcast. This results in a lower routing overhead for the broadcast scenarios.

5.3 Other Scenarios

We have also used the adaptive flooding protocol under various realistic ad hoc scenarios. These scenarios were generated using the scenario generator [12] which uses two parameters, the *model-spec* and *scen-spec* to generate mobility scenarios. The *model-spec* consists of the mobility pattern specifications. The mobility model used can be fixed waypoint, random waypoint or brownian motion. The *scen-spec* contains precise specifications of each node in the scenario such as the mobility model, the extent of movement of the node in the topology, the offset of the node in the topology, the speed of the node and the pause time. For the conference scenario the speaker node and 20 other nodes which were randomly chosen transmitted CBR packets, while in the disaster scenario 4 nodes representing helicopters and 16 other randomly chosen nodes transmitted CBR packets.

The conference scenario consisted of a total of 50 nodes with one speaker node and three groups of audiences, i.e. *audience1*, *audience2* and the *wanderers*. The speaker



(a) Routing Overhead: 10 Senders 20 Receivers

(b) Routing Overhead: 20 Senders 20 Receivers

Figure 2: Routing Overhead as a function of relative mobility of nodes

Flooding Type	Packet Delivery Ratio %	Routing Overhead (Bytes Xmitted per Data byte received)	Average Delay (msecs)
Adaptive Flooding	90.8	0.07	33.56
Plain Flooding	70.6	0.08	28.03
Scoped Flooding	84.4	0.07	39.06
Hyper Flooding	74.8	0.097	46.32

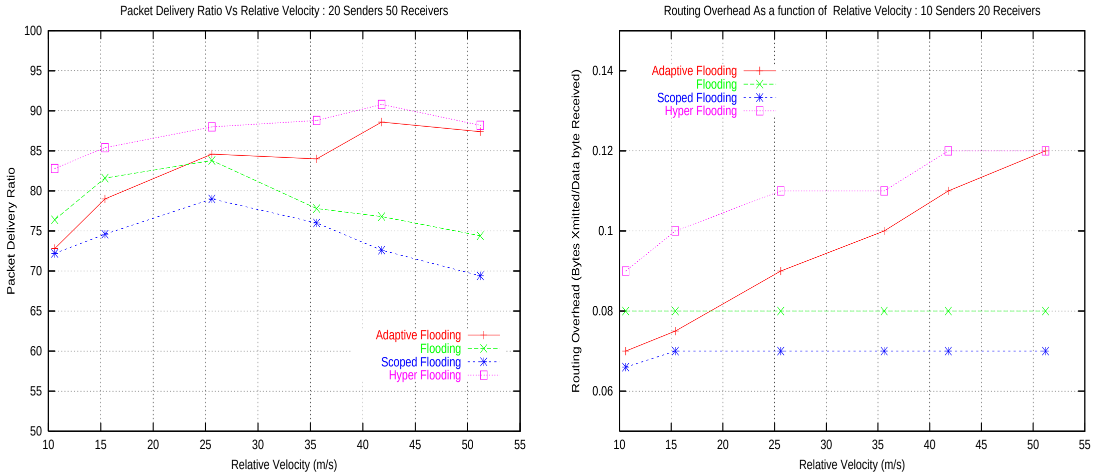
Table 2: Conference Scenario

node remained almost stationary through out the duration of the simulation. Both audience groups consisted of 20 members moving with speeds between 2-5 m/s. The movement of the audience groups was modeled using brownian motion and nodes were restricted to a subset of the topology. The final group or the *wanderers* consisted of 9 nodes who were capable of moving over the entire topology. The speeds for these nodes were randomly chosen between 1-5 m/s with pause times between 0-1 sec. The random waypoint model was used as the mobility model for all the wanderer nodes.

The second scenario was that of a disaster-rescue operation with a total of 50 nodes in a 1000 x 1000 field. This scenario consisted of 4 helicopters, a rescue team of foot soldiers and 2 teams on vehicles. The helicopters moved with speeds ranging between 0-40 m/s according to the fixed waypoint model with pause times between 0-1 secs. The first vehicle team consisted of 20 nodes while the second team consisted of 6 nodes. The members of both vehicle teams moved according to the random waypoint model with speeds ranging between 5-20 m/sec. The team of foot soldiers consisted of 20 nodes. The mobility model used for this team was the random waypoint model with speeds ranging between 5-10 m/sec and pause times between 0-2 secs. The coverage area for each team was a subset of the entire topology which did not overlap with the coverage area of any other team.

The tables 2 and 3 present the results for these two scenarios. For the conference scenario adaptive flooding delivered about 20% more packets than flooding. An interesting observation in this scenario is that it attains this higher reliability at a lower overhead than plain flooding. For this scenario the mobility of the audience groups is quite limited (0-5 m/s) and restricted to a certain area of the topology. Using flooding and hyper flooding proves to be an overkill since a large number of the retransmissions results in collisions leading to packet drops. However adaptive flooding uses the scoped flooding mode to restrict the re-broadcasts resulting in higher packet delivery ratios and lower overhead. Another observation is that for this scenario, "stand-alone" scoped flooding performs better than both flooding and hyper flooding. From the delay statistics it is observed that flooding has the lowest delay as compared to other variations. This is because flooding almost always delivers packets along the shortest path. The delay of hyper flooding is the greatest because nodes hold packets in their packet cache and retransmit them when they acquire new neighbors. Given the low mobility of the members, nodes acquire new neighbors less frequently and hence have to hold the packets in their packet cache for longer durations before they retransmit them. This increases the overall delay for hyper flooding.

In case of the disaster scenario adaptive flooding's delivery ratio is about 16% greater than plain flooding. Scoped flooding performs relatively well with 81% packet



(a) Packet Delivery: 20 Senders 50 Receivers

(b) Routing Overhead: 20 Senders 50 Receivers

Figure 3: Broadcast Results

Flooding Type	Packet Delivery Ratio %	Routing Overhead (Bytes Xmitted per Data byte received)	Average Delay (msecs)
Adaptive Flooding	86.4	0.09	55.447
Plain Flooding	69.8	0.08	28.593
Scoped Flooding	81.8	0.076	55.35
Hyper Flooding	73.3	0.112	67.86

Table 3: Disaster Scenario

delivery ratio. For this scenario the average relative velocity of the scenario is quite low. Only the choppers have significant mobility with small pause times. Flooding and hyper flooding are not as effective since redundant broadcasts compounds the hidden terminal problem and results in packet losses due to collisions. As expected the routing overhead for scoped flooding is the lowest while that of adaptive flooding is marginally higher than plain flooding.

6 Conclusions and Future Work

In summary, the paper investigates an adaptive, integrated approach to group communications. MANETs of the future will be composed of heterogenous networks spanning different network types. The diverse nature of such MANETs make it impossible for any one protocol to be be optimal under all scenarios and operating conditions. This calls for specialized multicast solutions for each type of network and the means for integrating those solutions. To this end we have proposed an adaptive approach to routing where the nodes dynamically switch routing mechanisms based on their perception of network conditions. Our adaptive protocol incorporates different variations of flooding in which nodes can switch from one mode of flooding to another depending on their mobility.

We reported simulation-driven experiments comparing the adaptive protocol with plain flooding *hyper flooding* and *scoped flooding* for both broadcast and multicast scenarios. The results demonstrate that the adaptive protocol performs consistently well in terms of both packet delivery ratios and routing overhead. At low mobility, for both multicast and broadcast adaptive flooding performs comparably to plain flooding. This performance is obtained at significantly lower routing overhead. As mobility of nodes increases adaptive flooding still performs commendably and its routing overhead approaches that of *hyper flooding*. However in case of multicast, the routing overhead is higher than broadcast. We also evaluated the adaptive protocol under typical MANET scenarios like disaster-rescue operations and conference scenario. Adaptive flooding's delivery ratio was about 16% higher than plain flooding, which was achieved at a lower routing overhead. The most important observation is that given the diversity of MANETs, adaptive protocols are capable of providing consistent performance benefits over a wide range of operating conditions. Our simulation results highlight these performance benefits and form the basis for other adaptive routing mechanisms which are not based on flooding.

We are currently working on evaluating other adaptive protocols which do not utilize flooding mechanisms. This poses interesting challenges such as:
(1) Interoperability and integration issues.

(2) Mechanisms for active, on-the-fly switching among different multicast routing mechanisms as a mobile host changes the network type it is part of.

We are also investigating the effect of using other switching criteria such as network load, number of senders and receivers and reliability. Nodes can periodically monitor the network for the traffic load and switch protocols when the traffic load exceeds certain thresholds.

References

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

[2] D. Estrin et. al. Pim border router specification for connecting pim-sm domains to a dvmrp backbone. Internet draft draft-ietf-idmr-PIM-SM-spec-09.ps, October 1996.

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

[4] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. In Tomasz Imielinski and Hank Korth, editors, *Mobile Computing*, chapter 5. Kluwer Academic Publishers, 1996.

[5] 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.

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

[7] D. Meyer. Some issues for an inter-domain multicast routing protocols. Internet draft draft-ietf-mboned-issues-03.txt, November 1997.

[8] S. Ni, Y. Tseng, Y. Chen, and J. Sheu. The broadcast storm problem in a mobile ad hoc network. *Proceedings of IEEE/ACM MOBICOM'99*, pages 151–162.

[9] Katia Obrazcka, Gene Tsudik, and Kumar Viswanath. Pushing the limits of multicast in ad hoc networks. *Proceedings of the 21st IEEE ICDCS*, pages 719–722, April 2001.

[10] C.E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance vector routing (dsdv). *Proceedings of the IEEE/ ACM SIGCOMM*, pages 234–244, September 1994.

[11] C.E. Perkins and E.M. Royer. Ad-hoc on demand distance vector routing. *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, pages 90–100, February 1999.

[12] Li Quiming. Scenario generator for manets. Available from http://www.comp.nus.edu.sg/ liqiming/fyp/scengen/index.html, April 2001.

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

[14] D. Thaler. Interoperability rules for multicast routing protocols. Internet draft draft-thaler-interop-00.ps, November 1996.