The Dynamic Source Routing Protocol for Multi-Hop Ad Hoc Networks

Thesis Proposal:

The Dynamic Source Routing Protocol for Multi-Hop Ad Hoc Networks

David A. Maltz

November 5, 1999

Abstract

An ad hoc network is a collection of wireless mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration. Routing protocols used inside ad hoc networks must be prepared to automatically adjust to an environment that can vary between the extremes of high mobility with low bandwidth, and low mobility with high bandwidth. Over the past 5 years, I have developed a routing protocol called *Dynamic Source Routing* (DSR) that shows extremely good performance in the environment of an ad hoc network. As a result of its unique design, the protocol adapts quickly to routing changes when host movement is frequent, yet requires little or no overhead during periods in which hosts move less frequently. My thesis will describe the DSR protocol, and the trade-offs encountered in its design. By presenting a detailed analysis of DSR's behavior in a variety of situations, I will generalize the lessons learned from DSR so they can be directly applied to the many other new routing protocols that have adopted the basic DSR framework.

1. Introduction

The need to exchange information outside the typical wired office environment is growing. For example, a class of students may need to interact during a lecture; business associates serendipitously meeting in an airport may wish to share files; or disaster recovery personnel may need to coordinate relief information after a hurricane or flood. In a typical ad hoc network, potentially mobile nodes come together for a period of time to exchange information. While exchanging information, the nodes may continue to move, so the network must be prepared to adapt continually. Since networking infrastructure such as repeaters or base-stations will frequently be either undesirable or not directly reachable, the nodes must be prepared to organize themselves into a network and establish routes among themselves without any outside support.

In the simplest cases, the nodes may be able to communicate directly with each other. However, ad hoc networks must also support communication between nodes that are only indirectly connected by a series of hops through other nodes. In general, an ad hoc network looks like a network in which every mobile node is potentially a router, and all nodes run a routing protocol. Unfortunately, standard routing algorithms work poorly in a mobile environment in which network topology changes may be drastic and frequent as the individual mobile nodes move.

In my thesis, I will argue that there are two keys to designing a routing protocol that operates successfully among the challenges of an ad hoc network. First, the protocol must be fundamentally *on-demand*, meaning that it must react to changes in the environment only as they are discovered to cause a problem. Second, the protocol must limit the number of nodes that are required to share consistent state information, since it is extremely expensive or impossible to maintain a distributed data structure in a consistent state across all the nodes in a rapidly changing network. I will describe the Dynamic Source Routing (DSR) [17, 16, 2] I developed to embody these principals, and explain how I decided which of the many mechanisms I investigated to include in the protocol. I will demonstrate that DSR delivers excellent routing performance across a wide range of ad hoc network environments. Finally, I will dissect DSR into its component mechanisms, showing how they combine to give DSR its overall performance, and enabling its lessons to be generalized to other protocols.

The remainder of this proposal is organized as follows. The next section describes the challenges a routing protocol for ad hoc networks must overcome. Section 3 presents a brief description of DSR, which is my solution to the challenges. Sections 4 and 5 end the proposal by explaining my plans for evaluating the DSR protocol and generalizing its lessons, and discussing the related work.

2. Problem Statement: Routing in Ad Hoc Networks

The basic routing problem is that of finding an ordered series of intermediate nodes that can transport a packet across a network from its source to its destination by forwarding the packet along the series of intermediate nodes. In traditional hop-by-hop solutions to the routing problem, each node in the network maintains a routing table. For each known destination, the routing table lists the next node to which a packet for the destination should be sent.

The routing table at each node can be thought of as part of a distributed data structure that, taken together, represents the topology of the network. The goal of the routing protocol is to ensure that the overall data structure contains a consistent and correct view of the actual network topology. If the routing tables at some nodes were to become inconsistent, then packets can loop in the network. If the routing tables were to contain incorrect information, then packets can be dropped. The problem of maintaining a consistent and correct view becomes harder as there is an increase in the number of nodes whose information must be consistent, and as the rate of change in the true topology increases.

The challenge in creating a routing protocol for ad hoc networks is to design a single protocol that can adapt to the wide variety of conditions present inside ad hoc networks. For example, the bandwidth available between two nodes in the network may vary from more than 10 Mbps, when using high-speed network interfaces with little interference, to 10 Kbps or less when using low-speed network interfaces or when there is significant interference from outside sources or other nodes' transmitters. Similarly, nodes in an ad hoc network may alternate between periods when they are stationary with respect to each other and periods when they change topology rapidly. Conditions across a single network may also vary, so that some nodes are slow moving, while others change location rapidly.

The routing protocol must perform efficiently in environments in which nodes are stationary and bandwidth is not a limiting factor. Yet, the same protocol must still function efficiently when the bandwidth available between nodes is low and the level of mobility and topology change high. Because it is often impossible to know *a priori* what environment the protocol will find itself in, and the environment can change unpredictably, the routing protocol must be able to adapt automatically.

Most routing protocols include at least some *periodic* behaviors, meaning that there are protocol operations that are performed regularly at some interval regardless of outside events. These periodic behaviors typically limit the ability of the protocols to adapt to changing environments. If the periodic interval is set too short, the protocol will be inefficient as it performs its activities more often than required to react to changes in the network topology. If the periodic interval is set too long, the protocol will not react sufficiently quickly to changes in the network topology, and packets will be lost.

Periodic protocols can be designed to adjust their periodic interval to try to match the rate of change in the network, but this approach will suffer from the overhead associated with the tuning mechanism and the lag between a change in conditions and the selection of a new periodic interval. In the worst case, which consists of bursts of topology change followed by stable periods, adapting the periodic interval could result in the protocol using a long interval during the burst periods and a short interval in the stable periods. This worst case may be fairly common, for example, as when a group of people enter a room for a meeting, are seated for the course of the meeting, and then stand up to leave at the end.

The alternative to a periodic routing protocol is one that operates in an *on-demand* fashion. On-demand protocols are based on the premise that if a problem or inconsistent state can be detected before it causes permanent harm, then all work to correct a problem or maintain consistent state can be delayed until it is proven to be needed. They operate using the same "lazy" philosophy as optimistic algorithms.

The Dynamic Source Routing protocol (DSR) is unique among the current set of routing protocols for ad hoc networks in the way it avoids periodic behavior, and in the way it solves the routing information consistency problem. First, DSR is completely on-demand, which causes the overhead of the protocol to automatically scale directly with the need for reaction to topology change. This dramatically lowers the overhead of the protocol by eliminating the need for any periodic activities, such as the route advertisement and neighbor detection packets that are present in other protocols. Second, DSR uses source routes to control the forwarding of packets through the network. The key advantage of a source routing design is that intermediate nodes do not need to maintain consistent global routing information, since the packets themselves already contain all the routing decisions. Beyond this, the source route on each packet describes a path through the network. Therefore, with a cost of no additional packets, every node overhearing a source route learns a way to reach all nodes listed on the route.

3. Overview of the DSR Protocol

Working with David Johnson, I have designed, developed, and simulated the Dynamic Source Routing (DSR) protocol for multi-hop ad hoc networks [15, 17, 16, 2]. Dynamic Source Routing (DSR) is in a different class than most other routing protocols in that it uses source routes supplied by a packet's originator to determine a packet's path through the network, rather than using independent hop-by-hop routing decisions made by each node that receives the packet.

In a source routing design, each packet to be routed through the network carries in its header the complete, ordered list of nodes through which the packet must pass. As mentioned in Section 2, the key advantage of a source routing design is that intermediate hops do not need to maintain fresh routing information in order to route packets they receive, since the packets themselves already contain all the routing decisions. This advantage eliminates the need for the periodic route advertisement and neighbor detection packets present in other protocols, and allows each node to ignore all topology changes except those that effect the source routes it is currently using.

3.1. The Basic Design of DSR

Figure 1 shows the basic operations of the DSR protocol, which we divide into two mechanisms: *Route Discovery* and *Route Maintenance*. Route Discovery is the mechanism by which a node **S** wishing to send a packet to a destination **D** obtains a source route to **D**. To perform a Route Discovery, the source node **S** broadcasts a ROUTE REQUEST packet that is flood-filled through the network in a controlled manner and is answered by a ROUTE REPLY packet from the destination node or from another node that knows a route to the destination. Nodes keep the routes they learn in a *Route Cache*, which can store multiple routes to the same destination. Before repropagating a ROUTE REQUEST, nodes check their Route Cache for a route that could be used to answer the REQUEST. Route Discovery is initiated only on-demand, meaning that nodes do not expend effort to obtain or maintain routes to nodes they are not currently using for communication.

Route Maintenance is the mechanism by which a packet's sender **S** is able to detect if the network topology has changed such that it can no longer use its route to the destination **D**. For example, a route becomes broken if two nodes listed as neighbors on the route have moved out of range of each other. When Route Maintenance indicates a source route is broken, it notifies **S** with a ROUTE ERROR packet identifying the broken hop in the route. For subsequent packets, **S** can use any other route it happens to know to **D**, or it can invoke Route Discovery again to find a new route.

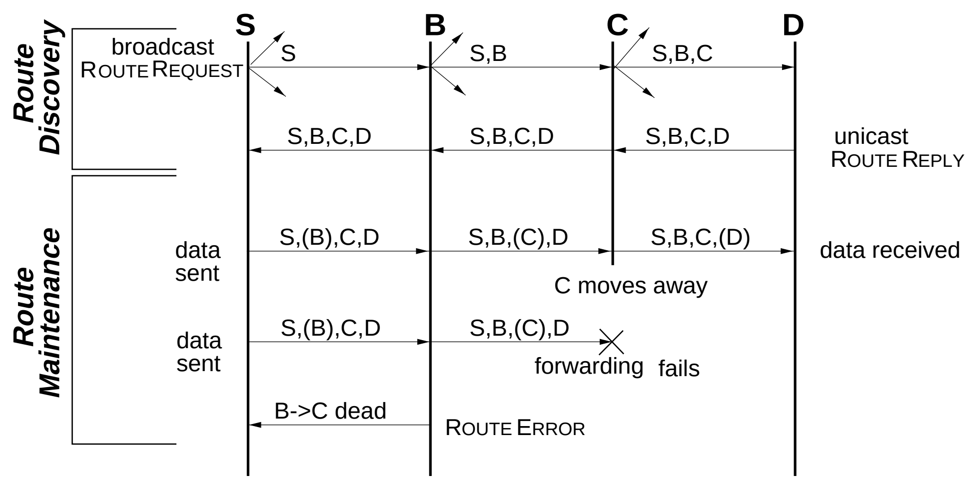


Figure 1 Basic operation of the DSR protocol showing the building of a source route during the propagation of a ROUTE REQUEST, the source route's return in a ROUTE REPLY, its use in forwarding data, and the sending of a ROUTE ERROR upon forwarding failure. The next hop is indicated by the address in parentheses.

3.2. Optimizations to DSR

Since 1994 I have iteratively refined the DSR protocol via designing, simulating, and analyzing optimizations. This work has fleshed out the exact rules required for a correct protocol (e.g., when is it safe for a node reply to a ROUTE REQUEST for another node), and led to a collection of improvements to Route Discovery and Route Maintenance [22, 16]. My optimizations and extensions to the protocol include:

- Several different Route Cache data structures and algorithms, and the rules that govern how the Route Cache can be used to limit the repropagation of Route Discoveries.
- Allowing source routes to be cheaply shortened if communicating nodes move closer together
- Salvaging packets that are sent with an incorrect source route, so that Route Maintenance has time to react without dropping packets.
- Improving the speed at which stale data is removed from the nodes' caches.
- The use of control message piggybacking to support asymmetric routes.
- The two-phase structure of Route Discovery.
- Techniques for avoiding ROUTE REPLY storms.

3.3. The Performance of DSR

The combined effect of the optimizations gives DSR significantly better performance than three other ad hoc network routing protocols, when measured on three key metrics. In a detailed simulation study [3], the four routing protocols were each run on networks of 50 nodes. The protocols were subjected to identical packet workloads while the nodes moved in identical patterns. The experimental design allows us to directly compare the performance of the protocols, since they were challenged in an identical fashion. The movement patterns were created by the Random Waypoint algorithm [16] I developed that tends to stress all operating modes of the routing protocols by creating many widely varying network topologies.

A brief description of each of the four routing protocols follows below. Each protocol was selected for inclusion in the study from the set of published proposals for two reasons: each represents a very different point in the design space of ad hoc network routing protocols, and each has been described in enough detail by their designers to permit simulation.

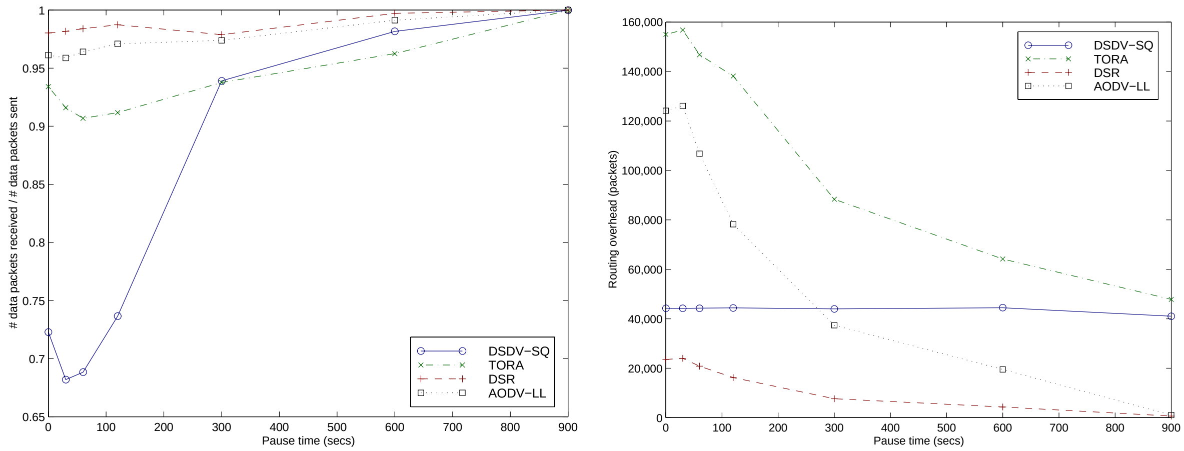
DSDV: Destination Sequenced Distance Vector [27]. A distance vector algorithm modified to guarantee loop freedom even in the face of the rapid topology changes of an ad hoc network. After experimenting with the protocol, we found that a change in the rule for sending triggered updates resulted in better performance. We report the performance of this improved variant, which we call DSDV-SQ.

TORA: Temporally-Ordered Routing Algorithm [5, 24, 25, 26]. A link reversal algorithm that emphasizes minimizing reaction to topology change.

DSR: Dynamic Source Routing. Our routing protocol, as described briefly in Sections 3.1 and 3.2.

AODV: Ad Hoc On Demand Distance Vector [29, 28]. A combination of DSDV and DSR, designed to use the on-demand nature of DSR's Route Discovery mechanism to adapt to ad hoc network topology changes, while using hop-by-hop routing to eliminate the need for source routes. Experimentation with AODV led us to develop an improved variant called AODV-LL that also incorporates ideas from DSR's on-demand Route Maintenance.

Figure 2(a) shows the fraction of the offered packets that were successfully delivered by the routing protocols when subjected to an identical workload of constant bit-rate (CBR) sources during a 900-second simulation of 50 nodes moving at a maximum speed of 20m/s in a site 300m $\times$ 1500m. The x-axis describes the degree of mobility in the scenario, with Pause Time 0 meaning constant motion and Pause Time 900 meaning no node motion. Each point on the curves is the average of 10 runs on different scenarios, with each protocol evaluated on the same set of 210 different scenarios.



(a) Packet Delivery Ratio **(b)** Routing Protocol Overhead

Figure 2 Comparison of routing protocol metrics when the protocols are subjected identical workloads. Pause Time 0 is constant motion and Pause Time 900 is no motion.

Figure 2(b) shows a comparison of the number of routing protocol packets that were sent in order to achieve the packet delivery rates shown in Figure 2(a). The ideal routing protocol would require zero packets, meaning it introduced no overhead on the functioning of the network. Both DSR and AODV-LL have the desirable property that their overhead drops to near zero as node mobility decreases, though AODV-LL's overhead increases much more rapidly than DSR's as node motion increases.

We believe this simulation study establishes DSR as strong foundation on which to build an ad hoc network. On the set of scenarios we evaluated, DSR consistently has significantly less overhead and higher packet delivery rate than the other protocols we studied. Further, careful analysis of the trace files from these simulations established that the measured performances stemmed directly from fundamental properties of the protocols, and was not due to the specific scenarios we used, measurement fluke, or implementation error. Our results have recently been confirmed by Johansson *et al* [14].

4. Evaluation

Due to the infeasibility of constructing large physical test networks with which to evaluate a protocol for ad hoc networks, the majority of the evaluation of DSR will be carried out in a detailed packet-level simulator that Josh Broch and I have constructed. However, we have also constructed an implementation of the DSR protocol that enables real world experiments when the resources are available. This section briefly describes the simulator, the methodology I use for analyzing the DSR protocol, and the physical testbed.

4.1. Simulating and Analyzing DSR's Performance

Our simulator for ad hoc networks is based on the widely-used *ns-2* network simulator [7], but extends it with a framework into which arbitrary movement, propagation, network interface, antenna and media access models can be inserted [20]. Currently, the models we provide include:

- Two Ray Ground Reflection and Friss Free-space radio propagation models.
- A complete implementation of the IEEE 802.11 DCF MAC protocol [12].
- An implementation of the Lucent WaveLAN-1 CSMA/CA MAC protocol [1].
- Models of the Lucent WaveLAN-IEEE and Lucent WaveLAN-1 radio network interfaces.
- Implementations of various routing protocols for ad hoc networks.

The input to a simulation is a *scenario file* describing the movement pattern of the nodes during the simulation and the traffic those nodes should generate. The output of the simulation is a trace of all packet send and receive events, as well as any additional information logged by the protocols themselves. Figure 2 is an example of the results obtainable from the simulator. I have also created tools that enable the graphical construction of scenarios and the visualization of simulations.

As described in Section 3, DSR is a framework with two basic mechanisms: Route Discovery and Route Maintenance. The basic mechanisms, however, can be extended by numerous optimizations with the potential to dramatically improve or hinder the performance of the overall protocol. Inside the simulator, I will construct variants of the DSR protocol containing different mixes of optimizations. Using the ability the simulator gives me to run different variants against identical movement patterns and traffic work-loads, I will determine which variants perform better.

By analyzing the performance of the DSR variants, I will be able to dissect which of the optimizations that compose DSR are most important to its performance. By varying the scenarios I use to challenge the optimizations, I will be able to characterize the situations in which each optimization is most effective. This information will help the designers of future protocols concentrate on the most effective designs, as well as help engineers fine-tune their protocols to the particular environments in which the protocols will be deployed.

I will also study issues such as the latency penalty imposed on data packets by the routing protocol, the time required to repair broken routes, and the reasons for packet loss in the network.

4.2. Ad Hoc Network Testbed

As a proof that DSR is a physically realizable protocol, I will present performance data from our ad hoc network testbed [23]. In the testbed, 5 car-mounted mobile nodes drove a circular course at approximately 25 KPH while communicating with each other, 2 additional stationary nodes, and unmodified fixed nodes reached via an internet connected to the ad hoc network. The traffic load offered to the network included a mix of TCP connections, synthetic audio connections, and periodic measurement and status packets with hard real-time constraints.

5. Related Work

Ramanathan and Steenstrup [31] provide an excellent survey of routing protocols designed for use in ad hoc networks. Two other routing protocols with very different designs than DSR, TORA and DSDV, were briefly described in Section 3.

The first protocol that I know of with a structure similar to DSR was created as part of the DARPA PRNET project [18]. This protocol [19] used source routes for controlling the forwarding of packets across an ad hoc network, and could use a flooding mechanism like basic Route Discovery to find these source routes. It does not appear to include the notion of Route Caches, Route Maintenance, or any of the optimizations to Route Discovery described in this proposal. The original insight for DSR came from IEEE 802.3 Source Routing Bridges [30], as a form of multi-hop ARP.

DSR has already had a significant impact on the development of routing protocols for ad hoc networks. Many of the currently proposed protocols either incorporate DSR itself, are based on the DSR framework of Route Discovery and Route Maintenance, or include DSR's component mechanisms.

The Zone Routing Protocol (ZRP) [9, 8, 10] is a framework for routing in ad hoc networks that uses a periodic Intra-zone Routing Protocol for creating and routing inside clusters. It uses an on-demand Inter-zone Routing Protocol for routing between clusters, and suggests DSR as a candidate. Several of the optimizations proposed for improving the efficiency of the Inter-zone Routing Protocol appear similar to DSR optimizations to Route Discovery.

The Cluster Based Routing Protocol (CBRP) [13] uses a clustering protocol to elect and maintain cluster heads, but then uses DSR for routing packets between clusters. Signal Stability Adaptive (SSA) Routing [6] uses a scheme like DSR for routing, but chooses routes based on the length of time the routes' component links have been receiving packets with strong signal levels.

AODV [29, 28] is a routing protocol for ad hoc networks that combines mechanisms from both DSR and DSDV [27]. However, the two protocols are fundamentally different, as AODV is based on hop-by-hop routing, rather than on source routing like DSR. AODV constructs routes on-demand using a Route Discovery mechanism like DSR's, but its hop-by-hop nature requires it to use hard-state sequence numbers (adapted from DSDV) in order to prevent routing loops. Hop-by-hop routing in AODV eliminates the need for a source route in each data packet, which reduces the byte overhead of the protocol, but at the cost of dramatically increasing packet count overhead due to missed opportunities for optimizations. The path-state extension citemaltz:path-state-proposal,dsr-id-03 to DSR will reduce source route overhead while maintaining the properties of DSR, such as route shortening and support of asymmetric routes, that AODV does not contain and that rely on the existence of source routes in some form.

Researchers both inside and outside our group are now working to add additional features to the basic DSR framework. Castaneda and Das have developed a query-localization optimization that operates as part of Route Discovery to reduce the overhead of finding a new route to a destination when an route that had been in use breaks [4]. Working from a suggestion I made, Hu has developed a more sophisticated Route Cache data structure that enables DSR to operate as an on-demand, loosely consistent Link State protocol [11]. I have developed extensions to DSR that enable the explicit management of network resources, such as battery power and the use of bandwidth, as well providing better-than-best-effort Quality of Service [21, 2].

Many research groups are now using our *ns-2* extensions to support their own research on ad hoc networks, including groups at UPenn, UT Dallas, Texas A&M, UCLA, Nokia, Telcordia, Ericsson, SICS, and many others. Several papers have been published using our simulator as the evaluation platform [14, 4].

6. Summary and Contributions

In summary, I have designed and developed the Dynamic Source Routing protocol, which I have demonstrated to be a successful routing protocol in a variety of ad hoc networking environments. In my thesis, I will describe how DSR's excellent performance results from its use of source routes and its completely on-demand nature. I will dissect the protocol into its component mechanisms and optimizations in order to determine the effect they have on the overall performance, and I will characterize the situations in which each is most beneficial.

The thesis will make three main contributions. First is the DSR protocol itself, which has already demonstrated its usefulness in the real world as part of our ad hoc network testbed. The second contribution will be the characterization of the importance of the different optimizations that comprise DSR, as it will help guide future designers in tuning their protocols. Finally, the simulator, the testbed, and their associated tools are concrete contributions that others have already begun using.

7. Milestones

- Development of simulation tool: **Completed, but refinements ongoing.**
- Simulation of DSR: **Completed.**
- Implementation of the DSR protocol and field testing of ad hoc network. Joint work with J. Broch: **Completed.**
- Evaluation of DSR component optimizations: **Sufficient set completed.**
  - Core optimizations over random waypoint scenarios: **Completed.**
  - Core optimizations over scenarios with destination and movement locality: **Not begun. Interesting, but not critical.**
  - More scenarios are always possible, and several optimizations have not yet been evaluated.

References

[1] AT&T. *Data Manual: WaveLAN Air Interface*, June 1995. Doc. No. 407-0024785.

[2] Josh Broch, David B. Johnson, and David A. Maltz. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet-Draft, draft-ietf-manet-dsr-03.txt, October 1999. Work in progress.

[3] Josh Broch, David A. Maltz, David B. Johnson, Yih-chun Hu, and Jorjeta Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*, pages 85–97, Dallax, TX, October 1998.

[4] Robert Castaneda and Samir R. Das. Query localization techniques for on-demand routing protocols in ad hoc networks. In *The Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, pages 186–194, Seattle, WA, August 1999.

[5] M. Scott Corson and Anthony Ephremides. A Distributed Routing Algorithm for Mobile Wireless Networks. *Wireless Networks*, 1(1):61–81, February 1995.

[6] Rohit Dube, Cynthia D. Rais, Kuang-Yeh Wang, and Satish K. Tripathi. Signal Stability based Adaptive Routing (SSA) for Ad Hoc Mobile Networks. *IEEE Personal Communications*, pages 36–45, February 1997.

[7] Kevin Fall and Kannan Varadhan, editors. *ns* Notes and Documentation. The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, January 1999. Available from http://www-mash.cs.berkeley.edu/ns/.

[8] Zygmunt J. Haas and Marc R. Pearlman. A New Routing Protocol for the Reconfigurable Wireless Networks. In *Proceedings of the 6th IEEE International Conference on Universal Personal Communications*, pages 562–566, October 1997.

[9] Zygmunt J. Haas and Marc R. Pearlman. The Performance of Query Control Schemes for the Zone Routing Protocol. In *Proceedings of SIGCOMM'98*, pages 167–177, September 1998.

[10] Zygmunt J. Haas and Marc R. Pearlman. The Zone Routing Protocol (ZRP) for Ad Hoc Networks. Internet-Draft, draft-ietf-manet-zone-zrp-01.txt, August 1998. Work in progress.

[11] Yih-Chun Hu. A Link State Cache for DSR. Private communication, October 1999.

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

[13] Mingliang Jiang, Jinyang Li, and Y.C. Tay. Cluster Based Routing Protocol (CBRP). Internet-Draft, draft-ietf-manet-cbrp-spec-01.txt, August 1999. Work in progress.

[14] Per Johansson, Tony Larsson, Nicklas Hedman, Bartosz Mielczarek, and Mikael Degermark. Scenario-Based Performance Analysis of Routing Protocols for Mobile Ad-Hoc Networks. In *The Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, pages 195–206. ACM, August 1999.

[15] David B. Johnson. Routing in Ad Hoc Networks of Mobile Hosts. In *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, pages 158–163, December 1994.

[16] David B. Johnson and David A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. In *Mobile Computing*, edited by Tomasz Imielinski and Hank Korth, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.

[17] David B. Johnson and David A. Maltz. Protocols for Adaptive Wireless and Mobile Networking. *IEEE Personal Communications*, 3(1):34–42, February 1996.

[18] John Jubin and Janet D. Tornow. The DARPA Packet Radio Network Protocols. *Proceedings of the IEEE*, 75(1):21–32, January 1987.

[19] Robert E. Kahn, Steven A. Gronemeyer, Jerry Burchfiel, and Ronald Kunzelman. Advances in Packet Radio Technology. *Proceedings of the IEEE*, 66(11):1468–1496, November 1978.

[20] David A. Maltz. *The CMU Monarch Project's Wireless and Mobility Extensions to ns*. The CMU Monarch Project, monarch@monarch.cs.cmu.edu, snapshot release 1.1.1 edition, August 1999.

[21] David A. Maltz. Proposal: Resource Management in Multi-hop Ad Hoc Networks. Available from dmaltz@cs.cmu.edu, November 1999.

[22] David A. Maltz, Josh Broch, Jorjeta Jetcheva, and David B. Johnson. The Effects of On-Demand Behavior in Routing Protocols for Ad Hoc Networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1439–1453, August 1999.

[23] David A. Maltz, Josh Broch, and David B. Johnson. Experiences Designing and Building a Multi-Hop Wireless Ad Hoc Network Testbed. Technical Report 99-116, School of Computer Science, Carnegie Mellon University, March 1999.

[24] Vincent D. Park and M. Scott Corson. A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks. In *Proceedings of INFOCOM'97*, pages 1405–1413, April 1997.

[25] Vincent D. Park and M. Scott Corson. A Performance Comparison of TORA and Ideal Link State Routing. In *Proceedings of IEEE Symposium on Computers and Communication '98*, June 1998.

[26] Vincent D. Park and M. Scott Corson. Temporally-Ordered Routing Algorithm (TORA) Version 1: Functional Specification. Internet-Draft, draft-ietf-manet-tora-spec-01.txt, August 1998. Work in progress.

[27] Charles E. Perkins and Pravin Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In *Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, August 1994. A revised version of the paper is available from http://www.cs.umd.edu/projects/mcml/pub.html.

[28] Charles E. Perkins and Elizabeth M. Royer. Ad Hoc On Demand Distance Vector (AODV) Routing. Internet-Draft, draft-ietf-manet-aodv-02.txt, November 1998. Work in progress.

[29] Charles E. Perkins and Elizabeth M. Royer. Ad-hoc On-Demand Distance Vector Routing. In *Proceedings of WMCSA99: 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, February 1999.

[30] Radia Perlman. *Interconnections: Bridges and Routers*. Addison-Wesley, Reading, Massachusetts, 1992.

[31] S. Ramanathan and Martha Steenstrup. A Survey of Routing Techniques for Mobile Communications Networks. *ACM/Baltzer Mobile Networks and Applications*, 1(2):89–103, 1996.