

RESEARCH PAPER

Available Online at www.jgrcs.info

STUDY AND PERFORMANCE ESTIMATION OF ROUTING PROTOCOLS MANET

Anuj Jain¹, Dr. Y.K.Jain²

¹Research Scholar-Mewar University, Rajasthan, ² Professor- Mewar University
Corresponding author's email: - akj_1112@rediffmail.com

Abstract: A mobile ad hoc network consists of wireless hosts that may move often. Movement of hosts results in a change in routes, requiring some mechanism for determining new routes. Several routing protocols have already been proposed for ad hoc networks. In mobile communications, effective internetworking is mandatory in order to support user roaming among various types of wireless networks while maintaining connectivity. In this paper we have study and performance estimation of routing protocols AODV and DSR. Both share similar On-Demand behaviour, but the protocol's internal mechanism leads to significant performance difference. We have analysed the performance of protocols by varying network load, mobility and type of traffic (CBR and TCP). A detailed simulation has been carried out in NS2. The metrics used for performance analysis are Packet Delivery Fraction, Average end-to-end Delay, Routing Overhead and Normalized Routing Load. It has been observed that AODV gives better performance in CBR traffic and real time delivery of packet. Whereas DSR gives better results in TCP traffic and under restricted bandwidth condition.

Keywords: MANET, TCP, CBR,

INTRODUCTION

In mobile environments without infrastructure, nodes must be self-organized and create a wireless ad hoc network in order to communicate or exchange messages with each other. Providing reliable and efficient networking services in a rapidly moving environment such as vehicular networks and tactical scenarios, however, is very challenging due to high mobility and external interference (e.g., jamming by the adversary). In such a "disruptive" environment, conventional links, networks, and transport protocols fail to operate properly; thus, a significant fraction of the packets can be received in error and/or can be lost. In a Mobile Ad Hoc Network, nodes move arbitrarily, therefore the network may experience rapid and unpredictable topology changes. Routing paths in MANETs potentially contain multiple hops, and every node in MANET has the responsibility to act as a router [4]. Routing in MANET has been a challenging task ever since the wireless networks came into existence. The major reason for this is the constant change in network topology because of high degree of node mobility. A number of protocols have been developed to accomplish this task. There are various mobility models such as Random Way Point, Reference Point Group Mobility Model (RPGM), Manhattan Mobility Model, Freeway Mobility Model, Gauss Markov Mobility Model etc. that have been proposed for evaluation [8, 15]. Several performance evaluation of MANET routing protocols using CBR traffic have been done by considering various parameters such as mobility, network load and pause time. Biradar, S. R. et. al. [13] have analysed the AODV and DSR protocol using Mobility Model and CBR traffic sources. Biradar, S. R. et. al. [13] investigated that DSR performs better in high mobility and average delay is better in case of AODV for increased number of groups. Also Rathy, R.K. et. al. [10] investigated AODV and DSR routing protocols under Random Way Point Mobility Model with TCP and CBR traffic sources. They concluded that AODV outperforms DSR

in high load and/or high mobility situations. In this paper we have investigated the performance of AODV and DSR On-Demand (reactive) routing protocol for performance comparison such as military battlefield. The purpose of this work is to understand their working mechanism and investigate that which routing protocol gives better performance in which situation or traffic.

DESCRIPTION OF ROUTING PROTOCOL

Routing protocols for mobile ad-hoc networks are classified into three different groups: proactive, reactive and hybrid protocols. The first protocols developed derive from static networks and require periodic advertisement and global dissemination of connectivity information for correct operation, which leads to frequent system-wide broadcasts. In Destination-Sequenced Distance Vector Routing (DSDV), for instance, every mobile node in the network holds a routing table where it lists all possible destinations and the corresponding hop counts to them.

The protocol transmits full dump and incremental control packets in order to update the route information in the entire network. This potentially large amount of network traffic strongly limits the use of this protocol to small ad-hoc networks.

A reactive protocol establishes a route only when needed, that is, if a node is willing to transmit data and is not aware of a route to the destination. Most often, reactive protocols rely on the transmission of route request and route reply messages, which are needed to establish and maintain the routes. Routes are only recorded as long as they are valid and expire after some idle time. On-demand

route establishment leads to a drastic reduction of control traffic required for routing. At AODV and DSR we present two examples of reactive routing protocols.

A hybrid protocols combine proactive and reactive aspects. Mostly, proactive mechanisms are used in the local neighbourhood of a node to establish routes within a limited radius. Thus, broadcasting of routes through the entire network is avoided. Routing between distant nodes is still performed by on-demand routing. Hybrid routing is illustrated by means of the ZRP protocol.

2.1 Ad-Hoc On-Demand Distance Vector Routing

The Ad-Hoc On-Demand Distance Vector routing protocol (AODV) described in [2] is it tries to minimize the number of required broadcasts. It creates the routes on a on demand basis, as opposed to maintain a complete list of routes for each destination.

2.1.1 Path Discovery Process

When trying to send a message to a destination node it initiates a path discovery process. A route request message (RREQ) is broadcasted to all neighbours, which continue to broadcast the message to their neighbours and so on. The forwarding process is continued until the destination node is reached or until an intermediate node knows a 'fresh enough' route to the destination (see figure 1(a)). To ensure loop-free and most recent route information every node maintains two counters: *sequence number* and *broadcast_id*. The *broadcast_id* and the address of the source node uniquely identify a RREQ message. *broadcast_id* is incremented for every RREQ initiated by the source node. An intermediate node can receive multiple copies of the same route request broadcast from various neighbours. In this case – if a node has already received a RREQ with the same source address and *broadcast_id* it will discard the packet without broadcasting it furthermore. When an intermediate node forwards the RREQ message, it records the address of the neighbour, which it received the first copy of the broadcast packet from. This way, the *reverse path* from all nodes back to the source is being built automatically. The RREQ packet contains two sequence numbers: the source sequence number and the last destination sequence number known by the source. The source sequence number is used to maintain 'freshness' information about the reverse route to the source while the destination sequence number specifies the actuality a route to the destination must have before being accepted by the source. As soon as the route request broadcast reaches the destination or an intermediate node with a fresh enough route, the node responds by sending a unicast route reply packet (RREP) back to the node which it received the RREQ from (figure 1b). Actually, the packet is sent back reverse the path that has been built during broadcast forwarding

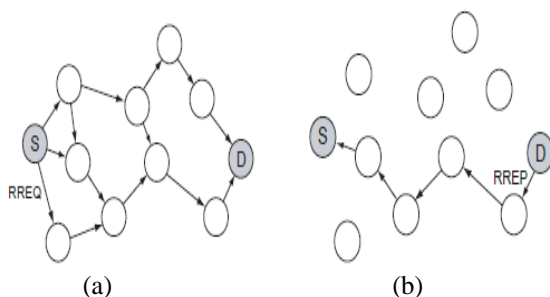


Fig 1: AODV Path Discovery Process.

2.1.2 Maintaining Routes

If the source node moves, it is able to send a new RREQ packet in order to find a new route to the destination. If an intermediate node along the forward path moves, its upstream neighbour notices the move and sends a link failure notification message to each of its active upstream neighbours to inform them of the erasure of that part of the route. The link failure notification is forwarded as long as the source node has not been reached. After having noticed the failure the source node may reinitiate the route discovery protocol. Optionally a mobile node may perform local connectivity maintenance by periodically broadcasting hello messages.

2.2 Dynamic Source Routing

The Dynamic Source Routing (DSR) protocol [4] is an on-demand routing protocol based on source routing. In the source routing technique a sender determines the exact sequence of nodes, which to propagate a packet through. The list of intermediate nodes for routing is explicitly contained in the packet's header. In DSR every mobile node in the network needs to maintain a route where it keeps source routes that it has learned. When wanting to send a packet to some other host, the initiating host first checks its route cache for a source route to the destination. In case of finding a route the sender propagates the packet that way. Otherwise the source node initiates the route discovery process.

2.2.1 Route Discovery

For route discovery the source node starts by broadcasting a route request packet that may be received by all neighbour nodes within its wireless transmission range. The route request contains the address of the destination host, referred to as the *target* of the route discovery, the source's address, a *route record* field and a unique identification number. At the end, the source host receives a route reply packet containing a list of network nodes, which it should propagate the packets through, provided that the route discovery process was successful. A simple example is illustrated in figure 2.

During the route discovery the route record field is used to accumulate the sequence of hops already taken. First of all, the sender initiates the route record as a list with a single element containing itself. The next neighbour node appends itself to the list and so on. Each route request packet also contains a unique identification number called *request_id*. *request_id* is a simple counter, which is increased whenever a new route request packet is being sent by the source node. Thus, every route request packet can be uniquely identified through its initiator's address and *request_id*. When a host receives a route request packet, it is important to process the request in the order described below. This way we make sure no loops will occur during the broadcasting of the packets.

1. If the pair h source node address, $request_id$ i is found in the list of recent route requests, the packet is discarded.
2. If the host's address is already listed in the request's route record, the packet is also discarded. This ensures removal of later copies of the same request that may arrive by using a loop.
3. If the destination address in the route request matches the host's address, the route record field contains the entire list of nodes, which have been passed through in order to reach the

destination from the source node. A route reply packet is sent back to the source node containing a copy of this route.

4. Otherwise, add this host's address to the route record field of the route request packet and rebroadcast the packet.

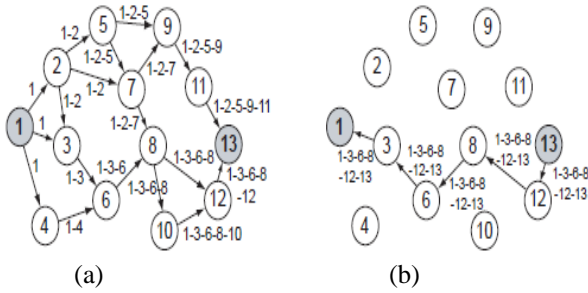


Fig 2: DSR Route Discovery Process

A route reply is sent back either if the request packet reaches the destination node itself or if the request reaches an intermediate node, which has an active route to the destination in its route cache. The route record field in the request packet indicates the sequence of hops taken. If the generating node for the route reply is the destination node, it only takes the route record field of the route request and puts it into the route reply. If the responding node is an intermediate node, it appends the cached route to the route record and then generates the route reply.

2.2.2 Route Maintenance

Route maintenance can be accomplished by two different processes:

- Hop-by-hop acknowledgment at the data link layer
- End-to-end acknowledgments

Hop-by-hop acknowledgment at the data link layer allows an early detection and retransmission of lost or corrupt packets. If the data link layer determines a fatal transmission error (for example, because the maximum number of retransmissions is exceeded), a route error packet is being sent back to the sender of the packet. The route error packet contains two parts of information: The address of the node detecting the error and the host's address, which it was trying to transmit the packet to. Whenever a node receives a route error packet, the hop in error is removed from the route cache and all routes containing this hop are truncated at that point.

End-to-end acknowledgment may be used, if wireless transmission between two hosts does not work equally well in both directions. As long as a route exists, by which the two end hosts are able to communicate, route maintenance is possible. There may be different routes in both directions. In this case, replies or acknowledgments on the application or transport layer may be used to indicate the status of the route from one host to the other. However, with end-to-end acknowledgment it is not possible to find out the hop, which has been in error.

SIMULATION SETUP

We have used Network Simulator (NS)-2 in our evaluation. The NS-2 is a discrete event driven simulator [5,6] developed at UC Berkeley. We have used Red Hat Linux environment with version NS-2.34 of network simulator. NS-2 is suitable for designing new protocols, comparing different protocols and traffic evaluations. It is an object oriented simulation written in C++, with an OTcl interpreter as a frontend. NS

uses two languages because simulator got to deal with two things:

- Detailed simulation of protocols which require a system programming language which can efficiently manipulate bytes, packet headers and implement algorithms,
- Research involving slightly varying parameters or quickly exploring a number of scenarios. is generated by software called Mobility Generator which is based on a frame. In the scenario we have considered four groups with twelve node and one group leader in each.

Table 1: Simulation Parameter

S.No.	Parameter	Value
1	Terrain Size	1200 X 1200 meters
2	Mobility Model	Random Waypoint
3	[Min, Max] speeds	(0 ms ⁻¹ , 20 ms ⁻¹)
4	MAC Protocol	CSMA
5	Routing Protocol	AODV
6	Nominal traffic type	Constant Bit Rate (CBR), 12 connections
7	Number of Nodes	50

We have used four traffic patterns with varying number of sources for each type of traffic (TCP and CBR). The source-destination pair may be in same group or in different group. The goal of our simulation is to evaluate the performance differences of these two on-demand routing protocols. The type of traffic (CBR and TCP) and the maximum number of sources are generated by inbuilt tool of NS2 [6]. The parameters used for carrying out simulation are summarized in the table 1.

3.1 Performance Metrics

A number of quantitative metrics that can be used for evaluating the performance of MANET routing protocols. We have used the following metrics for evaluating the performance of two on-demand reactive routing protocols (AODV & DSR)

3.1.1 Packet Delivery Fraction

It is the ratio of data packets delivered to the destination to those generated by the sources. It is calculated by dividing the number of packet received by destination through the number packet originated from source.

PDF = (Pr/Ps)*100 Where Pr is total Packet received & Ps is the total Packet sent.

3.1.2 Routing Overhead

It is the total number of control or routing (RTR) packets generated by routing protocol during the simulation. All

packets sent or forwarded at network layer is consider routing overhead.

Overhead = Number of RTR packets

3.1.3 Normalized Routing Load

Number of routing packets “transmitted” per data packet “delivered” at destination. Each hop-wise transmission of a routing is counted as one transmission. It is the sum of all control packet sent by all node in network to discover and maintain route. $NRL = \text{Routing Packet}/\text{Received Packets}$

3.1.4 Average End-to-End Delay (second)

This includes all possible delay caused by buffering during route discovery latency, queuing at the interface queue, retransmission delay at the MAC, propagation and transfer time. It is defined as the time taken for a data packet to be transmitted across an MANET from source to destination. $D = (Tr - Ts)$ Where Tr is receive Time and Ts is sent Time.

RESULT AND DISCUSSION

Packet delivery ratio:

In case of CBR traffic both protocols delivers almost all originated data packets (around 99-100%) when mobility is low and number of sources is also low (10). But the packet delivery fraction starts degrading gradually when there is increase in number of sources (40) and with the increase in speed of nodes. DSR perform better when number of sources is low, but when network load increases, packet delivery ratio decreasing. AODV perform equally under all assumed load condition in CBR traffic (fig. 4). But in case of TCP traffic, DSR performs better irrespective of network load and speed (fig. 5).

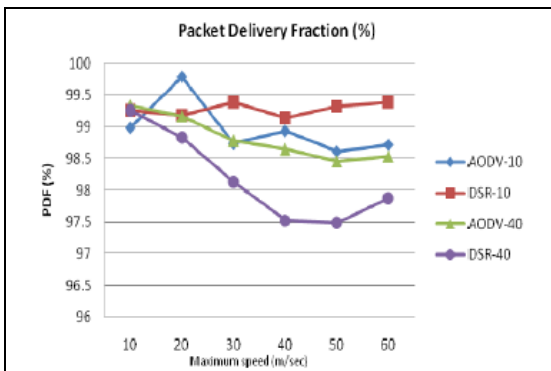


Fig 4: CBR Fraction Delivery Fraction vs. Speed

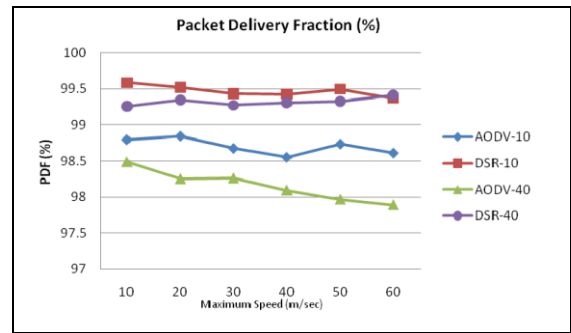


Fig 5: TCP Fraction Delivery Fraction vs. Speed

Routing Overhead:

For CBR traffic, DSR protocol have significantly low routing overhead than AODV (fig. 6) when the mobility is increased. We have investigated that, when number of sources is low (10), the performance of DSR and AODV is similar regardless of mobility.

But with large number of sources (40), DSR starts outperforming AODV for high mobility scenario. Further, DSR always have a lower routing overhead than AODV. In DSR route replies contribute to large fraction of routing overhead. Also in case of TCP traffic DSR performs better than AODV (fig. 7).

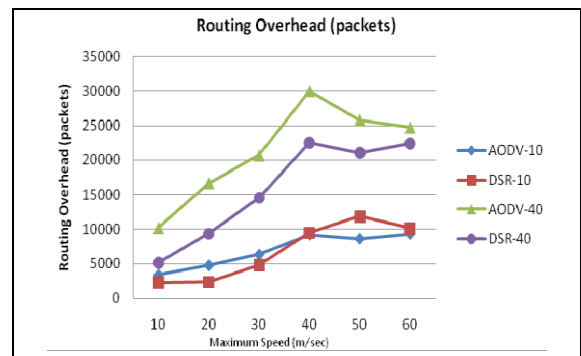


Fig 6: CBR Routing Overhead vs. Speed

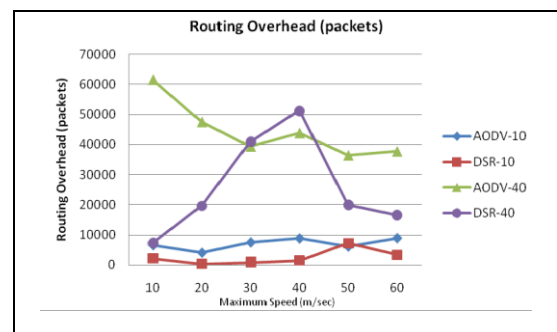


Fig 7: TCP Routing Overhead vs. Speed

Normalized Routing Load:

In case of CBR traffic, with low number of sources (10) and low mobility, DSR performs better. But when the mobility increases, AODV perform better than DSR. But when number of sources is high (say 40), DSR perform better than AODV as shown in (Fig 8). In case of TCP traffic, at low network load (10) both (AODV & DSR) gives almost similar performance. But when number of sources is high say 40 AODV perform better than DSR as shown in (Fig. 9).

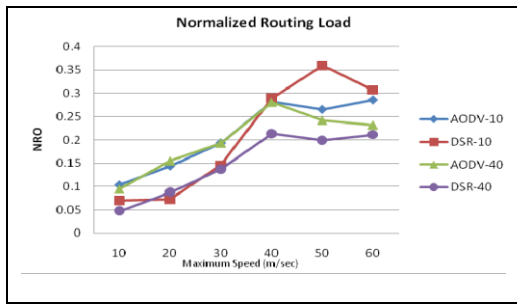


Fig 8: CBR Normalized Routing Load vs. Speed

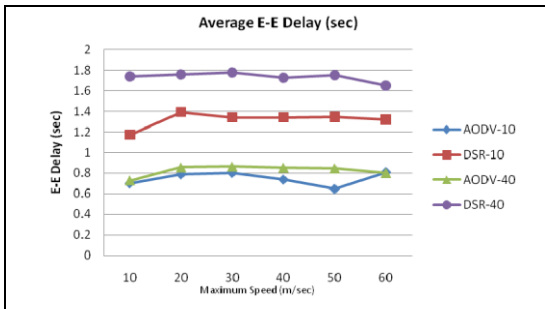


Fig 9: TCP Normalized Routing Load vs. Speed

Average end-to-end Delay:

In CBR traffic, average end-end delay of DSR is comparable to AODV when number of sources is low (10), but with the increase in network load (say 40), delay in DSR is too much higher than AODV (fig. 8). But in case of TCP traffic, AODV perform better in all condition (fig. 10). Over all in case of real time packet delivery, AODV is better choice. DSR produce more delay due to route caching. Average end-end delay in case of TCP traffic is at least three times more than CBR traffic.

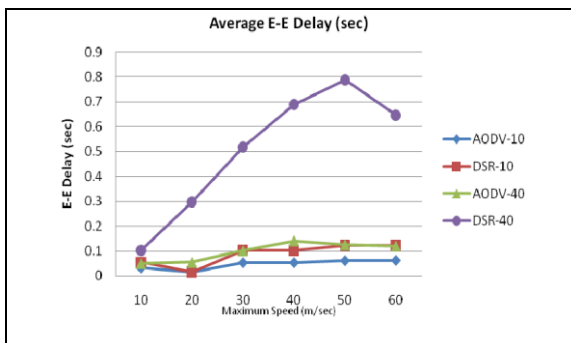


Fig 10: Average End-End Delay vs. Speed

5. Conclusions.

In Manet with CBR traffic sources AODV perform better. But in case of TCP traffic, DSR perform better in stressful situation (high load or high mobility). DSR routing load is always less than AODV in all type of traffic. Average end- to-end delay of AODV is less than DSR in both type of traffic. Over all the performance of AODV is better than DSR in CBR traffic and real time delivery of data. But DSR perform better in TCP traffic under restriction of bandwidth condition. In this paper, two routing protocol are used and their performance have been analysed with respect to four performance metrics. This paper can be enhanced by analysing the other MANET routing protocols under different

mobility model and different type of traffic sources with respect to other performance metrics.

REFERENCES

- [1]. S. Das, C. E. Perkins, E. Royer, “Ad Hoc On Demand Distance Vector (AODV) Routing”, IETF Draft, June 2002.
- [2]. C-K Toh “Ad Hoc Mobile Wireless Networks Protocols and Systems”, First Edition, Prentice Hall Inc, USA, 2002.
- [3]. C.E. Perkins and E.M.Royer, “Ad-Hoc On Demand Distance Vector Routing”, Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, USA, pages 90-100, February 1999.
- [4]. Elizabeth M. Royer and Chai-Keong Toh, “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks”, IEEE Personal Communications, pages 46-55, April 1999.
- [5]. UCB/LBNL/VINT Network Simulator, <http://wwwmash.cs.berkeley.edu/ns/> referred on March 2010.
- [6]. “The Network Simulator - ns-2,” available at <http://www.isi.edu/nsnam/ns/> referred on March 2010.
- [7]. Fan Bai, Ahmed Helmy “A Framework to systematically analyze the Impact of Mobility on Performance of Routing Protocols for Ad hoc Networks”, IEEE INFOCOM 2003.
- [8]. Tracy Camp, Jeff Boleng, Vanessa Davies “A Survey of Mobility Models for Ad Hoc Network Research”, Wireless Communication & Mobile Computing (WCMC): vol. 2, no. 5, pp. 483-502, 2002.
- [9]. D. Johnson, Dave Maltz, Y Hu, Jorjeta Jetcheva, “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks”, Internet Draft, February 2002.
- [10]. Suresh Kumar, R.K. Rathy and Diwakar Pandey, “Traffic Pattern Based Performance Comparison of Two Reactive Routing Protocols for Ad-hoc Networks using NS2”, 2nd IEEE International Conference on Computer Science and Information Technology, 2009.
- [11]. D. Johnson, Y. Hu, and D. Maltz, “The Dynamic Source Routing Protocol (DSR) for Mobile”, RFC 4728, Feb 2007.
- [12]. S. Corson and J. Macker, “Routing Protocol Performance Issues and Evaluation considerations”, RFC2501, IETF Network Working Group, January 1999.
- [13]. S. R. Biradar, Hiren H D Sharma, Kalpana Shrama and Subir Kumar Sarkar, “Performance Comparison of Reactive Routing Protocols of MANETs using Group Mobility Model”, IEEE International Conference on Signal Processing Systems, pages 192-195 2009.
- [14]. C. Perkins, E. Belding-Royer, S. Das, et al, “Ad hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561, July 2003.
- [15]. N. Aschenbruck, E. Gerhards-Padilla, P. Martini, “A Survey on mobility models for Performance analysis in Tactical Mobile networks,” Journal of Telecommunication and Information Technology, Vol.2 pp.54-61, 2008.
- [16]. X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, “A group mobility model for ad hoc wireless networks,” in ACM/IEEE MSWiM, August 1999.

- [17]. <http://www-scf.usc.edu/~fbai/important/>, referred on February 2010
- [18]. <http://nile.usc.edu/important/>, referred on February 2010.

- [19]. Performance Evaluation of Two Reactive Routing Protocols of MANET using Group Mobility Model IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 10, May 2010: 1694-0814