Volume 5, Issue 5, May 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Overhead Evaluation of Ad Hoc on Demand Distance Vector Routing Protocols Using NS2 Simulation

Geetanjali Rautela, Rashi Pandey, Bhavya Saxena
Computer Science & Engineering, Graphic Era Hill University,
Dehradun, Uttarakhand, India

Abstract*— Mobile Ad-Hoc Network (MANET) is an infrastructure less and highly challenged network environment due to its dynamic topology which needs a robust dynamic routing protocol. Ad-hoc on demand distance vector routing (AODV) is well-suited for mobile ad-hoc network, it has low memory overhead and low network utilization. In this paper, an attempt has been made to compare the effects of Hello packets in the performance of the AODV protocol for mobile ad hoc networks in different scenarios. We studied many factors that contribute to the overall performance of the protocol, and focused on one of them -control messages (Hello packets). Messages affect the parameter of protocols like throughput, overhead, hidden energy consumption, etc. We examine how the overhead of AODV protocol can be improved.

***Keywords*— MANET, AODV, Routing Overhead, Throughput, NS2, Hello Packets, NRL.**

I. INTRODUCTION

Mobile Ad hoc Networks (MANETs) [1] are autonomous networks, consist of a collection of wireless mobile nodes which can communicate without the need of fixed infrastructure or a wired backbone network. These networks are fully distributed. The change in topology in MANET makes routing as a crucial issue in the design of the MANET. In MANET routing protocol should adapt to dynamic topology and to maintain a connection between the communicating nodes even if path breaks due to mobility and or node failures. Routing in ad hoc networks faces a number of challenges like dynamic topology, node mobility, lack of infrastructure, low battery life, insecure medium and limited channel capacity.

Because of the decentralized infrastructure, MANETs have to rely on flooding based route discovery/maintenance mechanisms, which generates control overhead. In this paper, we have studied the overheads due to Hello messages [2] and proposed an approach how to reduce broadcast messages and simultaneously enhanced the performance of Ad-hoc on demand distance vector routing protocol in MANET.

The rest of the paper is organized as follows: Section 2 gives an overview of Reactive Routing Protocol AODV. Section 3 presents the related work on improvement of overhead in AODV. Section 4 presents the simulation results and analysis. We have drawn conclusions in Section 5.

II. OVERVIEW OF AODV

Ad-hoc On-Demand Distance Vector Routing (AODV)[3] is a source-initiated on-demand routing protocol, it discovers routes only when they are needed by a source node and maintain active routes only while they are in use. An AODV routing protocol provides unicast **[4]**, broadcast and multicast [5] communication in ad hoc mobile network.

AODV adopts a very different mechanism to maintain routing information. It uses traditional routing tables, one entry per destination. Without source routing, AODV relies on routing table entries to propagate a RREP back to the source and, subsequently, to route data packets to the destination [6].

AODV uses a broadcast mechanism for route discovery. AODV combine the concept of DRS and DSDV routing protocol [7]. The combination of these techniques yields an algorithm that uses bandwidth efficiently (by minimizing the network load for control and data traffic) is responsive to changes in topology and ensures Loop free routing [2].

AODV routing protocol uses different messages to discover and maintain links. These messages are Route Request (RREQ), Route Request (RREQ), and hello messages (a special RREP) which are periodically broadcasted to the immediate neighborhood.

A. Route Discovery

The Route Discovery process is initiated when a source node needs to communicate with another node for which it has no routing information in its table [2]. Route Discovery process follows a route request/route reply cycle. The source node broadcasts a Route Request message (RREQ). Any node with a current route to a particular destination can unicast a Route Reply message (RREP) to the source node. If the intermediate node does not have a route to the destination, it rebroadcasts the RREQ. When the source node receives a RREP, it establishes a route from source to destination, then the source can start using it.

B. Path Maintenance

AODV maintains routes using a soft state approach. AODV removed older unused entries from the table. When a node detects that a route to a neighbor no longer is valid, it sends a link failure (RERR) message to the neighbor that are actively using the route informing them that this route is no longer valid. Each intermediate node propagates the RERR message towards the source. When the source of the data receives the RERR, it invalidates the route and initiate route discovery if necessary. [8]

C. Local Connectivity Management

Nodes learn their neighbor through broadcasting Hello messages. Whenever a node receives a Hello message [8] from a neighbor it updates its local routing table if a route to the neighbor does not already exist. Otherwise, increasing its lifetime. If hello messages are not received from the next hop along an active path, the active neighbors using that next hop are sent notification of link failure. [2]

III. RELATED WORKS

This section analyses the related work which directly or indirectly aims at reducing the number of broadcast packets, control overhead, energy consumption generated by the control messages.

Perkins et al. [2], discussed the reasons for applying hello messages with AODV and presented some drawbacks for using these messages. They mentioned Hello messages, create extra control overhead and increase bandwidth consumption. They will investigate other ways to eliminate these messages.

Srdjan Krco et al**.** [9] Have proposed differentiation between 'good' and 'bad' neighbors, i.e. between good and bad quality links. They used Signal to Noise Ratio (SNR) value to detect the neighbor. AODV control (including hello) messages accepted only from 'good' neighbors. They increase the stability of the network. It means that once a route is established, it is not lost because of weak signal. This neighbor detection solution effectively improves data throughput and decrease delays.

Joo-Han Song et al**.** [10] Presented an extension of ad hoc on demand distance vector (AODV) routing protocol, called LB-AODV. They propose a load-balancing mechanism based on the concept of grouping. It reduces the number of unnecessary retransmissions of routing messages and prevents network congestion. They provided advantages over AODV in terms of throughput and routing overhead. It reduces the overhead of routing messages.

Chakeres et al. [11] Have examined the effectiveness of hello messages for monitoring link status and found some influencing factors on the utility of these messages, gave result better throughput.

Seon Yeong Han et al. [12] They proposed an adaptive Hello messaging scheme for neighbor discovery. The proposed scheme dynamically adjusts Hello intervals, and does not increase the risk that a sender will transmit a packet through a broken link that has not been detected by Hello messaging. Instead of using a constant Hello interval, the proposed scheme uses a constant risk level. As the event interval increases, the Hello interval can also increase without increasing risk. If the event interval is extremely large, the Hello messaging interval is also correspondingly large; that is Hello messaging is practically suppressed. The proposed scheme decreases network overhead and hidden energy consumption.

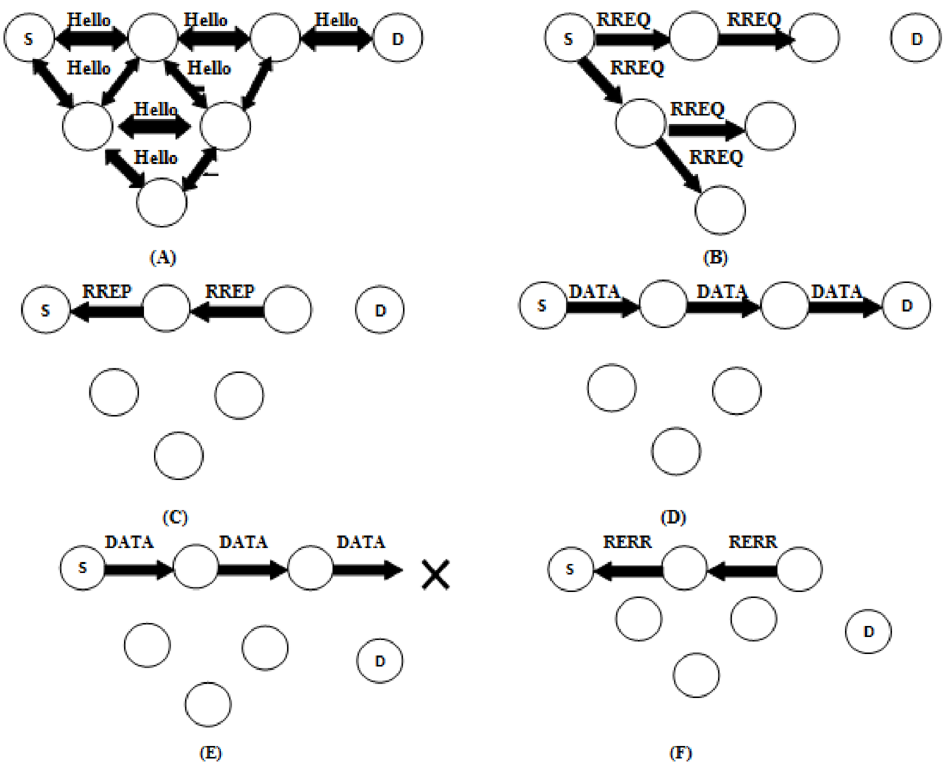

Fig 1. AODV Operation

IV. SIMULATION RESULTS AND ANALYSIS

Simulation results are based on the comparison of different Hello Timer and Node expire time. We use Ns-2 v.2.35 to do the simulations. There are four node scenarios as 6 nodes, 11nodes, 22 nodes and 40 nodes respectively and random node movement. In our simulation experiments, we consider Normalized routing overhead. The Normalized routing overhead is defined as:

$$NRL = \frac{Total\ number\ of\ routing\ packets\ sent}{Total\ number\ of\ received\ data\ packets}$$

Three scenarios are discussed in this section. For each Hello Timer, we have three different Expire Time. In the first scenario, we use the Hello Timer=MinHelloInterval+ ((MaxHelloInterval -MinHelloInterval) *Random:: uniform ()) and three different nodes expire time and analyze the routing overhead of three different expire time in the different node scenario. And we find the best result in different node scenarios.

Generally, when we increase node expiry time in first scenario we find the following results:

Table 1.NRL at Hello Timer=MinHelloInterval + ((MaxHelloInterval -MinHelloInterval) *Random:: uniform ())

	Expiry Time		
No Of Nodes	CURRENT_TIME + (1.5*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	CURRENT_TIME + (1.75*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	CURRENT_TIME + (2*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)
6	0.042	0.041	0.037
11	0.043	0.042	0.041
22	0.105	0.100	0.079
40	0.143	0.141	0.138

In the second scenario and the third scenario, we use the HelloTimer=MinHelloInterval+(1+(MaxHelloInterval-MinHelloInterval)*Random::uniform()) and HelloTimer=MinHelloInterval+(2+(MaxHelloInterval-MinHelloInterval) *Random:: uniform ()) and following the same procedure as the first scenario.Results of second and third scenario are as follows:

Table II.NRL at Hello Timer=MinHelloInterval + (1+ (MaxHelloInterval -MinHelloInterval) *Random:: uniform ())

	Expiry Time		
No of Nodes	CURRENT_TIME + (1.5*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	CURRENT_TIME + (1.75*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	CURRENT_TIME + (2*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)
6	0.041	0.040	0.029
11	0.043	0.042	0.041
22	0.086	0.081	0.079
40	0.148	0.144	0.136

Table III.NRL at Hello Timer=MinHelloInterval + (2+(MaxHelloInterval -MinHelloInterval) *Random:: uniform ())

	Expiry Time		
No of Nodes	CURRENT_TIME + (1.5*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	CURRENT_TIME+ (1.75*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	CURRENT_TIME+ (2*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)
6	0.042	0.041	0.039
11	0.045	0.043	0.042

22	0.085	0.085	0.080
40	0.142	0.098	0.140

Table IV.Best Normalized Routing Overhead

No of Nodes	Expiry Time	Hello Timer	NRL
6	CURRENT_TIME+(2*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	MinHelloInterval+(1+(MaxHelloInter-val -MinHelloInterval) *Random:: uniform ())	0.029
11	CURRENT_TIME+(2*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	MinHelloInterval+(1+(MaxHelloInter-val -MinHelloInterval) *Random:: uniform ())	0.041
22	CURRENT_TIME+(2*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	MinHelloInterval+(1+(MaxHelloInter-val -MinHelloInterval) *Random:: uniform ())	0.079
40	CURRENT_TIME+(1.75*ALLOWED_HELLO_LOSS*HELLO_INTERVAL)	MinHelloInterval+(2+(MaxHelloInter-val -MinHelloInterval) *Random:: uniform ())	0.098

After comparing the performance in different scenarios, we get the above results. We found that increasing the Hello-Timer and Expiry time; improve the Normalized Routing Overhead of AODV protocol.

V. CONCLUSIONS

We have compared the performance of AODV in different scenarios. The performance evaluation of AODV has been done with respect to normalized routing overhead under varying no of nodes, varying Hello Timer and varying node expiry time. From the result analysis, it has been observed that the increasing Hello Timer and node expiry time, improve the normalized routing overhead in AODV protocol.

ACKNOWLEDGMENT

We extend our sincere and warmest thanks to our advisor Mr.Kamlesh Chandra Purohit, Assistant Professor, Graphic Era University for his untiring supervision, valuable guidance, peerless and constructive criticism and continuous encouragement.

REFERENCES

[1] Bai, Rendong, and Mukesh Singhal. "DOA: DSR over AODV routing for mobile ad hoc networks." Mobile Computing, IEEE Transactions on 5.10 (2006): 1403-1416.

[2] Perkins, Charles E., and Elizabeth M. Royer. "Ad-hoc on-demand distance vector routing." Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on. IEEE, 1999.

[3] Royer, Elizabeth M., and Chai-Keong Toh. "A review of current routing protocols for ad hoc mobile wireless networks." Personal Communications, IEEE 6.2 (1999): 46-55.

[4] Royer, Elizabeth M., and Charles E. Perkins. "An implementation study of the AODV routing protocol." Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE. Vol. 3. IEEE,

[5] Royer, Elizabeth M., and Charles E. Perkins. "An implementation study of the AODV routing protocol." Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE. Vol. 3. IEEE.

[6] Perkins, Charles E., et al. "Performance comparison of two on-demand routing protocols for ad hoc networks." Personal Communications, IEEE 8.1 (2001): 16-28.

[7] Abolhasan, Mehran, Tadeusz Wysocki, and Eryk Dutkiewicz. "A review of routing protocols for mobile ad hoc networks." *Ad hoc networks* 2.1 (2004): 1-22.2000.

[8] Chakeres, Ian D., and Elizabeth M. Belding-Royer. "AODV routing protocol implementation design." Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on. IEEE, 2004.

[9] Krco, Srdjan, and Marina Dupcinov. "Improved neighbor detection algorithm for AODV routing protocol." Communications Letters, IEEE 7.12 (2003): 584-586.

[10] Joo-Han, Vincent WS Wong, and Victor CM Leung. "Efficient on-demand routing for mobile ad hoc wireless access networks." Selected Areas in Communications, IEEE Journal on 22.7 (2004): 1374-13'83.
[11] Chakeres I. D. And Royer, E. M., "The Utility of Hello Messages for Determining Link Connectivity," in Proceedings of the 5th International Symposium on Wireless Personal Multimedia Communications (WPMC), Honolulu, Hawaii, pp. 504-508, October 2002.
[12] Han, Seon Yeong, and Dongman Lee. "An Adaptive Hello Messaging Scheme for Neighbor Discovery in On-Demand MANET Routing Protocols." Communications Letters, IEEE 17.5(2013) 1040-1043.

