AODV-BR: Backup Routing in Ad hoc Networks

Sung-Ju Lee and Mario Gerla
Wireless Adaptive Mobility Laboratory
Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90095-1596
`http://www.cs.ucla.edu/NRL/wireless`

Abstract ***– Nodes in mobile ad hoc networks communicate with one another via packet radios on wireless multihop links. Because of node mobility and power limitations, the network topology changes frequently. Routing protocols therefore play an important role in mobile multihop network communications. A recent trend in ad hoc network routing is the reactive on-demand philosophy where routes are established only when required. Most of the protocols in this category, however, use single route and do not utilize multiple alternate paths. In this paper, we propose a scheme to improve existing on-demand routing protocols by creating a mesh and providing multiple alternate routes. Our algorithm establishes the mesh and multipaths without transmitting any extra control message. We apply our scheme to the Ad-hoc On-Demand Distance Vector (AODV) protocol and evaluate the performance improvements by simulation.***

I. INTRODUCTION

Ad hoc networking [7], [9] has emerged as one of the most focused research areas in the field of wireless networks and mobile computing. Ad hoc networks consist of hosts communicating one another with portable radios. These networks can be deployed impromptu without any wired base station or infrastructure support. In ad hoc mobile networks, routes are mainly multihop because of the limited radio propagation range, and topology changes frequently and unpredictably since each network host moves randomly. Therefore, routing is an integral part of ad hoc communications, and has received interests from many researchers. A new style of routing called "on-demand" routing has been proposed for ad hoc networks. Unlike table-driven (i.e., distance vector [12] and link state [13]) routing protocols, each node in on-demand routing does not need periodic route table update exchange and does not have a full topological view of the network. Network hosts maintain route table entries only to destinations that they communicate with. The Ad-Hoc On-Demand Distance Vector (AODV) [17] protocol, one of the on-demand routing algorithms that are receiving the most attention, however, does not utilize multiple paths. Consequently, when route disconnects, nodes of the broken route simply drop data packets because no alternate path to the destination is available until a new route is established. When the network traffic requires real time delivery (voice, for instance), dropping data packets at the intermediate nodes can be costly. Likewise, if the session is a best effort, TCP connection, packet drops may lead to slow start, timeout, and throughput degradation.

In this work, we propose an algorithm that utilizes a mesh structure to provide multiple alternate paths to existing on-demand routing protocols without producing additional control messages. Having multiple alternate paths in ad hoc networks is beneficial because wireless networks are prone to route breaks resulting from node mobility, fading environment, signal interference, high error rate, and packet collisions. It is also important to generate multiple routes without propagating more control messages than when building only single route. Minimizing the number of packet transmissions is critical in ad hoc networks with limited bandwidth and shared wireless medium.

There are a couple of multicast protocols that rely on the mesh topology for communications between multicast members: the On-Demand Multicast Routing Protocol (ODMRP) [1], [10] and the Core Assisted Mesh Protocol (CAMP) [4], [5]. Because of the richer connectivity of a mesh, these protocols have been shown to perform well compared with tree based single route protocols [4], [11]. When unicasting, we can also use alternate paths provided by the mesh to deliver data packets when the primary route becomes disconnected. Our scheme is inspired by the duct routing scheme [20] proposed in the early 1980s. Duct routing, however, suffers from some limitations; data packets are propagated in duplicates through multiple routes at all instances, thus creating excessive redundancy that causes congestion and collision. In our algorithm, on the contrary, multiple alternate paths are utilized only when the primary route is disconnected. Another difference between the two algorithms is that our protocol builds routes on demand. Wang and Crowcroft [22] also proposed a protocol that uses an alternate path only when data packets are not deliverable through the primary route. That scheme however, is based on Shortest Path First (SPF) algorithm for wire-line networks. There are some related work on protocols using multiple routes in ad hoc networks; the scheme by Nasipuri and Das [14], [15], Temporally-Ordered Routing Algorithm (TORA) [16], and Routing On-demand Acyclic Multipath (ROAM) [18], but these algorithms require additional control message to construct and maintain alternate

This work was funded in part by the Defense Advanced Research Projects Agency (DARPA) under contract DAAB07-97-C-D321, as a part of the Global Mobile Information Systems (GloMo) program.

routes.

The rest of the paper is organized as follows. Section II illustrates the protocol operation in detail. Performance evaluation via simulation is presented in Section III and concluding remarks are made in Section IV.

II. PROTOCOL CONCEPT

In this section, we present the operation details of our scheme. Since the purpose of our study is to improve the performance of existing on-demand protocols (specifically AODV in this paper), our protocol description is based on AODV. Our modifications to AODV for applying our scheme is also introduced.

A. Route Construction

Our scheme can be incorporated with reactive routing protocols that build routes on demand via a query and reply procedure. Our algorithm does not require any modification to the AODV's RREQ (route request) propagation process. When a source needs to initiate a data session to a destination but does not have any route information, it searches a route by flooding a ROUTE REQUEST (RREQ) packet. Each RREQ packet has a unique identifier so that nodes can detect and drop duplicate packets. An intermediate node, upon receiving a non-duplicate RREQ, records the previous hop and the source node information in its route table (i.e., backward learning). It then broadcasts the packet or sends back a ROUTE REPLY (RREP) packet to the source if it has a route to the destination. The destination node sends a RREP via the selected route when it receives the first RREQ or subsequent RREQs that traversed a better route (in AODV for instance, fresher or shorter route) than the previously replied route.

The mesh structure and alternate paths are established during the route reply phase. We slightly modify the AODV protocol in this procedure. Taking advantage of the broadcast nature of wireless communications, a node promiscuously "overhears" packets that are transmitted by their neighboring nodes. From these packets, a node obtains alternate path information and becomes part of the mesh as follows. When a node that is not part of the route overhears a RREP packet not directed to itself transmitted by a neighbor (on the primary route), it records that neighbor as the next hop to the destination in its *alternate route table*. A node may receive numerous RREPs for the same route if the node is within the radio propagation range of more than one intermediate node of the primary route. In this situation, the node chooses the best route among them and inserts it to the alternate route table. When the RREP packet reaches the source of the route, the primary route between the source and the destination is established and ready for use. Nodes that have an entry to the destination in their alternate route table are part of the mesh. The primary route and alternate routes together establish a mesh structure that looks similar to a fish bone (see Fig. 1).

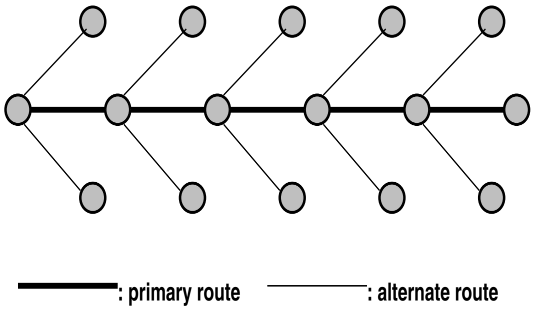

Fig. 1. Multiple routes forming a fish bone structure.

B. Route Maintenance and Mesh Routes

Data packets are delivered through the primary route unless there is a route disconnection. When a node detects a link break (for example, receives a link layer feedback signal from the MAC protocol,[1] does not receive passive acknowledgments,[2] does not receive hello packets for a certain period of time, etc.), it performs a one hop data broadcast to its immediate neighbors. The node specifies in the data header that the link is disconnected and thus the packet is candidate for "alternate routing." Upon receiving this packet, neighbor nodes that have an entry for the destination in their alternate route table, unicast the packet to their next hop node. Data packets therefore can be delivered through one or more alternate routes and are not dropped when route breaks occur. To prevent packets from tracing a loop, these mesh nodes forward the data packet only if the packet is not received from their next hop to the destination and is not a duplicate. When a node of the primary route receives the data packet from alternate routes, it operates normally and forwards the packet to its next hop when the packet is not a duplicate. The node that detected the link break also sends a ROUTE ERROR (RERR) packet to the source to initiate a route rediscovery. The reason for reconstructing a new route instead of continuously using the alternate paths is to build a fresh and optimal route that reflects the current network situation and topology.

Our alternate route utilization mechanism is similar to that of DSR (Dynamic Source Routing) [8], but has the following differences. Our scheme uses the mesh link only to "go around" the broken part of the route. In DSR, on the other hand, the node that detects a route disconnection can salvage the data by replacing in the source header the entire remaining route to the destination with an alternate route stored in its route cache. The DSR backup scheme requires considerable

---

[1] MAC protocols such as MACAW [3] and IEEE 802.11 [6] have this capability.

[2] This technique was introduced by Jubin and Tornow in their early work on packet radio networks [9]

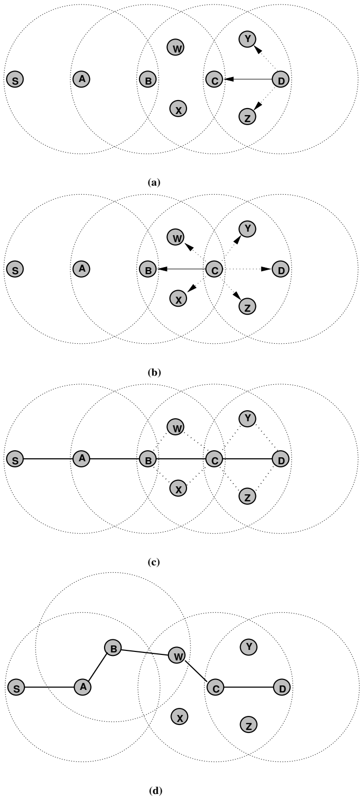

Fig. 2. Multiple route construction and their usage: (a) node D sends a RREP, (b) node C forwards the RREP, (c) the primary route and alternate routes are established, (d) data packet is delivered via an alternate route when the primary route is disconnected.

cache storage overhead. Another difference is that the node of DSR sends a RERR packet to the source only when it has no alternate route and cannot salvage the data. Therefore, routes in DSR are refreshed less often compared with our scheme.

In AODV, a route is timed out when it is not used and updated for a certain duration of time. We use the same technique for timing out alternate routes. Nodes that provide alternate paths overhear data packets and if the packet was transmitted by the next hop to the destination as indicated in their alternate route table, they update the path. If an alternate route is not updated during the timeout interval, the node removes the path from the table.

C. Example

Fig. 2 is an example showing how the mesh and alternate routes are constructed and used in data delivery. When the RREQ reaches the destination node D, the primary route $<S$-A-B-C-$D>$ is selected. The destination D sends a RREP to node C. Nodes Y and Z, who are within the propagation range of D, overhear the packet and insert an entry into their alternate route table. This process is shown in Fig. 2(a). After receiving this RREP, only node C relays the packet to node B since it is part of the route. Again, one hop neighboring nodes can overhear the packet. Nodes W and X record node C as the next hop to the destination D in their alternate route table. Node Y and Z, on the contrary, do not update their table since they already have a path to D. Likewise, node D does not react to the RREP transmission by node C since it is the destination (and part of the route). Fig. 2(c) shows the state when the RREP reaches the source node S and builds the primary and multiple alternate routes. Fig. 2(d) illustrates the usage of an alternate path when the primary route gets disconnected. Node B moved out the radio range of its next hop node C. After receiving the data packet from node A, node B forwards it to node C. The packet will fail to be delivered since node C is not reachable. Node B then broadcasts the packet to its neighbors for alternate paths to salvage the data. Nodes A and W receive the packet, but node A drops it upon duplicate detection. Node W, on the other hand, recognizes the primary route disconnection by reading the packet header. It looks up in its alternate route table and finds C as its next hop to the destination. It unicasts the packet to node C, and eventually the packet reaches the destination.

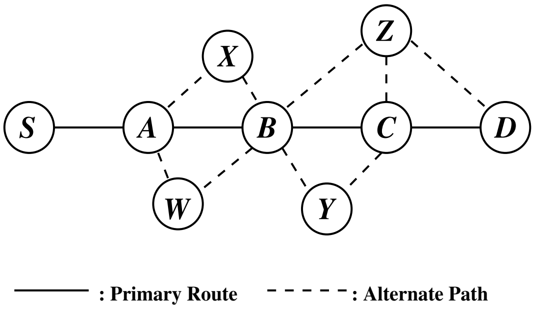

Fig. 3. An alternate path with the same path length as the primary route.

In the above example, the destination of the route receives the data packet via an alterate route that is longer in hop distance than the primary route. There can be instances where alternate routes have the same path length as the primary route. In Fig. 3, for example, when the link between nodes B and C fails, node Z of the mesh forwards the packet from node B directly to the destination node D without sending it through node C. Therefore, the packet is delivered through the path $<S$-A-B-Z-$D>$ that has the same hop length as the primary route $<S$-A-B-C-$D>$.

D. A Variant

To improve the efficiency of the protocol, mesh nodes optionally may not relay the data packet when they overhear other salvaged transmissions. Let us use Fig. 3 as an example again. Consider that node A has failed to verify the packet delivery to node B. When node A seeks help from neighboring nodes in the mesh, nodes W and X are available. Assume that node W

receives the packet first and sends it to node B. Node X hears the transmission from node W to node B if it is within the radio propagation range of node W. Node X can choose not to relay the data packet from node A, since node W already attempted to salvage the data. In our current implementation however, node X still sends the data packet to node B for added redundancy since node B might have moved out of the radio range of node W.

III. SIMULATION EXPERIMENTS

A. Simulation Environment

To evaluate the performance improvements made by our backup routing, we compare the simulation results of the AODV protocol with and without applying our scheme. In this section, we termed the AODV protocol that applied our algorithm as AODV-BR (AODV with Backup Routes).

The simulator was implemented within the Global Mobile Simulation (GloMoSim) library [21]. The GloMoSim library is a scalable simulation environment for wireless network systems using the parallel discrete-event simulation capability provided by PARSEC [2]. Our simulation modeled a network of 50 mobile hosts placed randomly within a 1500 meter × 300 meter area. Radio propagation range for each node was 250 meters and channel capacity was 2 Mb/s. Each run executed for 300 seconds of simulation time.

A free space propagation model with a threshold cutoff [19] was used in our experiments. In the free space model, the power of a signal attenuates as $1/d^2$ where d is the distance between radios. In the radio model, we assumed the ability of a radio to lock on to a sufficiently strong signal in the presence of interfering signals, i.e., radio capture. If the capture ratio (the minimum ratio of an arriving packet's signal strength relative to those of other colliding packets) [19] was greater than the predefined threshold value, the arriving packet was received while other interfering packets were dropped.

We used the IEEE 802.11 Distributed Coordination Function (DCF) [6] as the medium access control protocol. DCF is the mode which allows mobiles to share the wireless channel in an ad hoc configuration. The specific access scheme is Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) with acknowledgments. The nodes make use of Request To Send/Clear To Send (RTS/CTS) channel reservation control frames for unicast, virtual carrier sense, and fragmentation of packets larger than a given threshold. By setting timers based upon the reservations in RTS/CTS packets, the virtual carrier sense augments the physical carrier sense in determining when mobile nodes perceive that the medium is busy. Fragmentation is useful in the presence of high bit error and loss rates, as it reduces the size of the data units that need to be retransmitted. We did not employ fragmentation because our data packets were small enough that the additional overhead would reduce overall network throughput.

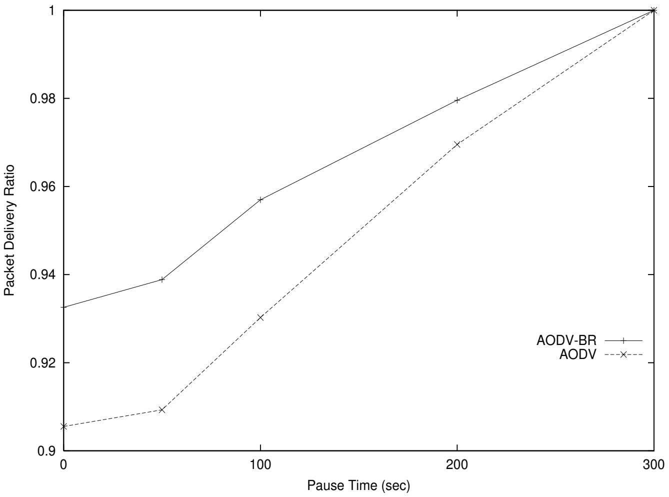

Fig. 4. Packet delivery ratio.

A traffic generator was developed to simulate constant bit rate sources. The sources and the destinations are randomly selected with uniform probabilities. There were ten data sessions, each with the traffic rate of four packets per second. The size of data payload was 512 bytes.

The random waypoint mobility model [8] was used. Each node randomly selects a position, and moves toward that location with a speed between the minimum and the maximum speed. Once it reaches that position, it becomes stationary for a predefined pause time. After that pause time, it selects another position and repeats the process. We varied the pause time to simulate different mobility degrees. Longer pause time implies less mobility. The minimum and the maximum speed were zero and 20 m/s, respectively.

B. Results and Analysis

Fig. 4 shows the throughput in packet delivery ratio. We can see that our scheme improves the throughput performance of AODV. As the mobility increases (i.e., pause time gets shorter), the performance gain by alternate routes becomes more significant. Because AMR attempts to use multiple alternate paths for data delivery in the presence of route breaks, the protocol is able to deliver more packets to the destination than AODV. AODV simply drops data packets when routes are disconnected. AODV-BR also has some packet losses. Alternate paths may be broken as well as the primary route because of mobility, or be unavailable and not discovered during the route reply phase. Moreover, packets can be lost because of collisions and contention problems.

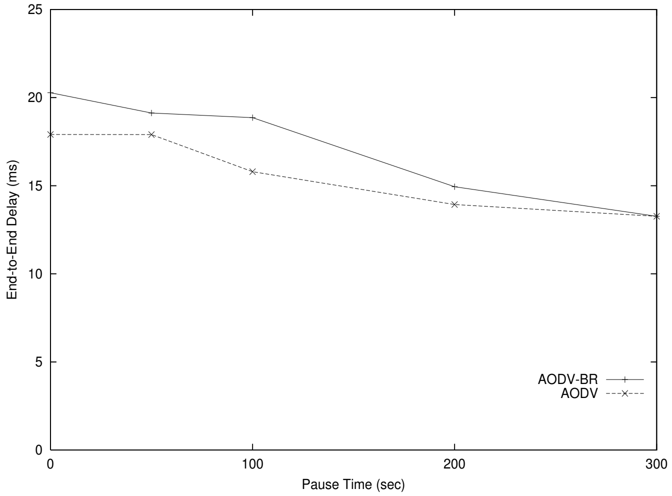

Fig. 5. End-to-end delay.

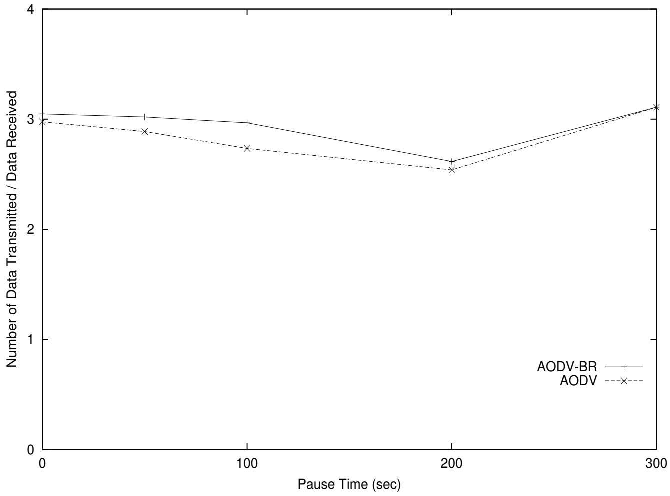

Fig. 6. The number of data transmitted per data delivery.

End-to-end delay is presented in Fig. 5. As expected, AODV-BR has longer delays than AODV. We can only measure delays for data packets that survived to reach their destination. AODV-BR delivers more packets, and those packets that are delivered in AODV-BR but not in AODV, take alternate and possibly longer hop routes. AODV-BR having longer delays than AODV does not represent its ineffectiveness since these protocols use the same primary route.

Because AODV-BR and AODV both have the same amount of control message overhead, we used a different metric for efficiency evaluation. We present the number of hop-wise data transmission per data delivery to the destination in Fig. 6. We can observe that AODV-BR transmits slightly more data packets than AODV. There are two reasons for this result. First, when route break occurs, AODV-BR uses longer alternate paths to deliver packets that are dropped in AODV, as explained above. Second, when there are multiple alternate paths, redundancy is created and hence increases the number of data transmission. We can learn from this result that we need to sacrifice efficiency in order to improve throughput and protocol effectiveness.

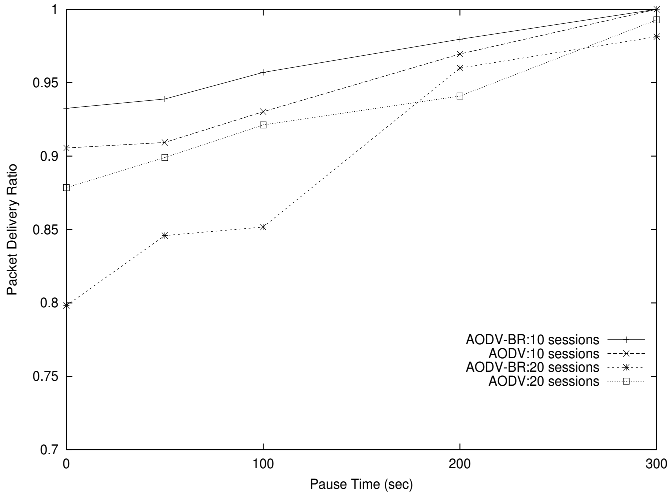

Fig. 7. Packet delivery ratio with increased number of data sessions.

To investigate whether our scheme is still effective in heavy traffic situations, we increased the traffic load. In one experiment, we increased the number of data sessions with each session having the same traffic rate of four packets per second. In another experiment, we kept the number of sessions constant to ten and varied the traffic rate. Fig. 7 shows the packet delivery ratio for ten sessions and twenty sessions. Ten sessions results are from Fig. 4. We can see that the effectiveness of both protocols decreases because of the increase in packet collisions when there are more data sessions. Even though AODV-BR improved the performance of AODV in a ten sessions network, it actually performs worse than AODV when we doubled the number of data sessions. Since there are more communication routes, AODV-BR generates more alternate routes accordingly. When the mobility rate is high, many route disconnects occur and a number of nodes that are part of the mesh transmit data packets. These transmissions cause collision and make the scheme lose its effectiveness. In fact, data packets traversing through alternate paths collide with packets using primary routes and degrade the overall throughput.

Fig. 8 illustrates the packet delivery ratio with various data session traffic rate. Similar to Fig. 7, throughput of both schemes degrades as the network traffic load increases. AODV-BR still performs better than AODV when each source

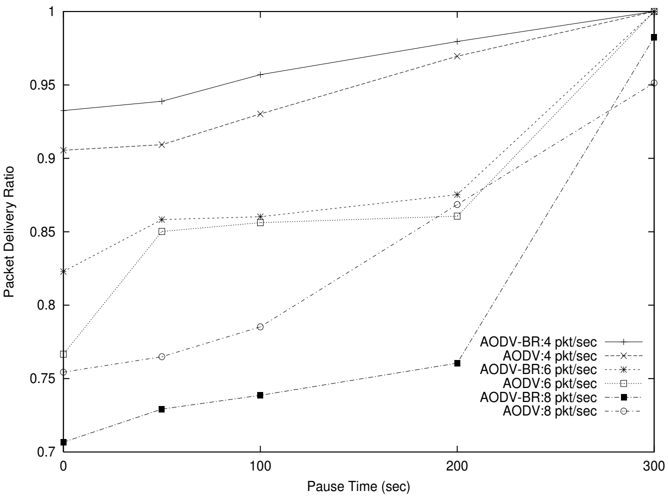

Fig. 8. Packet delivery ratio with increased number of data rate.

sends six packets per second. As we further increase the data traffic however, AODV-BR cannot deliver more data packets than AODV. We can explain this behavior in the same way as we analyzed results in Fig. 7. Basically, AODV-BR is not as effective and efficient in heavily loaded network as in lightly loaded network because of increased packet collisions and channel contention.

IV. CONCLUSION

We presented a scheme that utilizes a mesh structure and alternate paths. Our scheme can be incorporated into any ad hoc on-demand unicast routing protocol to improve reliable packet delivery in the face of node movements and route breaks. The mesh configuration provides multiple alternate routes and is constructed without yielding any extra overhead. Alternate routes are utilized only when data packets cannot be delivered through the primary route. As a case study, we applied our algorithm to AODV and measured performance improvements. Simulation results indicated that our technique provides robustness to mobility and enhances protocol performance. We also learned that however, our scheme does not perform well under heavy traffic networks. We are currently investigating ways to make our protocol robust to traffic load. Additionally, we plan to further evaluate our scheme by using more detailed and realistic channel models with fading and obstacles in the simulation. We believe the advantage of providing backup routes will be significant in those environments.

References

[1] S.H. Bae, S.-J. Lee, W. Su, and M. Gerla, "The Design, Implementation, and Performance Evaluation of the On-Demand Multicast Routing Protocol in Multihop Wireless Networks," *IEEE Network*, Special Issue on Multicasting Empowering the Next Generation Internet, vol. 14, no. 1, January/February 2000, pp. 70-77.

[2] R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, and H.Y. Song, "PARSEC: A Parallel Simulation Environment for Complex Systems," *IEEE Computer*, vol. 31, no. 10, October 1998, pp.77-85.

[3] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs," *Proceedings of ACM SIGCOMM'94*, London, UK, September 1994, pp. 212-225.

[4] J.J. Garcia-Luna-Aceves and E.L. Madruga, "The Core-Assisted Mesh Protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, August 1999, pp. 1380-1394.

[5] J.J. Garcia-Luna-Aceves and E.L. Madruga, "A Multicast Routing Protocol for Ad-Hoc Networks," *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, New York, NY, March 1999, pp. 784-792.

[6] IEEE Computer Society LAN MAN Standards Committee, *Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY) Specification*, IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.

[7] Internet Engineering Task Force (IETF) Mobile Ad Hoc Networks (MANET) Working Group Charter, Chaired by Joseph Macker and M. Scott Corson, `http://www.ietf.org/html.charters/manet-charter.html`.

[8] D.B. Johnson and D.A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," In *Mobile Computing*, edited by Tomasz Imielinski and Hank Korth, Kluwer Academic Publishers, 1996, pp. 153-181.

[9] J. Jubin and J.D. Tornow, "The DARPA Packet Radio Network Protocols," *Proceedings of the IEEE*, vol. 75, no. 1, January 1987, pp. 21-32.

[10] S.-J. Lee, M. Gerla, and C.-C. Chiang, "On-Demand Multicast Routing Protocol," *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, New Orleans, LA, September 1999, pp. 1298-1302.

[11] S.-J. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, "A Performance Comparison Study of Ad Hoc Wireless Multicast Protocols," *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, Tel Aviv, Israel, March 2000, pp. 565-574.

[12] G.S. Malkin and M.E. Steenstrup, "Distance-Vector Routing," In *Routing in Communications Networks*, edited by M.E. Steenstrup, Prentice Hall, 1995, pp. 83-98.

[13] J. Moy, "Link-State Routing," In *Routing in Communications Networks*, edited by M.E. Steenstrup, Prentice Hall, 1995, pp. 135-157.

[14] A. Nasipuri, R. Castaneda, and S.R. Das, "Performance of Multipath Routing for On-Demand Protocols in Mobile Ad Hoc Networks," *ACM/Baltzer Journal of Mobile Networks and Applications (MONET)*, to appear, 2000.

[15] A. Nasipuri and S.R. Das, "On-Demand Multipath Routing for Mobile Ad Hoc Networks," *Proceedings of the* 8^{th} *Annual IEEE Internation Conference on Computer Communications and Networks (ICCCN)*, Boston, MA, October 1999, pp. 64-70.

[16] V.D. Park and M.S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, Kobe, Japan, April 1997, pp. 1405-1413.

[17] C.E. Perkins and E.M. Royer, "Ad-Hoc On Demand Distance Vector Routing," *Proceedings of the* 2^{nd} *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, New Orleans, LA, February 1999, pp. 90-100.

[18] J. Raju and J.J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-Free Multipath Routing," *Proceedings of the* 8^{th} *Annual IEEE International Conference on Computer Communications and Networks (ICCCN)*, Boston, MA, October 1999, pp. 522-527.

[19] T.S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, Upper Saddle River, NJ, Oct. 1995.

[20] N. Shacham, E.J. Craighill, and A.A. Poggio, "Speech Transport in Packet-Radio Networks with Mobile Nodes," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, no. 6, December 1983, pp. 1084-1097.

[21] University of California Los Angeles Computer Science Department Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory, *GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network Systems*, `http://pcl.cs.ucla.edu/projects/domains/glomosim.html`.

[22] Z. Wang and J. Crowcroft, "Shortest Path First with Emergency Exits," *Proceedings of ACM SIGCOMM Symposium on Communications Architectures and Protocols*, Philadelphia, PA, September 1990, pp. 166-176.