

PERFORMANCE EVALUATION OF MOBILE AD HOC NETWORKS USING ROUTING WITH SELFISH NODES

Jatinder Singh¹ and Rajiv K. Ranjan²

¹Electronics and Communication Engineering Department,

²Sant Longowal Institute of Engineering & Technology, Longowal, Sangrur, Punjab, India

E-mail : ¹garai.jatinder@yahoo.co.in

Abstract

In mobile ad hoc networks (MANETs)[1], a mobile station (MS) can communicate with nodes outside of its transmission range by employing intermediate nodes as relays. To find a path to a specific destination node, the source node broadcasts route request packets that are forwarded to all other nodes. A simple flooding scheme is employed in AODV [2], where every node rebroadcasts these route request packets even if some of its neighbors have already broadcast the requests and thus the rebroadcasts may reach no additional nodes. Appropriate use of a probabilistic method can reduce the number of rebroadcasting; therefore reduce the chance of contention and collision among neighboring nodes. A good probabilistic broadcast protocol can achieve higher throughput and lower energy consumption, without sacrificing reach ability or having any significant degradation. In this paper, we propose a selfish node based approach.

Key words: ad hoc Networks, Manets, Protocol

I. INTRODUCTION

We observed that all existing probabilistic protocols introduce uncertainty in the decision making of MSs about whether or not to do rebroadcast. As a result, the network reach ability cannot be fully ascertained. Based on this observation, in this paper we propose an improvement over the totally blind probabilistic broadcasting approach by using a scheme based on selfish nodes. We categorize MSs into normal nodes and selfish nodes. Normal nodes relay packets for other while selfish nodes do not. This is a hierarchical approach where the relaying node set is the set of normal nodes. Compared to other hierarchical approaches, our approach incurs very little additional computations and communications. Specifically, our approach is based on a combination of gossiping [3] and a dominating set. The main difference between our approach and the dominating set approach is that we do not need to calculate a minimal connected dominating set. Instead, we classify nodes into two groups: normal nodes or selfish nodes. Since we do not need to calculate a minimal connected dominating set, we can reduce the initial determination and update cost in terms of computation and communication.

On the other hand, we use a gossiping-like probabilistic approach to assign selfish nodes. As a result, our approach exhibits simplicity of gossiping approach.

In this paper, we describe our selfish-node-based approach and evaluate its performance through simulation. We compare our approach with AODV in terms of several metrics, including average number of

rebroadcasts, average number of packet collisions, end-to-end delay, and throughput. Simulation results show that our approach outperforms the AODV protocol in dense networks and is comparable in other situations. In order to reduce the number of rebroadcasts, we introduce the use of selfish nodes. We categorize all the MSs in a MANET into normal nodes and selfish nodes. Normal nodes rebroadcast packet for other nodes while selfish nodes do not. Selfish nodes, however, may generate data packets and request the normal nodes to forward data for them. The advantage of using selfish nodes is that the total rebroadcast traffic may be reduced. The disadvantage is that the network reach ability may be adversely affected if all the nodes in any cut set of the MANET graph are assigned to be selfish.

Clearly, a critical issue in our approach is how to decide whether a MS should be selfish or not. We choose a specified number of MSs as selfish nodes during the network set up period. All selfish nodes remain selfish during their lifetime. Our goal is to assign selfish node so that the size of the normal node set is minimized while the connectivity constraint of the resultant network is probabilistically satisfied.

II. CALCULATING SELFISH PROBABILITY

Let us assume that we set the MSs to be selfish or not during the set up period according to a given percentage p . The following section describes how to determine the value of p so that the value is small but the resultant network is highly probable to be connected. Let A be the area of an ad hoc network, N be the number of MSs in the

network, and R be the communication range. Let α be the fraction of the whole network area a MS or a node can cover.

$$\alpha = \frac{\pi R^2}{A} \quad (1)$$

The average number of neighbors for any node $N_{neighbor}$ can be obtained by using the following formula:

$$N_{neighbor} = pN\alpha - 1 \quad (2)$$

In [4], they used a counter to record the number of reception of the same broadcasting message, if the counter exceed a threshold before time expires, that means that enough neighbors of current node already rebroadcast the message and no need to rebroadcast it at the current node. In [4] it is also shown that if the counter threshold if choosing > 6 , few rebroadcasts can be saved in sparser networks. So, we believe that the number of 6 neighbors is important to get a network connected. To ensure the network connection under the condition of the existence of selfish nodes, the average number of neighbors $N_{neighbor}$ should meet the following criteria:

$$N_{neighbor} \geq 6 \quad (3)$$

$$pN\alpha - 1 \geq 6 \quad (4)$$

$$p \geq \frac{7}{\alpha N} \quad (5)$$

The existence of selfish node also leads to increased data traffic. Therefore, we need to consider the data amount as well when choose the value for parameter p . Suppose there are m source destination connections, the data rate is λ data packets per second, and the total simulation time is T . Assume that the route-unavailable-probability is β . Then the number of route request messages can be obtained by

$$N_{RREQ} = T\lambda\beta \quad (6)$$

While the total number of forwarded route request

messages in AODV (N_{AODV}) is

$$N_{AODV} = N_{RREQ}(N - 1) \quad (7)$$

And the total number of forwarded route request messages in our protocol (N_{RSN}) is

$$N_{RSN} = N_{RREQ}(pN - 1) \quad (8)$$

The main purpose of this paper is to propose a novel route discovery protocol and compare it with other protocols using analytic method. For the comparison, good put is a key performance metric. We calculate the theoretical good put and make comparison in this section.

Assume that the data packet size is k bytes, and the control packet size is N_c bytes. Let N_R be the total number of forwarded route requests. In AODV, $N_R = N_{AODV}$, while in our protocol, $N_R = N_{AODV}$. Assume the collision probability is P_c . A rough estimate of the theoretical good put G can be

$$G = \frac{\lambda Tk}{\frac{\lambda T}{1 - P_c} k + \frac{N_R}{1 - P_c} k_c} \quad (9)$$

To increase good put G above a specific value, say, G_{goal} , the following:

$$G = \frac{\lambda Tk}{\frac{\lambda T}{1 - P_c} k + \frac{N_R}{1 - P_c} k_c} \geq G_{goal} \quad (10)$$

$$p \leq \frac{\frac{k(1 - P_c - G_{goal})}{k_c \beta} + 1}{N} \quad (11)$$

According to (5) and (11), we can obtain a range for p . For example, for a MANET with range 2000m x 600m and communication range 375m and 100 nodes, we have $p \geq 0.17$. If we have $G_{goal} = 0.8$, $\beta = 0.1$, $k = 256$, $k_c = 32$ and $\beta = 0.01$, we have $p \leq 0.81$.

The choice of the value for selfish probability p is important in our approach. A higher probability value leads to larger number of selfish nodes, and hence fewer forwarded packets. A lower value means fewer selfish

nodes and more forwarding packets. If the selfish probability is too high, a broadcasting message may die out quickly. If the selfish probability is too low, then there may be little improvement over the original flooding approaches. We use the formula (11) to set the selfish probability p .

III. PROTOCOL

We describe our protocol in this section. Our protocol is straightforward given the formula for calculating selfish probability p . It is specified as follows:

1. During network set up period:

- Given the parameters A , N , R , k , K_c , P_c and, G_{goal} calculate the selfish probability range using the formula (11);
- Randomly choose the selfish probability p within the range;
- Determine if the current node should be selfish or not based on the probability p .

2. Upon receiving a route request packet after set up:

If the current node is not selfish
 Rebroadcast the packet
 Else
 Discard the packet

IV. PERFORMANCE EVALUATION

We evaluate the performance of the routing with selfish nodes in this section. We compare our algorithm with the AODV protocol. The metrics of the comparison include average number of routing request rebroadcast, average number of collisions, end-to-end delay, and throughput and reach ability.

A Simulation Setup

We use the GloMoSim network simulator [5] (version 2.03) to evaluate our algorithm. We study the performance of our algorithm and compare with the AODV [2] routing protocol included in the GloMoSim package. The AODV protocol uses simple flooding to broadcast routing requests. Instead, our algorithm is based on selfish nodes. We use two different settings of the predefined selfish node probability: RSN ($p=0.1$) and RSN ($p=0.2$).

**Table 1. Simulation Parameters
(Selfish node based approach)**

| Simulation Parameter | Value |
|----------------------|--|
| Simulator | GloMoSim (v2.03) |
| Network Range | 2000m X 600m |
| Transmission Range | 377m |
| Number of MSs | 100 |
| Bandwidth | 2Mbps |
| Traffic Type | CBR (Constant Bit Rate) |
| Packet Rate | 4 packets per second for low traffic load and 8 packets per second for high traffic load |
| Packet Size | 256 bytes |
| Simulation Time | 2020s (1000s data traffic) |

In our simulation, we set up a 2000m X 600m area with 100 MSs. The radio frequency at the physical layer is 2.4GHz of the ISM band. The raw network bandwidth is 2Mbps and the MAC layer protocol is IEEE 802.11 [6]. Other simulation parameters are shown in Table 1.

We use random waypoint as our mobility model. Each node initially selects a random movement start time, direction, speed and distance. The minimum speed is set to 0 m/s and maximum speed is set to 10 m/s. Each node chooses a random speed between the minimum speed and the maximum speed values and starts the movement until it reaches the distance. It then pauses for some time. So the pause time can describe the dynamics of the network. We vary the pause time in different scenario settings, and study the performance of our algorithm and AODV in these scenarios.

B. Analysis

The objective of this work is to present the performance of the routing with selfish nodes (RSN), using AODV as a reference. Our main idea is to reduce the rebroadcasting number in route discovery phase, therefore reduce the network traffic and decrease the probability of channel contention and packet collision. As a result, end-to-end delay can also be reduced and the throughput can be improved. In our scheme, we set the selfishness property of a MS during the network setup period based on a given probability. And there is a small chance that the route requests cannot reach the destination in our algorithm. But, by choosing a lower probability, we would have more normal nodes and the route request fail should be decreased. The AODV protocol, on the other hand, uses flooding in route discovery phase. Therefore, all route requests will reach their destinations if the network is not partitioned. However, if we consider the contention and collision, the redundant rebroadcast may reduce the chance for the route request to reach the destination. Based on this analysis, our algorithm should perform better than AODV in dense networks with heavy traffic.

C. Rebroadcasts

In AODV, a MS rebroadcasts all routing request packets that are received for the first time. Therefore there are $N-1$ rebroadcasts, where N is the total number of nodes. In our algorithm, only normal nodes will do rebroadcast, so the total number of rebroadcast is $Np-1$. Next we compare the number of rebroadcasts in AODV and in our algorithms (RSN) through simulation. Figure 1 and Fig 2 show that our algorithms can significantly reduce the number of rebroadcasts for both low traffic load and high traffic load. As shown in the figure, the saving is higher when the traffic is heavier.

Using smaller selfish probability can reduce the number of route request rebroadcasts. However, sending too few route request rebroadcasts may result in request dying out, which incur another round of route request.

V. COLLISION

We measure the number of collisions for both AODV and RSN at physical layer. Since data packets and control packets share the same physical channel, the collision probability is higher when there are a large number of control packets. Fig 3 and Fig 4 show the average number of collisions during the whole simulation period for low traffic load and high traffic load.

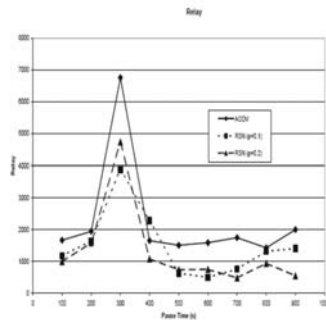


Fig 1 The Number of Rebroadcasts vs. Mobility (Low Traffic Load)

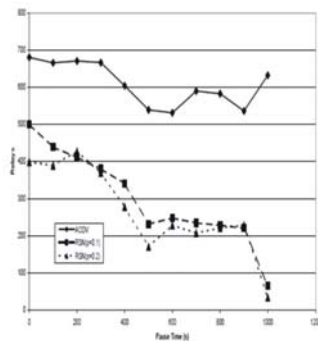


Fig 2 The Number of Rebroadcasts vs. Mobility (High Traffic Load)

In these figures, we can see that RSN has lower collision compares to AODV.

VI. LATENCY (END-TO-END DELAY)

We measure end-to-end delay for the AODV and the RSN. The number of total packets transmitted in the channel has a significant impact on the latency. If the number of packets is higher, then the number of

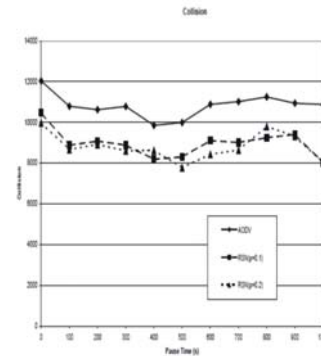


Fig 3 The Number of Packet Collisions vs. Mobility (Low Traffic Load)

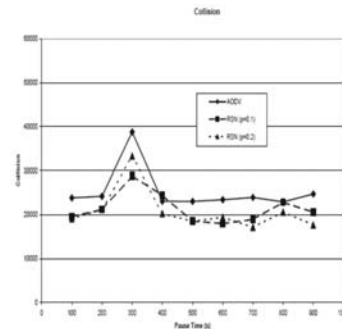


Fig 4 The Number of Packet Collisions vs. Mobility (High Traffic Load)

collisions is also high, and in turn more retransmissions are needed. As a result, fewer packets lead to lower delays. Fig 5 and Fig 6 show the end-to-end delay for low traffic load and high traffic load. As expected, our RSN algorithm exhibits lower latency than the AODV.

In Fig 6, we compare end-to-end latency for different mobility settings. The figure shows AODV incurs higher latency. This is due to the fact that the too many redundant rebroadcast cause the contention and collision problem and many route requests failed to reach the destinations. Thus another route request is required and the latency is prolonged.

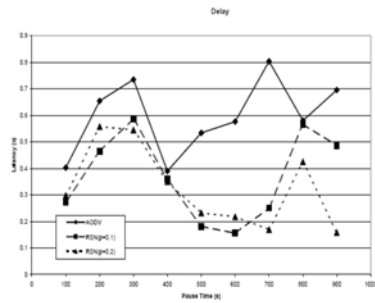


Fig 5 Latency vs. Mobility (Low Traffic Load)

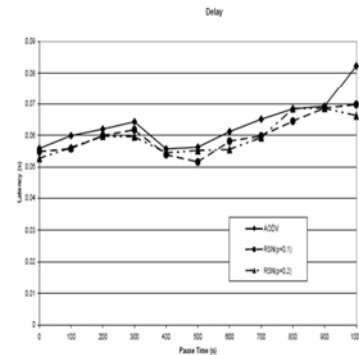


Fig 6 Latency vs. Mobility (High Traffic Load)

VII. THROUGHPUT

Throughput is an important metric that measures the transmission ability of a network. It is defined as the number of bits transmitted in a unit time. We compare the throughput of AODV and RSN in this section. Fig 7 we compare the network throughput for different mobility setting. The Fig shows that RSN outperforms AODV when high traffic load applies.

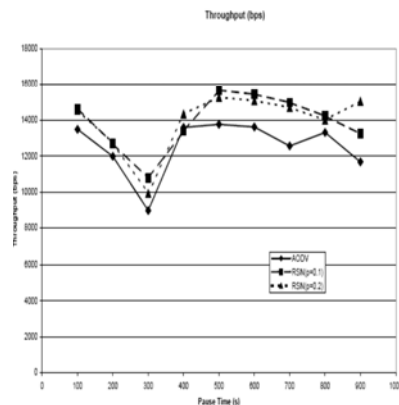


Fig 7 Throughput vs. Mobility (High Traffic Load)

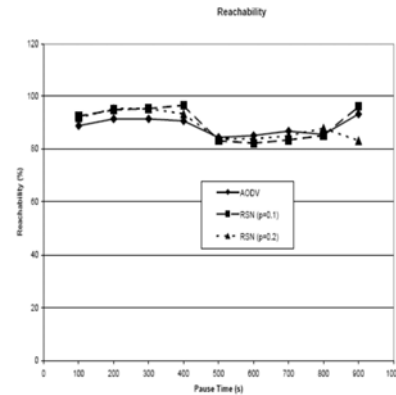


Fig 8 The Reachability vs. Mobility (Low Traffic Load)

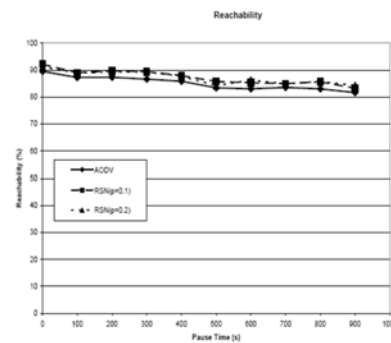


Fig 9 The Reachability vs. Mobility (High Traffic Load)

VIII. REACHABILITY

We measure the delivery rate of a broadcast message for both AODV and RSN at physical layer. Figure 8 and Figure 9 respectively show the average reach ability during the whole simulation period for low traffic load and high traffic load. In figures, we can see that RSN has lower collision compares to AODV.

IX. CONCLUSION

In this paper, we introduced a routing approach based on selfish nodes for MANETs. Our approach classifies MSs into selfish nodes and normal nodes. Selfish nodes always refuse to relay packets for others while normal nodes always relay packets. Simulation results show that our approach outperforms the AODV protocol in dense networks since the approach assigns relatively low forwarding probability to selected nodes, thus leading to fewer rebroadcasts.

This leads to an intuitive feeling that assigning different forwarding probabilities to nodes, may lead to smaller number of rebroadcasts.

REFERENCES

- [1] D. P. Agrawal and Q-A Zeng, 2003. *Introduction to Wireless and Mobile Systems*, Brooks/Cole Publishing, ISBN No. 0534-40851-6, 436 pages.
- [2] C. E. Perkins and E. M. Royer, February 1999. "Ad-hoc on-demand distance vector routing," *Proceedings of 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90-100.
- [3] Z. Haas, J. Halpern, and L. Li, 2000. "Gossip-based Ad Hoc Routing," in *Proceedings of the IEEE INFOCOM*.
- [4] Y. Tseng, S. Ni, and E. Shih, May 2003. "Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Network," *IEEE Transactions on Computers*, Vol. 52, No. 5, pp. 545-557.
- [5] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks," *Proceedings of the 12th Workshop on Parallel and Distributed Simulations. PADS '98*, May 26-29, Banff, Alberta, Canada.
- [6] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Standard 802.11-1997. The Institute of Electrical and Electronics Engineers, New York.



Jatinder Singh was born in Punjab, India, in 1981. He received the B.E. degree in electronics and communication engineering in 2003. He is currently pursuing M.E. in electronics. His current main research interests are in optical communication and MANET.



Rajeev kumar Ranjan was born in India on 1979. He received the B.tech in Electronics and communication Engineering from SLIET, Longwal in 2003 and M.tech Electronics design and technology from Tezpur (Central) University in 2007. He is currently pursuing the phd in VLSI design. His current research in communication and VLSI