ADPUMA: An Adaptive Multicast Routing Protocol for Mobile Ad Hoc Networks[1]

Rolando Menchaca-Mendez[2§], J. J. Garcia-Luna-Aceves[§*], Ravindra Vaishampayan[§]

[§] Department of Computer Engineering,
University of California, Santa Cruz
1156 High Street, Santa Cruz, CA 95064, U.S.A.
{menchaca, jj, ravindra}@soe.ucsc.edu
[*] Palo Alto Research Center (PARC), 3333 Coyote Hill Road
Palo Alto, CA 94304, U.S.A.

Abstract. In this paper we present ADPUMA, a mesh-based adaptive multicast routing protocol specifically designed to reduce the overhead needed to deliver multicast packets, saving bandwidth and energy, two of the scarcest resources in MANETS while providing high reliability. The three main features of ADPUMA are: (1) for each multicast group, it periodically floods a single control packet to build the mesh, elect the core of the mesh and get two-hop neighborhood information; (2), it computes the mesh's k-dominating set to further reduce overhead induced by flooding the mesh when forwarding data packets; and (3) each node dynamically selects one of the three different operation modes (PUMA, DPUMA-1k or DPUMA-2k) to adapt to the perceived mobility and contention conditions of the MANET.

Introduction

In this paper we present ADPUMA, an adaptive multicast protocol where nodes sense the current conditions of their neighborhood and based on that information select a mode of operation. Each operation mode (PUMA, DPUMA-1k or DPUMA-2k) is characterized by the amount of redundancy used to disseminate a multicast packet. As we will show, ADPUMA further improves the reliability and efficiency of its direct predecessor: PUMA [1]. The parameters used to define the state of the neighborhood are: the levels of relative mobility and the contention.

Detection of Relative Mobility and Contention Levels

In general, the performance of protocols which rely on the freshness of topological information is strongly impacted by the relative mobility among nodes. Hence, getting an accurate view of the instantaneous level of relative mobility can be very useful to select the particular strategy that performs best under each condition. In this paper we present a simple, yet accurate mechanism to detect relative mobility. To compute the level of relative mobility, each node keeps track of how its one-hop neighborhood has changed between two consecutive sampling periods, then, nodes compute an exponential average to avoid reacting so fast to changes in their perceived relative mobility.

We define instantaneous relative mobility *m* as follows:

$$m = \frac{d/(r+d)}{s_p}$$, where

- d is the number of new or missing one-hop neighbors detected in the current sampling period with respect to the neighbors detected in the previous sampling period
- r is the number of neighbors that didn't change from the previous sampling period to the current sampling period
- s_p is the sampling period

And the *degree of relative mobility* υ at sampling period n is: $v_n = \alpha v_{n-1} + (1-\alpha)m$.

The current default value for α is 0.2. However, our results show that the performance of ADPUMA is not very sensitive to this parameter. The other aspect that has a strong influence over the performance of protocols that use contention-based MACs is the traffic load. In the particular case of ADPUMA nodes detect their current level of local contention in order to decide the level of redundancy with which a packet has to be transmitted. To measure 1-hop contention we propose another simple and very intuitive metric that is based on the ratio between the number of received signals with errors and the total number of received signals during a fixed period of time. This ratio tries to approximate the probability of a successful transmission in a given period of time. Then, as in the previous case, we use an

[1] This work was supported in part by the National Science Foundation under Grant CNS-0435522, by the UCOP CLC under grant SC-05-33 and by the Baskin Chair of Computer Engineering at University of California, Santa Cruz

[2] This author is supported by the Mexican National Council for Science and Technology (CONACyT) and by the Mexican National Polytechnic Institute (IPN)

exponential average to cope with sudden and short term variations on this metric. In this way, we define the instantaneous contention level *c* as:

$$c = \frac{e/t}{s_p}$$, where

- e is the number of signals with errors received during the sampling period s_p
- t is the total number of signals received during the sampling period s_p
- s_p is the sampling period

And the *degree of contention* γ at sampling period *n* is as: $\gamma_n = \beta\gamma_{n-1} + (1-\beta)c$.

The current default value for β is 0.2, and as in the previous case our results show that the protocol is not very sensitive to this parameter.

ADPUMA's Adaptability

Based on the current values of the degree of relative mobility and the degree of contention, each ADPUMA node selects its current operation mode. The different combinations are shown on Table 1.

Table 1. Modes of operation of ADPUMA

	Low Contention	High Contention
Low mobility	PUMA	ADPUMA-1k
Medium mobility	ADPUMA-2k	ADPUMA-1k
High mobility	PUMA	ADPUMA-2k

Nodes use three treshold values to decide when to change form one mode of operation to the other. In this stage of our research this vales were manully set based on the preliminary results of our simulations.

Experimental Results

We compared the performance of DPUMA[3] and ADPUMA against the performance of PUMA, ODMRP[2] and MAODV[3] which are representatives of the state of the art in multicast routing protocols for MANETs. We used the discrete event simulator Qualnet version 3.5. We employed RTS/CTS when packets were directed to specific neighbors. All other transmissions used CSMA/CA. Each simulation was run for five different seed values (except for the mobility simulations were we used 10 different seed values). To have meaningful comparisons, all timer values (i.e., interval for sending JOIN requests and JOIN tables in ODMRP and the interval for sending multicast announcements in DPUMA) were set to 3 seconds. Table 2 lists the details about the simulation environment and the parameters of the MAODV code.

The metrics used for the evaluations are the average number of data packets delivered at receivers and the average number of data packets relayed.

Table 2. Simulation environment and parameters used in the MAODV code

Simulation Environment		MAODV parameters	
Total Nodes	50	Allowed Hello Loss	2
Simulation time	100s	Grp. Hello Interval	5 s
Simulation area	1300×1300m	Hello Interval	1 s
Node Placement	Random	Hello life	3 s
Mobility Model	Rdm. Waypoint	Pkt. Id save	3 s
Pause Time	10s	Prune Timeout	750 ms
Min-Max Vel.	0 – 10 m/s	Rev Route life	3 s
Tx Power	15 dbm	RREQ retries	2
Channel Capacity	2000000 bps	Route Disc. Timeout	1 s
MAC protocol	IEEE 802.11	Retransmit timer	750 ms
Data Source	MCBR		
Pkts. sent per src.	1000		

Packet Size

In this experiment we vary the packet size from 64 to 1024 bytes. This experiment shows the ability of 1k-DPUMA to reduce the channel contention by reducing the number of nodes that broadcast a packet in a given time. There is one sender and one group composed of the 50 nodes. As it is shown in Figure 3, for packets of 512 bytes and larger DPUMA-1k achieves higher delivery ratios than the other protocols (except ODMRP) while incurring less retransmission overhead. ODMRP has similar overhead, but under high contention conditions its reliability is higher to the one in DPUMA-1k. This is due ODMRP is particularly efficient when there is only one sender.

Among the family of PUMA-based protocols, from figure 3 we can also observe that under low contention conditions (small packets – lower probability of collisions), the basic version of PUMA outperforms the versions that use dominating-set techniques. On the other hand, under high contention conditions, DPUMA-1k outperforms the other versions of PUMA. This behavior was one of reasons why we develop ADPUMA, which is capable to adapt to different conditions, providing similar delivery ratio to

[3] DPUMA-nk use a *n*-dominating set algorithm to select a set of 1-hop neighbors that will forward a packet when disseminating a data message over a mesh.

PUMA under low contention and to 1k-DPUMA under high contention. An interesting situation is that for packets of 512 bytes, ADPUMA performs even better than the other protocols. The reason is that nodes can select the "best strategy" independently of the other nodes, so nodes which are in different regions of the MANET can use different operation modes, if the conditions are different from region to region.

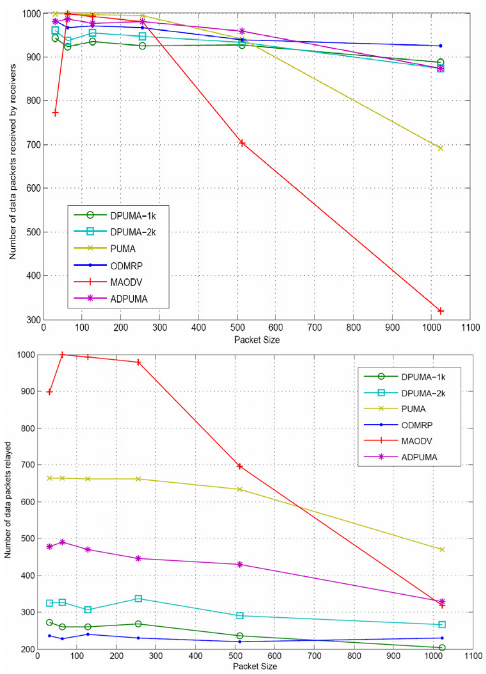

Fig. 1. Average number of data packets received at receivers and average number of data packets relayed when varying the packet size.

Node Mobility

As we previously discussed, the performance of protocols which rely on the freshness of topological information is strongly impacted by the relative mobility among nodes. In this experiment we show how effective is the proposed metric to detect relative mobility, and how this information is used by the nodes to autonomously decide which mode of operation has to be used. Figure 4 shows the results of varying the node's speed from 1m/s to 50 m/s with a pause time of 0 seconds, one group composed of 30 members, and one sender generating 20 packets of 512 bytes per second. As it can be seen on the figure, ADPUMA performs similar to the best of the pure options of the family of PUMA protocols (PUMA, DPUMA-1k, DPUMA-2k). This is a strong indication that ADPUMA nodes effectively detect the current mobility condition and select the appropriate mode of operation.

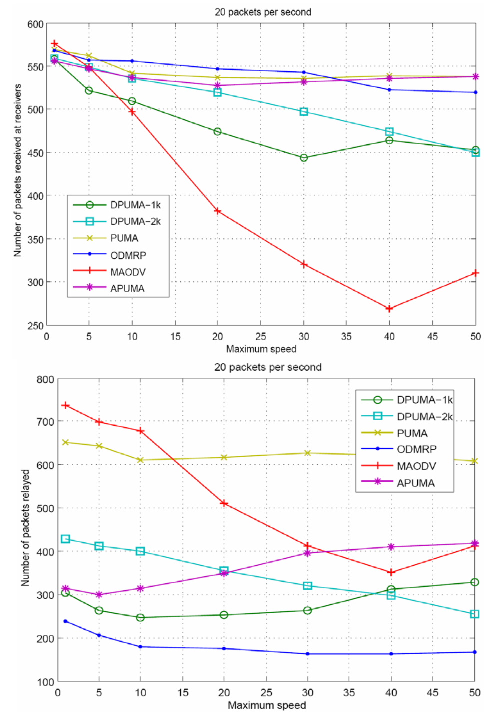

Fig. 4. Average number of data packets received at receivers and average number of data packets relayed when varying the node's speed.

Conclusions

We defined two metrics that measure local congestion and relative mobility perceived by each node. Then, based on these metrics we defined and implemented an adaptive approach that gets the reliability achieved by PUMA under light loads and high mobility; and the one achieved by DPUMA-1k under high loads and low mobility.

References

1. R. Vaishampayan and J.J. Garcia-Luna-Aceves, Efficient and Robust Multicast Routing in Mobile Ad Hoc Networks , Proc. IEEE MASS 2004: The 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems, Fort Lauderdale, Florida, October 25-27, 2004
2. S.J. Lee, M. Gerla, and Chian, "On-demand multicast routing protocol," in Proceedings of WCNC, September 1999
3. E. Royer and C. Perkins, "Multicast operation of the ad hoc on-demand distance vector routing protocol," in Proceedings of Mobicom, August 1999