

# AODV-DS with Dominant Pruning in Mobile Ad Hoc Networks

Sonali Mote<sup>1</sup>, Somnath Wategaonkar<sup>2</sup>, Surekha Khot<sup>3</sup>, and Sangita R. Wategaonkar<sup>4</sup>

<sup>1</sup>LBHSSST's ICA, Mumbai-400051, India

sonalimote@gmail.com

<sup>2</sup>BVCOE, Navi-Mumbai-410210, India

itssomnath@rediffmail.com

<sup>3</sup>A.C.P.C.O.E., Navi-Mumbai-410210, India

surekhaakhot@rediffmail.com

<sup>4</sup>S.P.I.T., Mumbai, India

sangitarw@gmail.com

**Abstract.** We investigate the use of dominating-set neighbor elimination as an integral part of the distribution of route requests using the Ad hoc On-demand Distance Vector (AODV) protocol as an example of on-demand routing protocols. We use detailed simulations to show that simply applying dominant pruning (DP) to the distribution of route requests in AODV results in pruning too many route requests in the presence of mobility and cross-traffic. Accordingly, we introduce several heuristics to compensate the effects of DP and show that the resulting AODV with Dominating Set heuristics (AODV-DS) has comparable or better delivery ratio, network load, and packet latency than the conventional AODV. AODV-DS exhibits over 70% savings on RREQ traffic than conventional AODV, and in some situations, AODV-DS may have a lower control overhead using Hello packets than conventional AODV without Hellos.

**Keywords:** MANET, AODV, AODV-DS.

## 1 Introduction

Routing in an ad hoc network has several challenges not present in wire-line networks. Bandwidth and energy are limited, so one must have message efficiency. Nodes typically use a single contention-based radio channel, so in multi-hop environments – where one would need routing – all-node broadcasts are error-prone due to hidden-terminal losses. Links are ephemeral and nodes have no means to detect the creation or loss of a link except through use of the link. Mobility further complicates matters because the network topology may be in a state of constant change and a node's picture of the network graph must be continually refreshed. A number of approaches to routing in ad hoc networks have been proposed in the recent past that address the aforementioned challenges by either having all nodes act as peers (i.e., execute the same protocols and algorithms), or by defining a backbone of nodes that carry out special routing functions.

To reduce the signaling overhead on a peer-to-peer basis, on-demand routing protocols maintain routes to only those destinations for which traffic exists. A couple of examples of such protocols are DSR (Johnson & Maltz 1996) and AODV (Perkins & Royer 1999) (Perkins et al. 2002a). For the purposes of this paper, the main feature of these protocols is that a node uses a series of network-wide all-node broadcasts to disseminate its route request to discover a route to an intended destination. In most situations, the node uses an expanding-ring search to limit flooding the whole network, but this comes at additional cost to the local area of a node where the same route request is likely to be repeated several times. Hence, it would be highly desirable to limit the number of unnecessary broadcast transmissions.

Another peer-oriented approach to reducing routing overhead is exemplified by the Optimized Link State Routing (OLSR) protocol (Clausen, Jacquet, Laouiti, Minet, Muhlethaler, Qayyum & Viennot 2001), which operates by flooding link-state information and limits the overhead incurred by flooding by only having the multipoint relays (MPR) of a node forward the flooded packets. A node selects MPRs from its symmetrical one-hop neighbors. A node designates a neighbor symmetrical after verifying a bi-directional channel to that neighbor. OLSR defaults to an MPR set of all symmetric neighbors and thus floods over well-connected nodes and uses all well-connected nodes in routing computations. Each node may reduce the MPR set based on a locally tunable parameter called MPR COVERAGE, which is the minimum number of covers desired per two-hop neighbor. The algorithm suggested in (Clausen et al. 2001) uses a greedy approach to minimize the number of repeater nodes while trying to achieve the desired MPR COVERAGE.

The Topology Broadcast Reverse Path Forwarding (TBRPF) (Ogier et al. 2001) routing mechanism uses broadcasts and limits flooding through a packet cache similar to AODV. It is based on (Perkins, Beling-Royer & Das 2001a), which has the potential to limit the default blind flooding but does not have any specific mechanisms to do so. A TBRPF node may choose to not participate in routing, in which case it only receives TBRPF topology packets but does not originate any. Thus, no other node will create a route through the passive member.

There have been a few proposals for establishing a virtual backbone over which routing takes place (e.g., (Das & Bharghavan 1997, Das, Sivakumar & Bhargavan 1997)). In (Das & Bharghavan 1997) a spine is used for all communications, while in (Das et al. 1997) the backbone is used as a secondary route in case shortest-path routes fail. These approaches assume a perfectly scheduled MAC layer. Subsequent work (Wu & Li 2001) provides more advanced algorithms and more sophisticated methods to handle node movement, shutdown and power-on. (Wu & Li 2001) also suggests a way to run Dynamic Source Routing (DSR) (Johnson & Maltz 1996) over a connected dominating set of the network. The dominating set of a network is a subset of nodes such that each node is either in the dominating set, or is adjacent to a node in the dominating set. Obtaining the minimum connected dominating set of a graph is known to be NP-hard (Amis, Prakash, Vuong & Huynh 2000) (Garey & Johnson 1979) even when the complete network topology is available.

Our work is distinct from (Wu & Li 2001) and OLSR's MPR scheme in that we only use dominating sets for flooding control, not packet routing, and we construct a more robust connected dominating set through several heuristics. In particular, we address the process of distributing route requests of an on-demand routing protocol in ways that

reduce the overhead incurred by the protocol without incurring a substantial negative impact on the ability of the network to deliver data packets to their destinations.

In the present work, we use Dominant Pruning (DP) (Lim & Kim 2001) as our dominating-set broadcast distribution mechanism and apply it to AODV, which we use as our example of on-demand routing protocols. Section 2 describes how DP is applied to the forwarding of route request (RREQ) packets in AODV. To facilitate DP, each node needs two-hop neighbor information. We use the neighbor-exchange protocol NXP (Mosko & Garica-Luna-Aceves 2002). Applying DP directly to AODV does not result in better performance from that obtained by the conventional AODV in many situations, due to the loss of RREQ packets. We present several heuristics to re-introduce some of redundancy. Although we have only used DP, the heuristics should apply to many dominating-set approaches. The resulting approach is called AODV-DS.

Section 3 presents the results of our simulation performance analysis between conventional AODV, AODV with DP, and AODV-DS. The results from our analysis are consistent with the findings for OLSR (Laouti, Muhlethaler, Najid & Plakoo 2002), which show that multipoint relays (MPR) significantly reduce the protocol overhead, but also results in a lower route availability and packet delivery rates, except for one topology. Our results show that AODV with DP has similar behavior: lower overhead and lower packet delivery ratio. The AODV-DS protocol using our heuristics has lower overhead and equal or higher delivery ratio.

## 2 Dominating Sets in AODV

This section reviews the RREQ process of AODV and the Dominant Pruning algorithm. It then describes our integration of a neighbor elimination scheme to AODV. We present several heuristics to boost the performance of dominant pruning by adding more redundancy.

In AODV, a node generates a RREQ to find a path to a specific destination, generally using an expanding ring search. The expanding ring search begins with a small TTL flood over the neighborhood of the source. If a RREQ times out, the source re-transmits the RREQ with a larger TTL until it finds a route to the destination or has exceeded a threshold and terminates the search in failure. A node receiving a RREQ with positive TTL will relay the RREQ if it cannot send a Route Reply (RREP) for the desired destination. RREP packets are sent unicast. Nodes keep a packet cache of recently seen RREQ packets, and drop duplicates.

Dominant Pruning is an algorithm to achieve a minimum connected dominating set (MCDS). A connected dominating set of graph  $G=(N,V)$  is a subset  $S \subseteq N$  such that every node in  $N-S$  has an edge to at least one node in  $S$  and that  $S$  is connected. A minimum CDS is a CDS with minimal set size. All nodes are expected to have information about the two-hop neighborhood. For a packet originated at node  $i$ , DP performs a greedy set cover (GSC) of all two-hop nodes  $N2[i]$  using the one-hop node set  $N1[i]$ . This cover set is appended to the data packet and broadcast to the neighborhood. When a node  $i$  receives a packet from node  $j$ ,  $i$  will relay the packet if it is listed in the forwarding set in the packet. When  $i$  relays the packet, it will use a last-hop specific forwarding set.  $i$  creates a last-hop effective one- and two-hop neighbor sets.

Let  $N1[i,j]$  be the  $N1$  set of  $j$  known at  $i$  via Hello messages. The effective one-hop set is  $E1[i,j]=N1[i]\setminus N1[i,j]$ . The effective two-hop set is  $E2[i,j]=N2[i]\setminus N1[i,j]\setminus N1[i]$ . Node  $i$  then performs a greedy set cover of  $E2[i,j]$  with  $E1[i,j]$  yielding the forwarding set for the relayed packet from  $j$ .

## 2.1 AODV-DS

It is straightforward to apply a neighbor elimination scheme to the process of flooding RREQs in an on-demand routing protocol. In the case of AODV, every node connected by the dominating set may receive the RREQ, and any node with an active route to the destination and appropriate sequence numbers may respond. Only nodes listed in the forwarder set RREQ extension may relay the RREQ. The main issues in making use of dominating sets worthwhile are how to make the dominating-set scheme more robust, and how to ensure fairness so the broadcast backbone is not unduly burdened with both broadcast and unicast traffic. Our main implementation difficulty with combining a neighbor elimination scheme with the AODV RREQ process arises from packet loss. We found that replacing the greedy set cover of DP with a *least-first* set cover (LFSC) and using hints from the AODV routing table yielded the best performance.

The AODV-DS algorithm is based on three heuristics to the DP scheme. We eliminate certain nodes from the eligible one-hop neighbors when performing the set cover of two-hop nodes, we use a LFSC rather than a GSC, and we add certain nodes to the forwarder set in addition to the LFSC results. When computing the DP cover set, we first compute the set *invalid*, being any broken 1-hop AODV route. We compute the DP cover set by first removing all *invalid* nodes from the one-hop set reported by the neighbor protocol and then compute the DP cover (which could be an empty set). We compute the cover using a LFSC, which is essentially the inverse of GSC: begin with the node whose cover size is minimal but non-zero. After we have the set cover from LFSC, we add in all nodes in the *invalid* set to the forwarder list. Finally, if we have any route information for the destination (either an active or broken route), we add the listed next-hop to the forwarder list.

To summarize, we construct the forwarder list on a hop-by-hop basis. We first remove from consideration any one-hop nodes listed by AODV as broken routes (but listed by the neighbor protocol as Up) and perform LFSC to get the forwarder list. We then add the excluded *invalid* nodes to the forwarder list. Finally, if we have any routing information for the destination from either an active route or broken route, we add the listed next-hop to the forwarder list.

## 3 Simulation Results

We implemented AODV draft 10 and Dominant Pruning in GloMoSim (Bajaj et al. 1999). We did not use the version of AODV distributed with GloMoSim, but rather made a new version conforming to recent AODV specification. This section first reviews our implementation of AODV, and then describes the simulation environment before finishing with the results of simulation. We used all default parameters from AODV draft 10, with the following differences. We set  $TTL\ START = 2$ , we use local repair, we do not use link-layer drop detection, and we do not use reboot hold.

ALLOWED HELLO LOSS only applies to conventional AODV as AODV with DP and AODV-DS use an external two-hop neighbor protocol. Following the recommendation in draft 10, we unicast RERR packets whenever there is only one destination in the precursor list.

Our version of AODV has the following implementation-specific features. If a RREQ for a destination has failed within DELETE PERIOD and a new packet arrives for that destination, it starts with a TTL of NETWORK DIAMETER. After a RREQ failure for a destination, a node will not issue another RREQ for that node for 6 seconds, and it will have a TTL of NETWORK DIAMETER. After each failure (the conventional 2 retries are allowed), queued packets are dropped. We have not investigated the Hello reduction technique that is specified in the more recent AODV specification (Perkins, Belding-Royer & Das 2002*b*), where only nodes participating in active routes need to issue Hello packets to maintain connectivity. All AODV control packets – broadcast and unicast – are jittered by an exponential delay with mean value of 10 milliseconds, with a minimum of 1 ms and a maximum of 100 ms.

Our simulations generally replicate (Perkins, Royer, Das & Marina 2001*b*) for a 50-node network. We have scenarios with 10 source nodes and 30 source nodes, transmitting 4 pack-ets/sec CBR traffic of 512 byte UDP packets. Nodes begin transmitting at 50 seconds plus an offset uniformly chosen over a 5 second period to avoid all nodes sending a packet at exactly 50s. Destination nodes are chosen uniformly from any node except the source. All simulations run for 900 seconds.

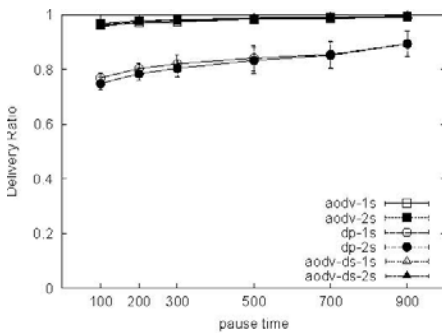
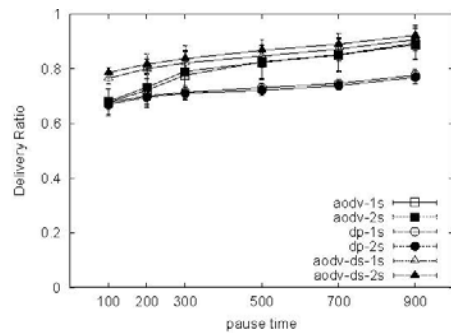
We use a random waypoint movement model with velocities between 0 and 20 m/s in a 1500m x 300m space with random initial node placement. We use six pause times of 100s, 200s, 300s, 500s, 700s, and 900s. The radio is a 2 Mbps IEEE 802.11 device with a maximum range of 280m. The radio uses an accumulated noise interference model and a two-ray path loss. We used two Hello periods of 1 second and 2 seconds. We repeated all experiments over 10 trials with different random number seeds. Each data point represents the mean over the 10 trials. We show 95% confidence intervals all graphs except some cumulative distribution plots.

Our performance metrics are similar to (Perkins et al. 2001*b*). We measure the delivery ratio of CBR packets received to packets transmitted, the latency of received CBR data packets, and the control overhead. The control overhead is the ratio of the total number of AODV control packets (RREQ, RREP, RERR, Hellos) to the number of data CBR packets received. In cases where we used NXP, all NXP packets are counted in the control overhead.

Table 1 presents the four performance metrics averaged over all pause times and the 95% confidence interval. Due to space, we only show graphs for the delivery ratio, RREQ load, and RREQ distribution. Any entries in a column with overlapping confidence intervals are statistically identical. AODV-DS has a statistically identical delivery ratio to AODV in all cases. AODV-DS has a significantly better delivery ratio than AODV with DP in all cases. For 10source network load, AODV-DS is statistically identical to AODV, while for 30-sources, AODVDS has about 1/3 the load of AODV. In terms of the number of RREQ packets transmitted, AODV-DS averages under 1/3 the number of AODV, but is at times an order of magnitude higher than AODV with DP. For 10-sources, AODV-DS has about double the latency of AODV, but for 10-sources, it has about 1/2 the latency of AODV.

**Table 1.** Performance average over all pause times

sources	nodes	hello	protocol	delivery ratio	net load	rreq load	latency (sec)
10	50	1S	AODV	0.977 $\pm$ 0.009	1.943 $\pm$ 0.309	558.541 $\pm$ 253.097	0.029 $\pm$ 0.006
10	50	1S	AODV w/ DP	0.830 $\pm$ 0.032	1.648 $\pm$ 0.303	34.385 $\pm$ 5.257	1.281 $\pm$ 0.300
10	50	1S	AODV-DS	0.981 $\pm$ 0.009	1.530 $\pm$ 0.316	106.130 $\pm$ 46.467	0.054 $\pm$ 0.012
10	50	2S	AODV	0.982 $\pm$ 0.007	1.089 $\pm$ 0.190	367.967 $\pm$ 156.603	0.029 $\pm$ 0.005
10	50	2S	AODV w/ DP	0.818 $\pm$ 0.038	0.956 $\pm$ 0.187	32.327 $\pm$ 4.030	1.257 $\pm$ 0.281
10	50	2S	AODV-DS	0.977 $\pm$ 0.010	0.922 $\pm$ 0.216	102.843 $\pm$ 43.386	0.057 $\pm$ 0.012
30	50	1S	AODV	0.790 $\pm$ 0.060	3.545 $\pm$ 1.373	3924.323 $\pm$ 1365.625	0.687 $\pm$ 0.072
30	50	1S	AODV w/ DP	0.723 $\pm$ 0.026	0.822 $\pm$ 0.135	150.888 $\pm$ 5.169	1.595 $\pm$ 0.174
30	50	1S	AODV-DS	0.833 $\pm$ 0.037	1.579 $\pm$ 0.460	1158.063 $\pm$ 370.558	0.399 $\pm$ 0.007
30	50	2S	AODV	0.793 $\pm$ 0.056	3.328 $\pm$ 1.270	3797.293 $\pm$ 1219.152	0.739 $\pm$ 0.062
30	50	2S	AODV w/ DP	0.716 $\pm$ 0.025	0.512 $\pm$ 0.079	141.950 $\pm$ 3.698	1.661 $\pm$ 0.133
30	50	2S	AODV-DS	0.851 $\pm$ 0.036	1.147 $\pm$ 0.353	1023.992 $\pm$ 321.734	0.328 $\pm$ 0.007

**Fig. 1.** Delivery ratio, 10 sources**Fig. 2.** Delivery ratio, 30 sources

Figures 1 and 2 shows the delivery ratio for 10 and 30 source nodes. For 10 source nodes, AODV and AODV-DS have approximately equal delivery rates. AODV with DP has a significantly lower delivery ratio, due to multiple failures of RREQs. Overall, conventional AODV averaged under 0.5 failed route requests per node (for both 1s and 2s intervals), AODVDS averaged under 1.0 failed route requests per node. AODV with DP averaged 3.3 failed route requests per node (3.29 for 1s and 3.35 for 2s hello intervals), but had the fewest number sources of transmitted RREQs. For 30 source nodes, the differences are not as pronounced as for 10 source nodes. AODV-DS has the highest delivery ratio, AODV is next, and AODV with DP has the lowest delivery ratio.

Figures 3 and 4 shows the average number of RREQ packets transmitted per node over the simulation period. For both 10 sources and 30 sources, AODV with DP transmitted significantly fewer RREQ packets than AODV or AODV-DS. On average over the ten trials, AODV with DP transmitted 34 RREQs for 1s Hellos and 32 RREQs for 2s Hello. AODV-DS transmitted between on average 104 RREQs per node, but had a very wide range between 24 and 205, depending on pause time. Conventional AODV averaged 428 RREQs per node, with a range of 109 to 965, depending on pause time. AODV-DS exhibits over a 70% savings in RREQs compared to conventional AODV and has a similar or better delivery ratio.

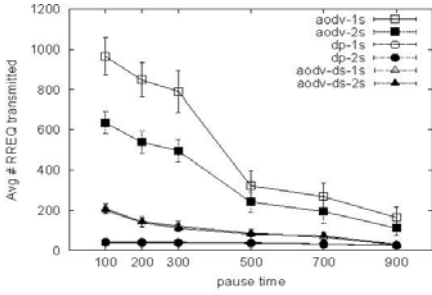


Fig. 3. Average #RREQ transmitted, 10 sources

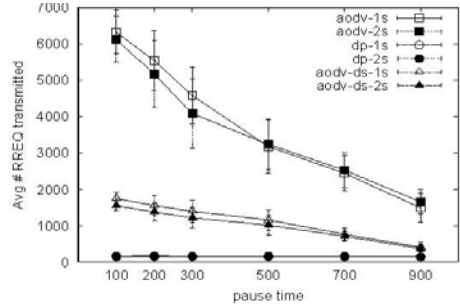


Fig. 4. Average #RREQ transmitted, 30 sources

Figures 5 and 6 shows the cumulative distribution (CDF) of RREQ transmissions for a 100s and 900s pause times. The CDF measures the fraction of RREQs transmitted by nodes. These plots illustrate a sense of fairness in the routing protocol – if the CDF is linear, then all nodes bear an equal share of load. The 30-source 100-second pause time graph is closest to linear for all scenarios. The 10-source 900-second pause time graph is the least linear. Intuitively, when there are more flows and more movement, the flows spread over the graph more evenly. When there is no movement and few flows, paths become established early and do not change. In both graphs, AODV is closest to linear because it completely floods the network. AODV with DP is the furthest from linear while AODV-DS is close to AODV.

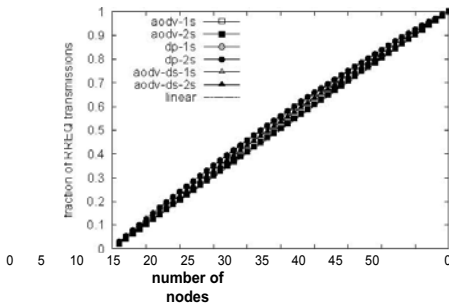


Fig. 5. RREQ distribution, 30 sources, 100s pause

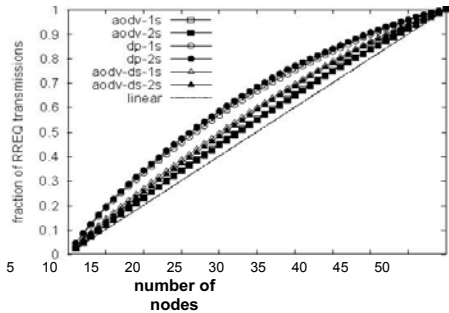


Fig. 6. RREQ distribution, 30 sources, 900s pause

When we compare our results with those in (Perkins et al. 2001b), there are three significant differences between AODV in (Perkins et al. 2001b) and our implementation of AODV. (Perkins et al. 2001b) uses link-layer feedback for failed links. When the MAC layer fails an RTS/CTS/ACK handshake and has used all allowable retries, it notifies AODV that a specific packet failed. AODV immediately breaks the link. (Perkins et al. 2001b) does not use Hello packets, which we rely on to detect link failures and exchange two-hop data. (Perkins et al. 2001b) broadcasts all RERR packets while our implementation sometimes unicasts them.

In terms of delivery ratio, our simulations of AODV show similar performance to (Perkins et al. 2001*b*), which observes a 97% or higher delivery ratio for 10 source nodes and between a 76% to 85% ratio for 30 source nodes. Our results show for 10 source nodes a similar 96% or higher delivery ratio. For 30 source nodes, our work shows between 68% and 89% delivery for conventional AODV and between 75% and 92% delivery for AODV-DS.

For the control load (“routing load” in (Perkins et al. 2001*b*)), one would expect a large difference because we used Hello packets. For 10 source nodes, (Perkins et al. 2001*b*) reports a routing load of 1.0 or less, dropping towards zero as the pause time increases. For 30 source nodes, the routing load is between 1.5 and 2.4. If we look at our best results, which is for a 2s Hello interval, for 10 source nodes conventional AODV showed 0.8 to 1.5 control load and AODV-DS 0.5 to 1.3. For 30 source nodes, AODV ranged between 1.3 and 5.9 while AODV-DS ranged between 0.4 and 1.7. Interestingly, for 30 source nodes we achieved a lower control load using Hello packets and AODV-DS than (Perkins et al. 2001*b*) reported for conventional AODV without Hello packets.

## 4 Conclusion

We present a method to combine dominating-set broadcast distribution with the AODV RREQ process. The novelty of our contribution is in addressing the fragility of a minimum connected dominating set in the presence of mobility and cross-traffic. We develop three heuristics to fortify the dominating set process against loss by re-introducing some redundancy using a least-first set cover rather than a greedy set cover. We also use hints from the AODV routing table to compute the forwarding list. AODV-DS exhibits about a 70% savings in RREQ traffic while maintaining the same or better latency and delivery ratio for 30 source nodes in a graph of 50 nodes. AODV-DS is also about as fair as conventional AODV in distributing the RREQ burden among all nodes, except in cases of low-mobility and few source nodes. For low-mobility networks, AODV-DS is not as fair to forwarding nodes as AODV, but is better than AODV with DP.

Future work includes detailed study of over-covering two-hop neighbors for more efficient mechanisms than a LFSC. We need a more thorough understanding of the interactions between a two-hop neighbor protocol and the AODV route expiry mechanism. From the RREQ load results, it appears at times that the LFSC adds very many nodes to the cover set. Investigating how to create more tightly bound cover sets should further reduce the network load.

## References

- [1] Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. In: Mobile Computing, vol. 353. Kluwer Academic Publishers, Dordrecht (1996)
- [2] Johnson, D.B., Maltz, D.A., Hu, Y.-C.: The dynamic source routing protocol for mobile ad hoc networks (DSR), IETF Internet draft, draft-ietf-manet-dsr-09.txt (2003)



- [3] Johnson, D.B., Maltz, D.A., Hu, Y.-C., Jetcheva, J.G.: The dynamic source routing protocol for mobile ad hoc networks (DSR), IETF Internet draft, draft-ietf-manet-dsr-07.txt (2002)
- [4] Perkins, C.E., Royer, E.M.: Ad hoc on-demand distance vector routing. In: Proc. WMCSA 1999 (1999)
- [5] Clausen, T., Jacquet, P.: RFC 3626: Optimized link state routing protocol OLSR (2003)
- [6] Clausen, T., Jacquet, P., Laouiti, A., Minet, P., Muhlethaler, P., Qayyum, A., Viennot, L.
- [7] Amis, A., Prakash, R., Vuong, T., Huynh, D.: MaxMin D-cluster formation in wireless ad hoc networks. In: IEEE INFOCOM (2000)
- [8] Das, B., Bharghavan, V.: Routing in ad hoc networks using minimum connected dominating sets. In: Proc. ICC 1997, vol. 1 (1997)
- [9] Das, B., Sivakumar, R., Bhargavan, V.: Routing in ad hoc networks using a spine. In: Proc. ICCCN (1997)
- [10] Laouti, A., Muhlethaler, P., Najid, A., Plakoo, E.: Simulation results for the OLSR routing protocol for wireless network, Technical Report 4414, INRIA (2002)
- [11] Ogier, R.G., Templin, F.L., Bellur, B., Lewis, M.G.: Topology broadcast based on reverse-path forwarding (TBRPF), IETF Internet draft, draft-ietf-manet-tbrpf-05.txt (2001)
- [12] Bajaj, L., Takai, M., Ahuja, R., Tang, K., Bagrodia, R., Gerla, M.: GloMoSim: A scalable network simulation environment, Technical Report 990027, UCLA Computer Science Department (1999)
- [13] Perkins, C.E., Belding-Royer, E.M., Das, S.: Ad hoc on demand distance vector (AODV) routing, IETF Internet draft, draft-ietf-manet-aodv-11.txt (2002b)
- [14] Perkins, C.E., Royer, E.M., Das, S.R., Marina, M.K.: Performance comparison of two on-demand routing protocols for ad hoc networks. IEEE Personal Communications 8(1) (2001b)