Journal of Computer Science 3 (4): 218-222, 2007
ISSN 1549-3636

Utilization of AODV in Wireless Ad Hoc Networks

[1]Zeyad M. Alfawaer, [1]GuiWei Hua, and [2]Nidal Abu Hamdeh
[1]School of Information Sciences and Engineering, Central South University,
Changsha 410083,Hunan, China
[2]School of Computer Science and Engineering, Bei hang University, Beijing 100083, China

Abstract: AODV is a mature and widely accepted routing protocol for Mobile Ad hoc Networks (MANET), it has low processing and memory overhead and low network utilization, and works well even in high mobility situation. We modified AODV to use these dominating sets, resulting in the AODV-DS protocol. Our contribution 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. 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. It was 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, it was not as fair to forwarding nodes as AODV, but better than AODV with Dominant Pruning (DP).

Keywords: ad hoc wireless network, RREQ process, AODV, on-demand routing.

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 [1] and AODV [2-3].F or 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.

The Topology Broadcast Reverse Path Forwarding (TBRPF) [4] routing mechanism uses broadcasts and limits flooding through a packet cache similar to AODV. It is based on [5], 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., [6-7]). In [8] a spine is used for all communications, while in [9] the backbone is used as a secondary route in case shortest-path routes fail. These approaches assume a perfectly scheduled MAC layer. Subsequent

Corresponding Author: Zeyad M. Alfawaer, School of Information Sciences and Engineering, Central South University, Changsha 410083,Hunan, China

work [10] provides more advanced algorithms and more sophisticated methods to handle node movement, shutdown and power-on. In [10] also suggests a way to run Dynamic Source Routing (DSR) [1] 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 [11-12] even when the complete network topology is available.

In this paper, we use Dominant Pruning (DP) [10] as our dominating set broadcast distribution mechanism and apply it to AODV, which we use as our example of on-demand routing protocols. 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.

Dominating Sets in AODV: 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 $S \leq N$ 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 $N1[i]$. his 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 s is $E1[i,j] = N1[i] \backslash N1[i,j]$. The effective two-hop set is $E2[i,j] = N2[i] \backslash N1[i,j] \backslash N1[i]$. 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 .

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.

RESULTS AND DISCUSSION

Our simulations generally replicate for a 50-node network. We have scenarios with 10 source nodes and 30 source nodes, transmitting 4 packets 1 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.

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: Performance average over all pause times

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

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 10- source network load, AODV-DS is statistically identical to AODV, while for 30-sources, AODVDS has about 113 the load of AODV. In terms of the number of RREQ packets transmitted, AODV-DS averages under 113 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 112 the latency of AODV.

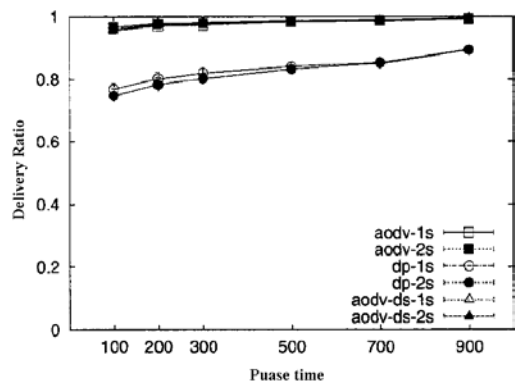

Fig.1 Delivery ratio, 10 sources

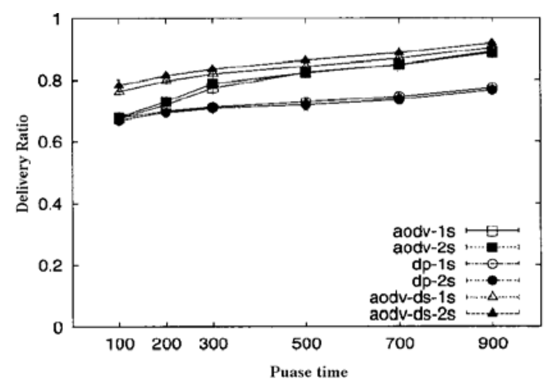

Fig.2 Delivery ratio, 30 sources

Fig.1 and 2 show 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 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.

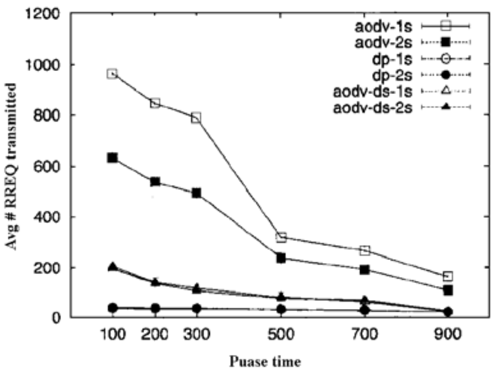

Fig.3. Average # RREQs transmitted, 10 sources

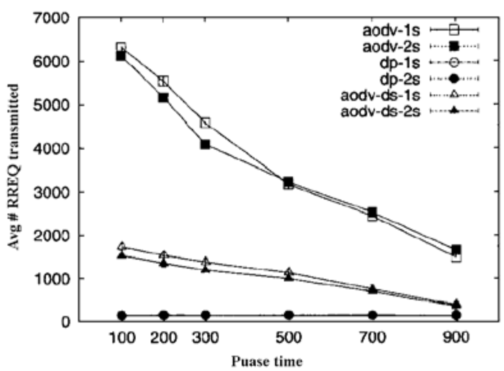

Fig.4. Average # RREQs transmitted, 30 sources

Fig.3 and 4 show 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.

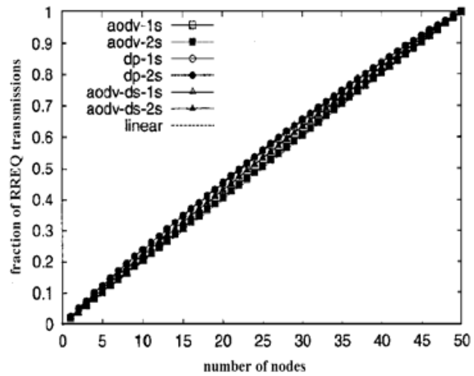

Fig.5 RREQ distribution, 30 sources, 100s pause

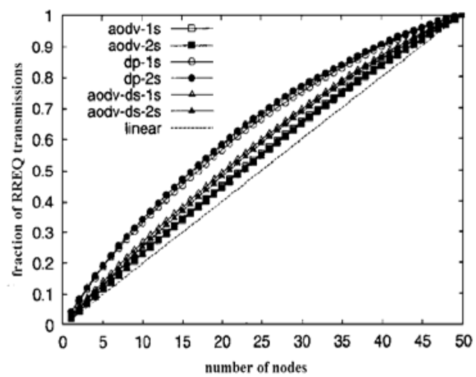

Fig.6 RREQ distribution, 10 sources, 900s pause

Fig.5 and 6 show 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 move 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.

CONCLUSIONS

In this paper, we present a method to combine dominating-set broadcast distribution with the AODV RREQ process, with the contribution to addressing the fragility of a minimum connected dominating set in the presence of mobility and cross-traffic. During the above study, three heuristics have been developed 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. And from the simulation results, it is clear that 1) 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, and 2) it 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, 3) although not as fair to forwarding nodes as AODV for low-mobility networks, AODV-DS is better than AODV with DP.

REFERENCES

1. Johnson, D. B. & Maltz, D. A. 1996, Dynamic source routing in ad hoc wireless networks, in 'Mobile Computing', Vol. 353, kluwer Academic Publishers.
2. Perkins, C. E. & Royer, E. M. 1999, 'Ad hoc on-demand distance vector routing', *Proc. WMCSA '99* pp. 90-100.
3. Perkins, C. E., Belding-Royer, E. M. & Das, S. 2002a, Ad hoc on demand distance vector (AODV) routing, IETF Internet draft, draft-ietf-manet-aodv- 10.txt.
4. Ogier, R. G., Templin, F. L., Bellur, B. & Lewis, M. G. 2001, Topology broadcast based on reverse-path forwarding (TBRPF), IETF Internet draft, draft-ietf-manet-tbrpf-05.txt.
5. Perkins, C. E., Beling-Royer, E. M. & Das, S. R. 2001a, IP flooding in ad hoc mobile networks, IETF Internet draft, draft-ietf-manet-bcast-00.txt.
6. Das, B. & Bharghavan, V. 1997, Routing in ad hoc networks using minimum connected dominating sets, *in* 'Proc. ICC', Vol. 1, pp. 376-380.
7. Das, B., Sivakumar, R. & Bhargavan, V. 1997, Routing in ad hoc networks using a spine, *in* 'Proc. ICCCN' .
8. Das, B., Sivakumar, R. & Bhargavan, V. 1997, Routing in ad hoc networks using a spine, *in* 'Proc. ICCCN' .
9. Wu, J. & LC H. 2001, 'A dominating-set-based routing scheme in ad hoc wireless networks', Telecommunicati*on Systems* 18(1-3), 13-36.
10. Amis, A., Prakash, R., Vuong, T. & Huynh, D. 2000, MaxMin D-cluster formation in wireless ad hoc networks, *in* 'IEEE INFOCOM', pp. 32-41.
11. Garey, M. & Johnson, D. 1979, *Computers and intractability. A guide to the theory of NPcompleteness,* Freeman, Oxford, UK.
12. Lim, H. & Kim, C. 2001, 'Flooding in wireless ad hoc networks', *Computer Communications* 24(3-4), 353-363.