International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 3, March 2015
Available at http://internationaljournalofresearch.org

A Study of Throughput and Delay Comparison of AODV and DSR Routing Protocols

Manikant Sharma[1]; Praveen Tripathi [2]
Department of Computer Science and Engineering Kanpur Institute of Technology, Kanpur
manikant.1983@gmail.com [1], praveentri81@gmail.com[2]

ABSTRACT

A wireless Ad-hoc network consists of wireless nodes communicating without the need for a centralized administration, in which all nodes potentially contribute to the routing process. Mobile wireless ad hoc networks are infrastructureless and often used to operate under unattended mode. So, it is significant in bringing out a comparison of the various routing protocols for better understanding and implementation of them. In this paper, we studied the performance of routing protocols like Adhoc On-Demand Vector routing (AODV) and Dynamic source routing (DSR). And we have studied the various performances metric Packet Delivery Ratio, Routing Overhead, End-to-End Delay, Throughput and Path Optimality.

Key words-
MANET; Wireless networks; routing; AODV; DSR

1. INTRODUCTION

A wireless Ad-hoc network consists of wireless nodes communicating without the need for a centralized administration. A collection of autonomous nodes or terminals that communicate with each other by forming a multihop radio network and maintaining connectivity in a decentralized manner is called an ad hoc network. There is no static infrastructure for the network, such as a server or a base station. The idea of such networking is to support robust and efficient operation in mobile wireless networks by incorporating routing functionality into mobile nodes. Figure.1 shows an example of an ad hoc network, where there are numerous combinations of transmission areas for different nodes. From the source node to the destination node, there can be different paths of connection at a given point of time. But each node usually has a limited area of transmission range as shown in fig 1 by oval circle around each node. A source can only transmit to node B but B can transmit either C or D. It is challenging task to choose the route to establish the connection between source and destination so that they can transmit the data and communicate.

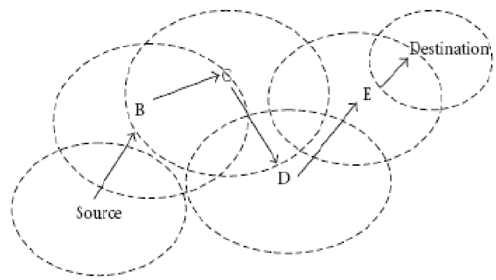

Figure 1: Mobile Ad Hoc Network

Scalable routing is one of the key challenges in designing and operating large scale Mobile Ad hoc NETworks (MANET). In order to ensure effective operation as the total number of nodes in the MANET becomes very large, the overhead of the employed routing algorithms should be low and independent of the total number of nodes in MANET [1]. An important consideration in the development of scalable routing algorithms in large scale MANET is that the overhead properties of the scalable routing formally studied and analyzed. In order for the ad hoc networks to operate as efficiently as possible, appropriate on-demand routing protocols have to be incorporated, to find efficient routes from a source to a destination, taking node mobility into consideration. The Mobility influences ongoing transmissions, since a mobile node that receives and forwards packets may move out of range. As a result, links fail over time. In such cases a new route must be established. Thus, a quick route recovery procedure should be one of the main characteristics of a routing protocol. It is also important to study the various performance metrics for better understanding and utilization of the routing protocols. In this paper, we studied the throughput and delay over AODV and DSR routing protocols.

2. ROUTING PROTOCOLS FOR MANETS

Most widely used routing protocols for wireless ad hoc networks used in Glomosim simulator available till today are Bellman-Ford, AODV, DSR, WRP, ZRP, FISHEYE and LAR1. All these protocols are constantly being improved by IETF. Since these protocols have different characteristics, the comparison of all performance differentials is not always possible. In this study we have considered two routing protocols AODV and DSR.

A. Ad-hoc On Demand Distance Vector (AODV)

AODV [5, 6] shares DSR's on-demand characteristics in that it also discovers routes on an *as needed* basis via a similar route discovery process. However, AODV adopts a very different mechanism to maintain routing information. It uses traditional routing tables, one entry per destination. This is in contrast to DSR, which can maintain multiple route cache entries for each destination. Without source routing, AODV relies on routing table entries to propagate an RREP back to the source and, subsequently, to route data packets to the destination. AODV uses sequence numbers maintained at each destination to determine freshness of routing information and to prevent routing loops [5]. These sequence numbers are carried by all routing packets.

An important feature of AODV is the maintenance of timer-based states in each node, regarding utilization of individual routing table entries. A routing table entry is *expired* if not used recently. A set of predecessor nodes is maintained for each routing table entry, indicating the set of neighboring nodes which use that entry to route data packets. These nodes are notified with RERR packets when the next-hop link breaks. Each predecessor node, in turn, forwards the RERR to its own set of predecessors, thus effectively erasing all routes using the broken link. In contrast to DSR, RERR packets in AODV are intended to inform all sources using a link when a failure occurs. Route error propagation in AODV can be visualized conceptually as a tree whose root is the node at the point of failure and all sources using the failed link as the leaves.

The recent specification of AODV [6] includes an optimization technique to control the RREQ flood in the route discovery process. It uses an *expanding ring search* initially to discover routes to an unknown destination. In the expanding ring search, increasingly larger neighborhoods are searched to find the destination. The search is controlled by the Time-To-Live (TTL) field in the IP header of the RREQ packets. If the route to a previously known destination is needed, the prior hop-wise distance is used to optimize the search. This enables computing the TTL value used in the RREQ packets dynamically, by taking into consideration the temporal locality of routes.

B. Dynamic Source Routing Protocol (DSR)

The key distinguishing feature of DSR [3, 4] is the use of *source routing*. That is, the sender knows the complete hop-by-hop route to the destination. These routes are stored in a *route cache*. The data packets carry the source route in the packet header.

When a node in the ad hoc network attempts to send a data packet to a destination for which it does not already know the route, it uses a *route discovery* process to dynamically determine such a route. Route discovery works by flooding the network with *route request* (RREQ) packets. Each node receiving an RREQ rebroadcasts it, unless it is the destination or it has a route to the destination in its route cache. Such a node replies to the RREQ with a *route reply* (RREP) packet that is routed back to the original source. RREQ and RREP packets are also source routed. The RREQ builds up the path traversed across the network. The RREP routes itself back to the source by traversing this path backward.1 The route carried back by the RREP packet is cached at the source for future use.

If any link on a source route is broken, the source node is notified using a *route error* (RERR) packet. The source removes any route using this link from its cache. A new route discovery process must be initiated by the source if this route is still needed.

DSR makes very aggressive use of source routing and route caching. No special mechanism to detect routing loops is needed. Also, any forwarding node caches the source route in a packet it forwards for possible future use. Several additional optimizations have been proposed and have been evaluated to be very effective by the authors of the protocol [7], as described in the following:

- *Salvaging*: An intermediate node can use an alternate route from its own cache when a data packet meets a failed link on its source route.
- *Gratuitous route repair*: A source node receiving an RERR packet piggybacks the RERR in the following RREQ. This helps clean up the caches of other nodes in the network that may have the failed link in one of the cached source routes.
- *Promiscuous listening*: When a node overhears a packet not addressed to it, it checks whether the packet could be routed via itself to gain a shorter route. If so, the node sends a *gratuitous* RREP to the source of the route with this new, better route. Aside from this, promiscuous listening helps a node to learn different routes without directly participating in the routing process.

C. An Analysis of DSR and AODV

The two on-demand protocols share certain salient characteristics. In particular, they both discover routes only when data packets lack a route to a destination. Route discovery in either protocol is based on query and reply cycles, and route information is stored in all intermediate nodes along the route in the form of route table entries (AODV) or in route caches (DSR). However, there are several important differences in the dynamics of these two protocols, which may give rise to significant performance differentials.

First, by virtue of source routing, DSR has access to a significantly greater amount of routing information than AODV. For example, in DSR, using a single request-reply cycle, the source can learn routes to each intermediate node on the route in addition to the intended destination. Each intermediate node can also learn routes to every other node on the route. Promiscuous listening of data packet transmissions can also give DSR access to a significant amount of routing information. In particular, it can learn routes to every node on the source route of that data packet. In the absence of source routing and promiscuous listening, AODV can gather only a very limited amount of routing information. In particular, route learning is limited only to the source of any routing packets being forwarded. This usually causes AODV to rely on a route discovery flood more often, which may carry significant network overhead.

Second, to make use of route caching aggressively, DSR replies to *all* requests reaching a destination from a single request cycle. Thus, the source learns many alternate routes to the destination, which will be useful in the case that the primary (shortest) route fails. Having access to many alternate routes saves route discovery floods, which is often a performance bottleneck. However, there may be a possibility of a route reply flood. In AODV, on the other hand, the destination replies only once to the request arriving first and ignores the rest. The routing table maintains at most one entry per destination.

Third, the current specification of DSR does not contain any explicit mechanism to expire stale routes in the cache, or prefer “fresher” routes when faced with multiple choices. As noted in [7], stale routes, if used, may start polluting other caches. Some stale entries are indeed deleted by route error packets. But because of promiscuous listening and node mobility, it is possible that more caches are polluted by stale entries than are removed by error packets. In contrast, AODV has a much more conservative approach than DSR. When faced with two choices for routes, the fresher route (based on destination sequence numbers) is always chosen. Also, if a routing table entry is not used recently, the entry is expired. The latter technique is not problem-free, however. It is possible to expire valid routes this way if unused beyond an expiry time. Determination of a suitable expiry time is difficult, because sending rates for sources, as well as node mobility, may differ widely and can change dynamically. In a recent paper [8], the effects of various design choices in caching strategies for on-demand routing protocols are analyzed.

Fourth, the route deletion activity using RERR is also conservative in AODV. By way of a predecessor list, the error packets reach *all* nodes using a failed link on its route to any destination. In DSR, however, a route error simply backtracks the data packet that meets a failed link. Nodes that are not on the upstream route of this data packet but use the failed link are not notified promptly.

The goal of our simulations that follow is to determine the relative merits of the aggressive use of source routing and caching in DSR, and the more conservative routing table and sequence-number-driven approach in AODV.

3. PERFORMANCE METRICES

These metrics are interesting because they can be used to point out what really happened during the simulation and provide valuable information about the routing protocol. In the following sections some metrics of this type are described.

A. Packet delivery ratio

The packet delivery ratio presents the ratio between the number of packets sent from the application layer and the number of packets actually received at the destination nodes. It is desirable that a routing protocol keep this rate at a high level since efficient bandwidth utilization is important in wireless networks where available bandwidth is a limiting factor.

This is an important metric because it reveals the loss rate seen by the transport protocols and also characterizes the completeness and correctness of the routing protocol.

B. Routing overhead

Routing overhead is of course an interesting metric. In some way it reveals how bandwidth efficient the routing protocol is. The routing overhead metric simply shows how much of the bandwidth (which often is one of the limiting factors in a wireless system) that is consumed by the routing messages, i.e., the amount of bandwidth available to the data packets.

An interesting observation is that for all protocols there is a theoretical limit where some properties of the scenario force the data rate down to zero because all the bandwidth is used for routing messages. The ideal case is naturally no overhead at all i.e., only data packets traverse the network. An ideal routing protocol can be implemented in a simulator but a routing protocol without routing messages is a contradiction and cannot be implemented in a real network.

The routing overhead is typically much larger for a proactive protocol since it periodically floods the network with update messages. As mobility in the network increases reactive protocols will of course have to send more routing messages too. This is where the real strength or weaknesses of the routing protocol can be revealed.

In DSR another type of overhead presents itself even though it is easily overlooked in the previously described packet delivery ratio metric. DSR works by finding source routes to the destination on-demand. By storing information about all intermediate nodes in the packet header as the route discovery packet traverses the network it knows the full route once the route discovery packet returns. These source routes cause the packet headers to grow and hence produce more routing overhead. Considering this, the traditional metric, packets sent versus packets delivered, might give the impression that DSR is able to deliver more packets than other protocols. Looking at the ratio payload bytes sent versus payload bytes received instead could result in a different performance for DSR. This would be most obvious in a network with long routes (many hops).

C. End-to-end delay

The term end-to-end is used to an average measure of performance between nodes in a network. It is the sources and the receivers that are involved. The end-to-end delay is therefore the total delay that a data packet experiences as it is traveling through a network. This delay is built up by several smaller delays in the network that adds together. These delays might be time spent in packet queues, forwarding delays, propagation delay (the time it takes for the packet to travel through the medium) and time needed to make retransmissions if a packet got lost etc.

Typically, in a packet based radio network without QoS (Quality of Service) the delay could vary much depending on the routing protocol. One parameter that is critical is the time a packet is kept in a buffer before it is dropped if there is no route for its destination. This buffering time is controlled by a timer in each node. If this timer is set to a high value it could imply that packets are delayed in a network for this rather long period of time. A high value would probably decrease the number of dropped packets but it would also result in a somewhat higher average delay. Of course this is a question of what is important in a particular network, low delay or few dropped packets. It is a tradeoff that the system designer need to do, and as stated earlier, this will have an impact on the end-to-end delay.

D. End-to-end throughput

Since the available bandwidth in a network is fairly well known, it is interesting to see what the actual throughput achieved in a simulation is. If a good estimation of this value can be extracted it would be possible to see how efficient the routing protocol is. The higher the average throughput, the less routing overhead consuming the bandwidth.

E. Path optimality

Traditionally this measurement compares the optimal path usually defined as the shortest path between two nodes in the simulator at the sending moment with the length of the path that the packet actually travelled. If the average actual path length is close to the shortest path, the protocol is said to be good. However, it is hard to know what the actual optimal path is. Just settling with the shortest path does not address queuing and congestion in the network or high latency links.

4. SUMMARY

In this paper we have studied the routing protocols AODV and DSR over various numbers of nodes and various speeds. Here we study five performance metrics like Packet Delivery Ratio, Routing Overhead, End-to-End Delay, Throughput and Path Optimality. And the studied shows that the behavior of routing protocols varies as the no. of nodes, speed of nodes (Nodes Mobility Models) are changed. The performance of routing protocols varies with the above models.

For future work we can implement other routing protocols with the above mobility models and different models (scenario). And we can use different performance metrics.

5. REFERENCES

[1] Manoj Kumar and Ramesh Kumar," Analysis of Delay & Throughput in MANET for DSR Protocol" in World Academy of Science,Engineering and Technology Vol:5 2011-05-28.

[2] A.Boomarani Malany and V.R.Sarma Dhulipala, " Throughput and Delay Comparison of MANET Routing Protocols" in Int. J. Open Problems Compt. Math., Vol. 2, No. 3, September 2009 ISSN 1998-6262; www.i-csrs.org.

[3] Charles E. Perkins and Elizabeth M. Royer, " Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks" in IEEE INFOCOM 2000 conference.

[4] E. Celebi, “Performance evaluation of wireless multi -hop adhoc network routing protocols,” http://cis .poly.edu/ecelebi/esim.pdf.

[5] Kumar Manoj, S.C. Sharma, Sandip Vijay and Amit Dixit “Performances Analysis of Wireless Adhoc Network Using OPNET Simulator “ International Conference on “Intelligent Systems and Networks” ISN-2008 ISTK Haryana, 22-24th 2008, 267-270 Feb. 2008.

[6] GLOMOSIM Simulator, Retrieved 15 June, 2010, [Online], Available: http://www.GloMoSim.com

[7] David B. Johnson. Routing in ad hoc networksof mobile hosts. In Proceedings of the IEEE Workshop on Mobile Computing Systems andApplications, pages 158– 163, December 1994.

[8] David B. Johnson and David A. Maltz. Dynamic source routing in adhoc wireless networks. In Mobile Computing, edited by Tomasz Imielinski and Hank Korth, chapter 5, pages 153– 181. Kluwer Academic Publishers, 1996.

[9] Josh Broch, David B. Johnson, and David A. Maltz. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet-Draft, draft-ietf-manet-dsr-00.txt, March 1998. Work in progress.

[10] A.K. Jain, S.C. Sharma, Kumar Manoj, R. Jha “Effect of Congestion on the Global Performance of IEEE 802.11 Network” International Conference on “Intelligent Systems and Networks” ISN-2008 ISTK Haryana, 22-24th 2008, 231-235 Feb. 2008.