Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

MANET: Empirical Analysis and Performance Evaluation of Routing Protocol Using NS-2

Raj Singh, Mtech, Dinesh Kumar
Dept. of computer sci. & Engineering, Shri Ram College of Engineering and Management, Palwal, affiliated to M.D.U Rohtak, Haryana, India

Abstract- A Mobile Ad-hoc Network (MANET) is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. One of the main issues in such networks is performance- in a dynamically changing topology; the nodes are expected to be power-aware due to the bandwidth constrained network. Another issue in such networks is security - since every node participates in the operation of the network equally, malicious nodes are difficult to detect.. To study these issues, a scenario based simulation analysis of a secure routing protocol is done and is compared with traditional non-secure routing protocols. The scenarios used for the experiments depict critical real-world applications such as battlefield and rescue operations, which tend to have contradicting needs. An analysis of the tradeoffs between performance and security is done to gain an insight into the applicability of the routing protocols by using simulation tool NS-2 which is the main simulator.

Keywords— MANET, OSPF, DSR, AODV, TORA, OLSR.DSDV.

I. INTRODUCTION

Over the past decade, there has been a growing interest in wireless networks, as the cost of mobile devices such as PDAs, laptops, cellular phones, etc have reduced drastically. The latest trend in wireless networks is towards *pervasive and ubiquitous computing* - catering to both nomadic and fixed users, anytime and anywhere. Several standards for wireless networks have emerged in order to address the needs of both industrial and individual users.In such a network, a set of mobile nodes are connected to a fixed wired backbone. WLANs have a short range and are usually deployed in places such universities, companies, cafeterias, etc. However, there is still a need for communication in several scenarios of deployment where it is not feasible to deploy fixed wireless access points due to physical constraints of the medium. For example, consider communication amongst soldiers in a battlefield, involving troops spread out over a large area. In this case, it is not only feasible to deploy a fixed wireless access point, but also risky since an enemy attack would bring down the whole network. This problem has led to a growing interest among the research community in *mobile ad hoc networks*, wireless networks comprised of mobile computing devices communicating without any fixed infrastructure. Initially a classification of wireless networks in use today is described followed by the background and origins of ad hoc wireless networks. The general issues in ad hoc wireless networks are then discussed, followed by a few interesting applications. The final section gives an outline of the chapters to follow.

II. DESIGNING ISSUES IN MANET

The following design issues must be considered before designing a routing protocol for MANETs [1]-

A. Dynamic Topology:

In a MANET, the network topology keeps changing with time due to the movement of the nodes, and hence the links between the nodes suffers frequent breaks. Thus the ordinary routing protocols for wired networks are not efficient since they are designed for static networks.

B. Bandwidth constraint:

The nodes in the network have a relatively low bandwidth when compared to traditional wired networks. This is an important issue to consider when designing routing protocols for MANETs since the utilization of bandwidth by the routing protocol in the network must be minimized.

C. Error prone broadcast channel:

The nodes in the MANET broadcast the information to all the neighboring nodes on the wireless channel. The channel itself is prone to several errors such as attenuation, multi-path fading, etc. Thus the routing protocol itself must be designed taking into consideration these issues.

D. Resource limitations:

As discussed in chapter 1, MANETs consist of nodes such as PDA, laptops, etc. which have stringent power requirements. Further, some of these devices have limited processing power. Thus the routing protocols must be efficient in terms of power conservation.

***E. QoS limitations*:**
For applications such as multimedia, QoS guarantees must be provided by the routing protocol. However, such guarantees come at the cost of higher latency and poor performance since multimedia applications require higher bandwidth and traffic rates.

***F. Hidden and exposed terminal Problems*:**
The hidden terminal problem is shown in Figure 2.1.

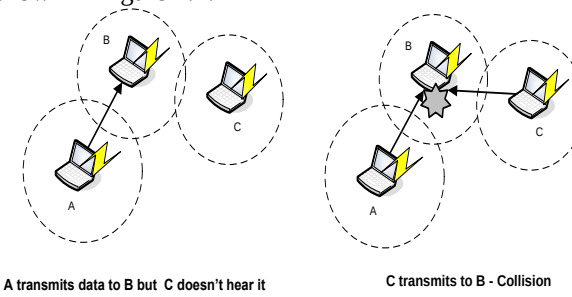

Figure 2.1: The Hidden Terminal Problem

This problem occurs in networks using contention based protocols such as ALOHA, CSMA/CD, etc. When two nodes which are out of range of each other send data frames to a node which is within their respective radio ranges, a collision of data frames occurs. As shown in Figure.2.1, when both nodes A and C transmit data frames to node B a collision occurs. This problem can be resolved by using a mechanism called RTS/CTS handshake [2]. The exposed node problem is shown in Figure.2.2. An exposed node is one which is in the range of the transmitter, but out of the range of the receiver. In Figure.2.2, when node C is transmitting to node D, B overhears this and is blocked. Now if node B wants to transmit to node A, it cannot do so.

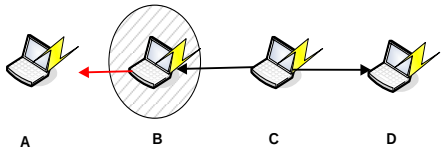

Figure 2.2: The Exposed Terminal Problem

This results in wasted bandwidth. The hidden and exposed terminal problems occur at the MAC layer and prevent successful transmission of data packets. This, in turn also affects the design of the routing protocol. In order to prevent this, the routing protocol must reduce the number of broadcast packets to minimize collisions.

***G. Security*:**
Due to on open environment where MANETs are typically deployed, the routing protocols are prone to several attacks. Further, there is also the issue of secure key distribution. This issue will be further explored further when secure routing is discussed in Chapter-4.

III. CLASSIFICATION OF THE ROUTING PROTOCOLS

Several routing protocols have been proposed for ad hoc networks. In this section a broad classification of these routing protocols is given. Only the unicast routing protocols are considered and an in-depth classification of all available protocols is beyond the scope of this thesis. Figure 2.3 shows the classification of the routing protocols for MANETs. At one end are the table-driven or proactive routing protocols such as the Destination Sequenced Distance Vector (DSDV) routing protocol, Wireless Routing Protocol (WRP), etc. At the other end, are the on-demand or reactive protocols such as Dynamic Source Routing (DSR) protocol and the Ad hoc On-demand Distance Vector (AODV) routing protocols. Each of these types of protocols is discussed in more detail.

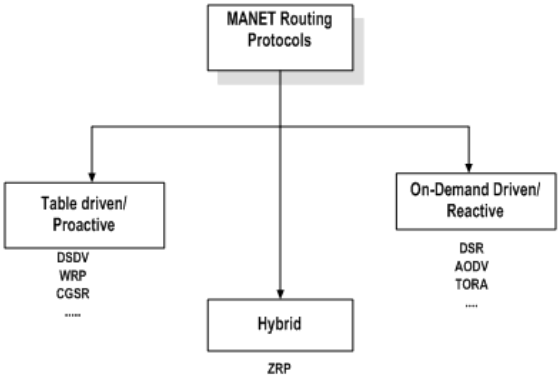

Fig3.1 : Classification of MANET routing protocols

A. Table-driven/Proactive Routing Protocols:

In table-driven or proactive protocols, the nodes maintain an active list of routes to every other node in the network in a routing table. The tables are periodically updated by broadcasting information to other nodes in the network. Thus, they are an extension to the wired network routing protocols such as the Routing Internet Protocol (RIP). Any node wishing to communicate with another node has to obtain the next hop neighbor on the route to the destination from its routing table. Some examples of table-driven routing protocols are Destination Sequenced Distance-Vector routing protocol (DSDV) [3], Wireless Routing Protocol (WRP) [4], Cluster Switch Gateway Routing protocol (CGSR) [5], etc. In the following sections, working of DSDV and WRP are explained, and the general pros and cons of table-driven routing protocols are enumerated.

1. Destination Sequenced Distance Vector (DSDV) Routing Protocol: The Destination Sequenced Distance Vector (DSDV) protocol is a proactive routing protocol based upon the distributed Bellman Ford algorithm [6]. In this routing protocol, each mobile host maintains a table consisting of the next-hop neighbor and the distance to the destination in terms of number of hops. It uses *sequence numbers* for the destination nodes to determine "freshness" of a particular route, in order to avoid any short or long-lived routing loops. If two routes have the same sequence number, the one with smaller distance metric is advertised. The sequence number is incremented upon every update sent by the host. All the hosts periodically broadcast their tables to their neighboring nodes in order to maintain an updated view of the network. The tables can be updated in two ways – either *incrementally* or through a *full dump*. An incremental update is done when the node doesn't observe any major changes in the network topology. A full dump is done when network topology changes significantly or when an incremental update requires more than one NPDU (Network Packet Data Unit).

Table: 3.1: Routing table for node 1

Destination	Next hop	Metric	Sequence number
1	-	0	S40_1
2	2	1	S340_2
3	3	1	S22_3
4	4	1	S334_4
5	2	2	S76_5
6	3	2	S84_6
7	4	2	S98_7

Let us consider an example to understand the routing mechanism better. Consider the network topology shown in figure 2.4. The routing table for this network is shown in table 2.1. As shown in the table, each node maintains a route to every other node in the network during the route establishment phase. Whenever there is a link break in the network, the end node of the broken link propagates a routing table update message with the broken link's weight assigned to infinity. This message is broadcasted by every node to its neighbors. A broken link is denoted by an odd sequence number and an ordinary link by an even sequence number. When node 1 wants to send data to node 7, it checks the next hop neighbor for node 7, which is 2 and passes the data packet to it.

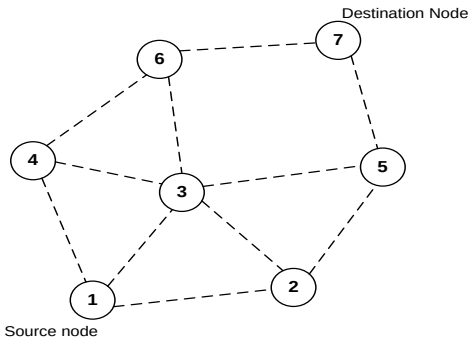

Fig 3.2 Topology graph of the network

Table 3.2: Modified routing table for node 1

Destination	Next hop	Metric	Sequence number
1	-	0	S40_1
2	2	1	S340_2

3	3	1	S22_3
4	4	1	S334_4
5	2	2	S76_5
6	3	2	S84_6
7	2	3	S94_7

Figure 2.5 shows the case when node 7 moves out of range of nodes 6 and 5. Thus the link 6-7 and 7-5 are broken and the routing table at 1 is now reorganized as shown in Table 2.2. When node 4 hears the update request from node 7 with a higher sequence number, it broadcasts this information to all nodes. This eventually reaches node 1 which changes the next hop, metric and the sequence number entry in routing table for 7.

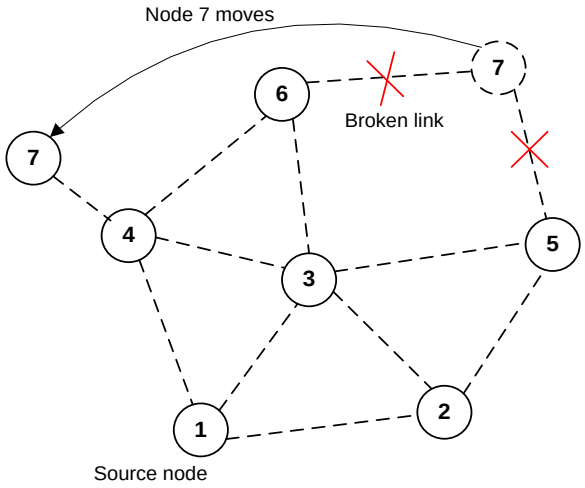

Fig 3.3 : Topology graph of the network when node 7 moves

DSDV guarantees loop free routes to each destination and also finds the optimal path. It uses an *average settling delay* to prevent frequent routing table updates and any fluctuations caused by two similar routing advertisements which are in an incorrect order of the sequence numbers.

2. Wireless Routing Protocol (WRP) The Wireless Routing protocol (WRP) [4] is a table-driven protocol based upon the distributed Bellman Ford algorithm and is similar to DSDV. The difference between DSDV and WRP is the number of tables maintained at each node. In WRP, the following tables maintained at each node-

a). Routing table (RT): It is used for maintaining an up-to-date view of the network for all the destinations. It consists of the destination node, the predecessor node (penultimate node), the successor node (the next hop neighbor) and a flag to indicate status of the path.

b). Link Cost Table (LCT): This table stores the cost (no. of hops to reach the destination) of relaying messages through each link. The cost of a broken link is taken as infinity. It also stores the number of update periods passed since the last successful update was received from that link. This is done to detect link breaks.

c). Distance Table (DT): This table stores the number of hops between a node and its destination.

d). Message Retransmission List (MRL): The MRL contains an entry for every update message retransmitted and has a counter for each entry which is retransmitted after the message is sent. Other fields are an acknowledgement flag and a list of messages in each update.

Every node periodically sends an update message to its neighbors, which contains a list of updates and a list of responses indicating which node must acknowledge the update. When a node detects a link break, it sends an update message to its neighbors with the link cost of the broken link set to infinity. All the nodes which had an active route to the nodes affected by the link break then update their corresponding entries to them. By storing the predecessor node information and forcing every node to check if the information is correct, WRP avoids the *count to infinity* problem .

B. On-Demand/Reactive Routing Protocols

In contrast to table driven routing protocols, on-demand routing protocols find route to a destination only when it is required. The on-demand protocols have two phases in common – route discovery and route maintenance. In the route discovery procedure, a node wishing to communicate with another node initiates a discovery mechanism if it doesn't have the route already in its cache. The destination node replies with a valid route. The route maintenance phase involves checking for broken links in the network and updating the routing tables. The working of a few reactive routing protocols is now described.

1. Dynamic Source Routing (DSR) Protocol

The Dynamic Source Routing Protocol [8] is an on-demand routing protocol which is based on the concept of *source routing*. In source routing, a sender node specifies in the packet header, the complete list of nodes that the packet must traverse to reach the destination node. This essentially means that every node just needs to forward the packet to its next hop specified in the header and need not check its routing table as in table-driven routing protocols. Furthermore, the nodes don't have to periodically broadcast their routing tables to neighboring nodes. The DSR protocol works in two phases as described below-

a) Route Discovery

In the route discovery phase, the source node establishes a route by broadcasting route request (RREQ) packets to all its neighbors. Each neighboring node, in turn rebroadcasts the packets to its neighbors if it has not already done so, or if it is not the destination node, provided that the TTL (Time To Live) counter is greater than zero. Further, *request ids* are used to determine if a particular route request has been previously received by the node. Each node maintains a list of recently received <*initiator*, *request id*> pairs. If two route requests with the same <*initiator*, *request id*> are received by a forwarding node, it broadcasts only one of them and drops the other. This also prevents formation of routing loops in the network. When the packet reaches the destination node, it unicasts a reply packet (RREP) on the reverse path back to sender. This reply packet contains the route to that destination. Figure 2.7 shows an example of the route discovery mechanism. When node 1 wants to communicate with node 7, it initiates a route discovery mechanism and broadcasts request packet RREQ to its neighboring nodes 2, 3 and 4 as shown. However, node 3 also receives the broadcast packets from nodes 4 and 2 with the same <*initiator*, *request id*> *pair*. It drops both of them and broadcasts the other packet to its neighbors. The other nodes follow the same procedure. When the packet reaches node 7, it inserts its own address and reverses the route in the record and unicasts it back on the reverse path to the destination.The destination node unicasts the best route (received first) and caches the other routes for future. A *route cache* is maintained at every node so that, whenever a node receives a route request and finds a route for the destination node in its own cache, it sends a RREP packet itself without broadcasting it further.

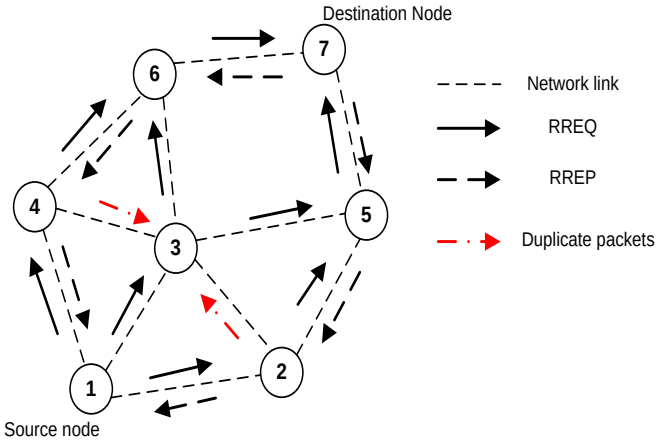

Fig 3.4 : Route Discovery in DSR

b) Route Maintenance:

The route maintenance phase is carried out whenever there is a broken link between two nodes. A failed link can be detected by a node by either passively monitoring in promiscuous mode or actively monitoring the link. As shown in Figure 2.8, when an intermediate node in the path moves away, causing a wireless link to break (6-7), a route error packet (RERR) is sent by the intermediate node back to the originating node. The source node re-initiates the route discovery procedure to find a new route to the destination. It also removes any route entries it may have in its cache to the destination node.

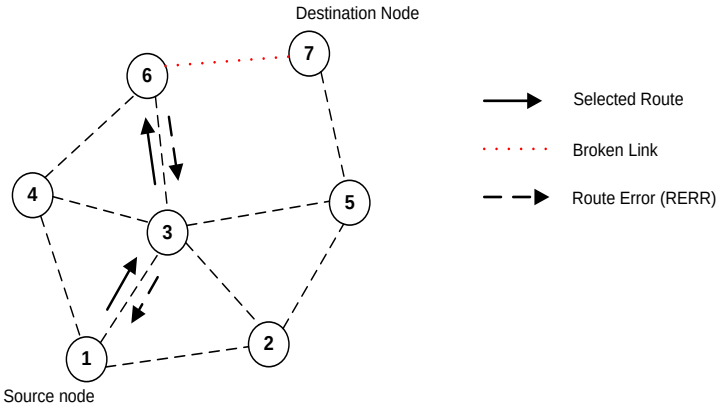

Fig 3.5: Route Maintenance in DSR

DSR benefits from source routing since the intermediate nodes need not maintain up-to-date routing information in order to route the packets that they forward. There is also no need for any periodic routing advertisement messages. However, as size of the network increases, the routing overhead increases since each packet has to carry the entire route to the destination with it. The use of route caches is a good mechanism to reduce the propagation delay but overuse of the cache may result in poor performance [8]. The disadvantage of DSR is that whenever there is a link break, the RERR packet propagates to the original source, which in turn initiates a new route discovery process. Thus the link is not repaired locally. Several Optimizations to DSR are possible such as *non- propagating route requests* (when sending RREQ, nodes set the hop limit to one preventing them from re-broadcasting), *gratuitous route replies* (when a node over hears a packet with its own address listed in the header, it sends a RREP to the originating node bypassing the preceding hops), etc. A detailed explanation of DSR optimizations can be found in [8].

2. Ad hoc On-demand Distance Vector (AODV) Routing Protocol

The Ad hoc On-demand Distance Vector routing protocol [9] inherits the good features of both DSDV and DSR. The AODV routing protocol uses a reactive approach to finding routes and a proactive approach for identifying the most recent path. More specifically, it finds routes using the route discovery process similar to DSR and uses destination sequence numbers to compute fresh routes. The two phases are discussed in more detail-

a) Route Discovery

During the route discovery process, the source node broadcasts RREQ packets similar to DSR. The RREQ packet contains the source identifier (SId), the destination identifier (DId), the source sequence number (SSeq), the destination sequence number (DSeq), the broadcast identifier (BId) and TTL fields. When an intermediate node receives a RREQ packet, it either forwards it or prepares a Route Reply (RREP) packet if it has a valid route to the destination in its cache. The (SId, BId) pair is used to determine if a particular RREQ has already been received in order to eliminate duplicates. Every intermediate node enters the previous node's address and its BId while forwarding a RREQ packet. The node also maintains a timer associated with every entry in order to delete a RREQ packet if the reply is not received before it expires. Whenever a RREP packet is received by a node, it stores the information of the previous node in order to forward the packet to it as the next hop towards the destination. This acts as a "forward pointer" to the destination node. Thus each node maintains only the next hop information unlike source routing in which all the intermediate nodes on the route towards the destination are stored.

Figure 2.9 shows an example of route discovery mechanism in AODV. Let us suppose that node 1 wants to send a data packet to node 7 but it doesn't have a route in its cache. Then it initiates a route discovery process by broadcasting a RREQ packet to all its neighboring nodes.

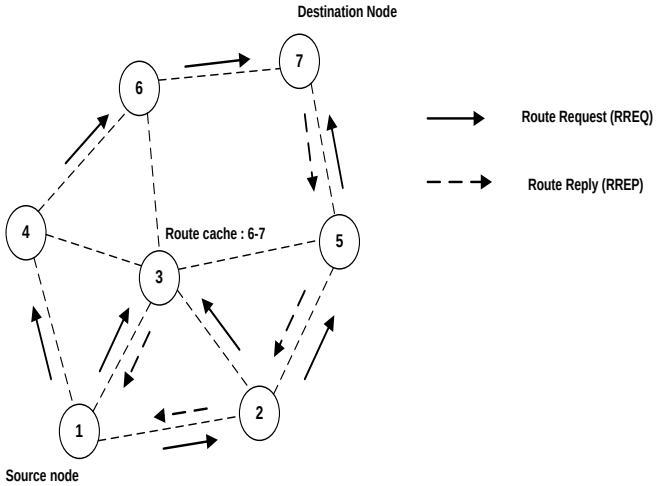

Fig 3.6: Route discovery in AODV

It inserts the SId, DId, SSeq, DSeq, BId, and TTL fields in the RREQ packet. When nodes 4, 3 and 2 receive this, they check their route caches to see if they already have a route. If they don't have a route, they forward it to their neighbors, else the destination sequence number DSeq in the RREQ packet is compared with the DSeq in its corresponding entry in route cache. If the DSeq in RREQ packet is greater, then it replies to the source node with a RREP packet containing the route to the destination. In figure 2.9, node 3 has a route to 7 in its cache and its DSeq is higher compared to that in RREQ packet. So, it sends a RREP back to the source node 1. Thus the path 1-3-6-7 is stored in node 1. The destination node also sends a RREP back to the source. For example, one possible route is 1-2-5-7. The intermediate nodes on the path from source to destination update their routing tables with the latest DSeq in the RREP packet.

b) Route Maintenance

The route maintenance mechanism works as follows – Whenever a node detects a link break by link layer acknowledgements or HELLO beacons [2], the source and end nodes are notified by propagating an RERR packet similar to DSR. This is shown in Figure 2.10. If the link between nodes 3 and 5 breaks on the path 1-3-5-7, then both 5 and 3 will send RERR packets to notify the source and destination nodes.One optimization possible in AODV route

maintenance is to use an expanding ring search to control the flood of RREQ and discover routes to unknown destinations [10]. The main advantage of AODV is that it avoids source routing thereby reducing the routing overload in large networks. Further, it also provides destination sequence numbers which allows the nodes to have more up-to-date routes. However, AODV requires bidirectional links and periodic link layer acknowledgements to detect broken links. Further, it has to maintain routing tables for route maintenance unlike DSR.

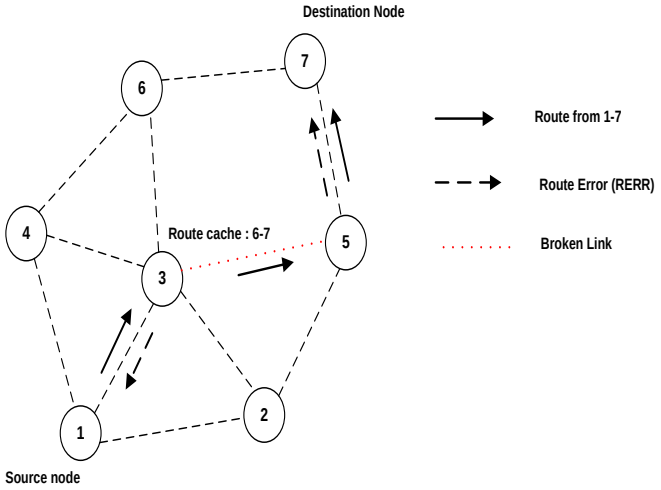

Fig 3.7: Route Maintenance in AODV

Comparison of DSR and AODV

Table 3.3 provides a comparison of the features of DSR and AODV:

Protocol / Feature	DSR	AODV
Destination sequence numbers	Not used	Used
Link Layer acknowledgements	Not Required	Required (using HELLO beacons) for link breakage detection
Routing mechanism	Source routing – Multiple route caches for each destination	Table driven – one entry per destination. Sequence numbers used for
Route storage mechanism	Using route caches	Using routing tables
Timers	Not Used	Used
Multiple Route caches	Yes	No
Optimizations	Salvaging, Gratuitous route replies (RREP) and Route Error (RERR), non-propagating route requests [11]	Expanding ring search [10]

The main difference is the source routing employed by DSR in contrast to table-driven routing used by AODV. Due to this, DSR has a higher routing load when the size of the network increases since each packet header has typically more information when compared to AODV. Another important difference is that AODV requires link layer acknowledgements or HELLO beacons at periodic intervals in order to detect link breaks. However, DSR avoids this feature and hence more efficient. Further, DSR stores multiple route caches for a destination whereas AODV does not. It has been found that this has an impact on the end-to-end delay and the delivery fraction as the size of the network increases [10]. DSR has been found to perform well in lightly loaded networks, whereas AODV performs well in more stressful networks (with higher density of nodes). AODV also benefits from its timer mechanisms by maintaining fresher route entries as compared to DSR, which doesn't implement any timers. Besides, in DSR all requests reaching a destination node are replied to, whereas in AODV the destination replies only once to the request arriving first and ignores others

C. Hybrid Routing Protocols

Hybrid routing protocols inherit the characteristics of both on-demand and table-driven routing protocols. Such protocols are designed to minimize the control overhead of both proactive and reactive routing protocols. The working of hybrid routing protocols is illustrated with an example – the Zone Routing Protocol (ZRP).

1. Zone Routing Protocol (ZRP)

The Zone Routing Protocol [12] is a hybrid protocol which combines the best features of both reactive and proactive

routing protocols. The protocol itself consists of four components: (i) the Intra Zone Routing Protocol (IARP) (ii) the reactive Inter zone Routing Protocol (IERP), and (iii) Border cast Resolution Protocol (BRP). The working principle of ZRP is as follows.

The whole network is effectively divided into *zones*, where each zone represents a small part of the network. It also specifies a zone radius which represents the maximum number of hops to reach the farthest node in the zone. Within a zone, the routing is done by a table-driven mechanism using the IARP protocol. A node can belong to more than one zone. Figure 2.11 depicts the concept of routing zone and zone radius. The nodes within a zone exchange periodic route updates. Between zones, the communication occurs using the IERP, in which a node wishing to communicate with a node in a different zone sends a route request packet to all nodes on the border of the zone. For example, in Figure 2.11 if node 2 wishes to communicate with node 7, it will send request packets to nodes 1, 3 and 5.

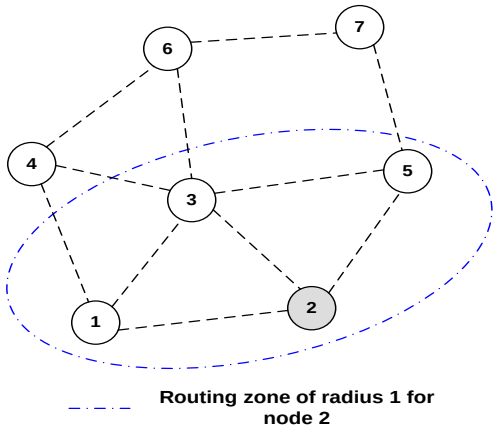

Fig 3.8: Routing zone and zone radius

IV. SIMULATION ANALYSIS OF MANET PROTOCOL

In this simulation we check behavior of a network using DSR, AODV, TORA protocols in scenario with 150 node, many network simulators are available to design and simulate networks in many perspectives. NS-2 (Network Simulators-2) and OPNET (Optimized Network Engineering Tools) are the two very well-known

Experimental Setup and Metrics

The ns-2 simulator was used for the experiments. We now describe the traffic pattern, the scenario description and the metrics that were used for the experiments.

(i) The traffic pattern

The traffic pattern file was generated using the "cbrgen.tcl" script (explained in section 4.2.2). The parameters used were as follows –

Table 4.1: Traffic pattern

Type of traffic	Constant Bit Rate
Packet Size	512 bytes
Packet Rate	4 pkts/sec
Maximum number of connections	20

(ii) Scenario description

The scenario was generated using the BonnMotion software. BonnMotion is a Java-based software which creates and analyses mobility scenarios. It generates the movements of nodes in an ad hoc network as a trace file which can be imported into ns-2 (explained in the tutorial provided in appendix-B). It has support for several mobility models such as the RPGM, Random Waypoint model, Gauss Markov mobility model, etc. The following metrics were used to depict a battlefield scenario.

Table 4.2: Parameters for the battlefield scenario

Dimensions	2000*2000
Mobility Model	Reference Point Group Mobility Model (RPGM)
No. of nodes	50
Min. speed	1 m/s
Max. speed	5 m/s

Average number of nodes in a group	10
Probability of group change	0.01
Pause time	60 sec

(iii) Metrics:

The following metrics were used for performance evaluation-

a. *Packet Delivery Fraction (PDF)*: This is the ratio of total number of packets successfully received by the destination nodes to the number of packets sent by the source nodes throughout the simulation.

$$PDF = \frac{numberOf\,\mathrm{Re}ceivedPackets}{numberOfSentPackets}$$

This estimate gives us an idea of how successful the protocol is in delivering packets to the application layer. A high value of PDF indicates that most of the packets are being delivered to the higher layers and is a good indicator of the protocol performance.

b. *Normalized Routing Load (NRL)*: This is calculated as the ratio between the no. of routing packets transmitted to the number of packets actually received (thus accounting for any dropped packets).

$$NRL = \frac{numberOfRoutingPacketsSent}{numberOfDataPackets\,\mathrm{Re}ceived}$$

This metric gives an estimate of how efficient a routing protocol is since the number of routing packets sent per data packet gives an idea of how well the protocol maintains the routing information updated. Higher the NRL, higher the overhead of routing packets and consequently lower the efficiency of the protocol.

c. *Average end to end delay (AED)* : This is defined as the average delay in transmission of a packet between two nodes and is calculated as follows-

$$AED = \frac{\sum_{i=0}^{n}(timePacket\,\mathrm{Re}ceived_{\mathrm{i}} - timePacketSent_{i})}{totalNumberOfPackets\,\mathrm{Re}ceived}$$

A higher value of end-to-end delay means that the network is congested and hence the routing protocol doesn't perform well. The upper bound on the values of end-to-end delay is determined by the application. For example multimedia traffic such as audio and video cannot tolerate very high values of end-to-end delay when compared to FTP traffic.

(iv) Research methodology

Three parameters in the battlefield scenario were varied - pause time, the total number of nodes and average number of nodes in a group and their impact on the three metrics described above were studied. The results are discussed in the next section.

Results

i. Effect of varying the number of nodes

The number of nodes was varied from 50 to 100 and the effect on PDF, NRL and AED was studied. The results can be found in table 4.3 and figures 4.3, 4.4 and 4.5.

Table 4.3: Effect of varying the number of nodes

No. Of Nodes	**Packet Delivery Fraction (%)**	**Average End-end delay (sec)**	**Normalized Routing Load**
50	99.91438	0.006738278	0.2570694
60	100	0.006566893	0.3088803
70	100	0.013576984	0.42168674
80	99.95756	0.032688957	0.47558385
90	99.95761	0.010179137	0.49618322
100	99.872444	0.010737591	0.553427

It is found that the packet delivery fraction decreases as the number of nodes in the network increases. This is due to the fact that as number of nodes increases, the congestion in the network also increases and hence the number of lost packets due to retransmission also increases. Further, since AODV uses a table driven approach, the processing delay at the nodes also increases with an increase in the size of the network thereby accounting for the higher end-to-end delay. The normalized routing load increases with an increase in number of nodes due to an increase in the routing packets in the network.

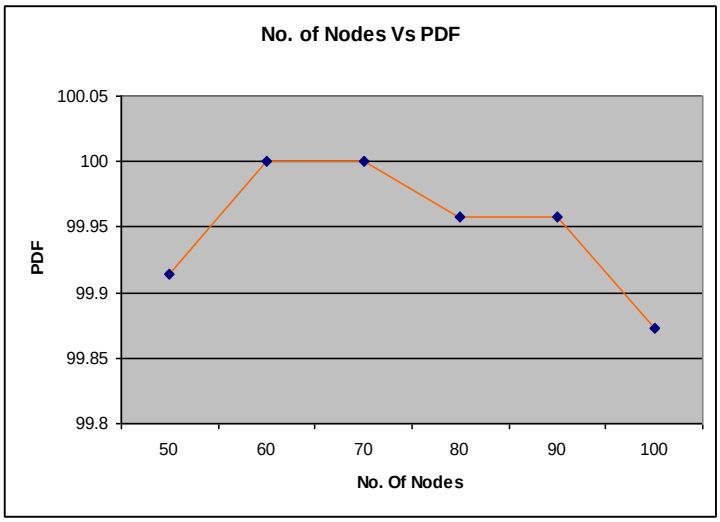

Fig 4.3: Effect of varying the number of nodes on the pause time

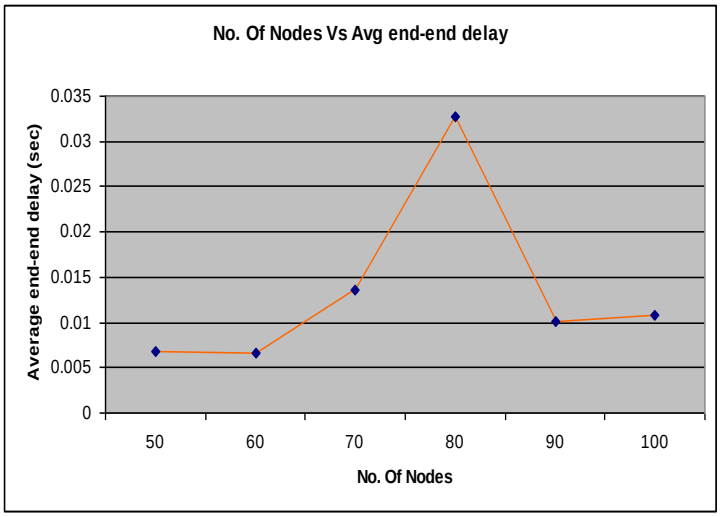

Fig 4.4: Effect of varying the number of nodes on the Average end-end delay

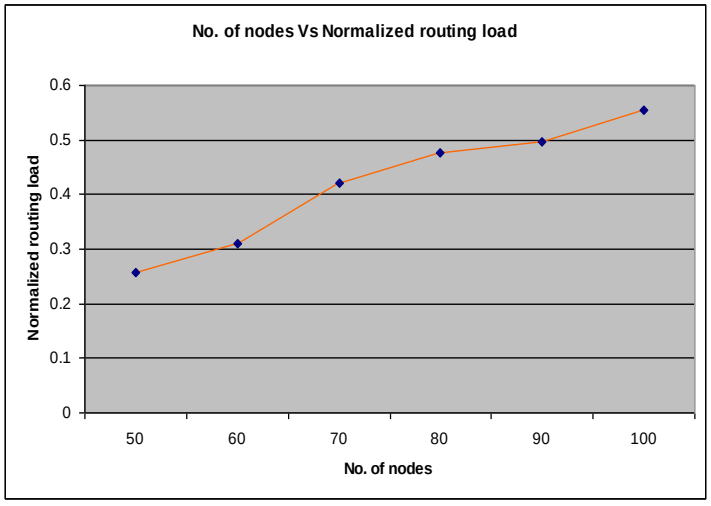

Fig 4.5: Effect of varying the number of nodes on the Normalized Routing Load

The blue circles in figures 4.3, 4.4 and 4.5 represent the "optimal points" which corresponds to the highest PDF, lowest end-to-end delay and the lowest normalized routing load. It is found that for 60 nodes we achieve this optimal point.

ii. Effect of varying the pause time

The effect of varying the pause time on the three metrics are shown in table 4.4 and the corresponding graphs are shown

in figures 4.6, 4.7 and 4.8. It can be inferred that as pause time varies, the packet delivery fraction also increases. This is due to the fact that as pause time increases, the relative mobility of the nodes decreases, and hence the congestion also decreases in the network.

Table 4.4: Effect of varying the pause time

Pause Time (sec)	**Packet Delivery Fraction (%)**	**Average End-to-end delay (sec)**	**Normalized routing load**
10	99.87218	0.006634372	0.25597268
20	99.957466	0.006683255	0.25531915
30	99.91536	0.006524965	0.25412962
40	100	0.010312819	0.27754056
50	100	0.010314601	0.2742616
60	99.91438	0.006738278	0.2570694

The end-to-end delay also decreases as the pause time is increased. This can be explained as follows – as the pause time increases, the network topology is relatively stable and hence the number of stale routes in the routing tables decreases. Thus route discovery and maintenance take less time. This also reduces the number of routing packets in the network, thereby decreasing the NRL.

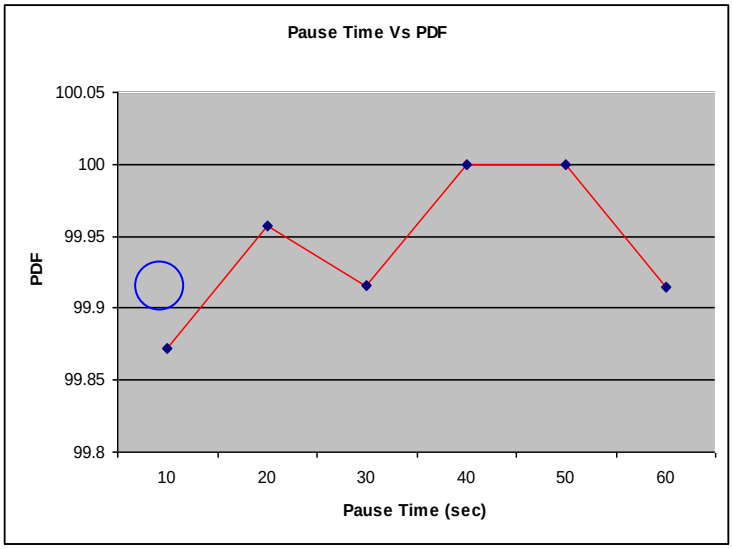

Fig 4.6: Effect of varying the pause time on PDF

Fig 4.7: Effect of varying the pause time on average end to end delay

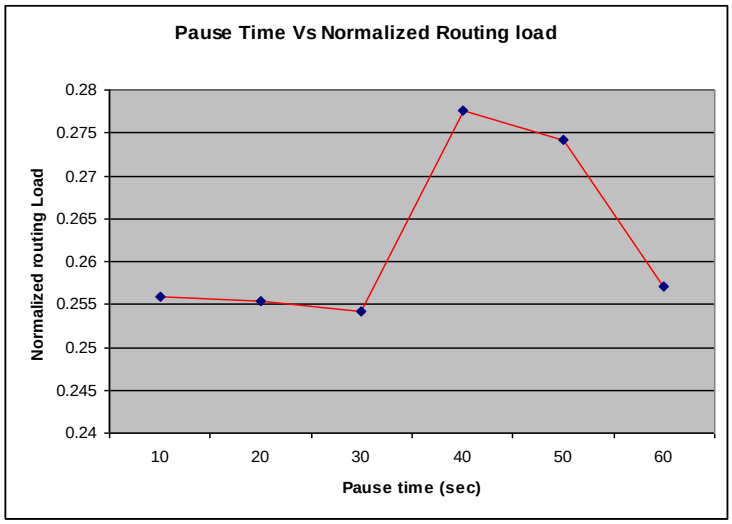

Fig 4.8: Effect of varying the pause time on NRL

From figures 4.6, 4.7 and 4.8 it can be inferred that for a pause time of 20 sec (represented by a blue circle), we obtain optimal values for the three metrics.

V. CONCLUSION

For the battlefield scenario, AODV has found to perform well for lower pause times (20 sec), higher density of nodes (9 per group) and smaller networks. As the network size increases, the performance drops due to a table-driven approach. However, since it does not use source routing, it has a much lower end to end delay for In order to analyze the performance of routing protocols in practice, such a scenario-based approach is vital. It also helps identify the suitable routing protocol for an optimal network size, the mobility of the nodes, the network density and a given traffic pattern.A more comprehensive study of other routing protocols such as DSR, TORA, DSDV, etc. is needed to choose the right protocol for a given scenario.

ACKNOWLEDGMENT

It is a great pleasure for me to express my sincere gratitude to my supervisor, **Dr. Dinesh Kumar,** Professor /Assistant Professor, Department of Computer Science & engineering of Shri Ram College of Engineering & Management, for his valuable guidance, timely advice and constant encouragement during the project work.

REFERENCES

[1] C. Siva Ram Murthy, B.S. Manoj, "*Ad Hoc Wireless Networks : Architectures and Protocols*", Prentice Hall Publishers, May 2004, ISBN 013147023X

[2] C.-K. Toh, "*Ad Hoc Mobile Wireless Networks: Protocols and Systems*", Prentice Hall publishers, December 2001, ISBN 0130078174

[3] C. Perkins and P. Bhagwat, *Highly Dynamic Destination-Sequenced Distance-Vector Routing* (*DSDV*) for Mobile Computers. In Proc. of the ACM SIGCOMM, October 1994. http://www.cs.umass.edu/~mcorner/courses/691M/papers/perkins.pdf

[4] Shree Murthy, J.J. Garcia-Luna-Aveces, "*A Routing Protocol for Packet Radio Networks*," Proc. ACM International Conference on Mobile Computing and Networking, pp. 86-95, November, 1995 http://www.pdos.lcs.mit.edu/decouto/papers/dube97.pdf

[5] C.-C. Chiang, "*Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel*," Proc. IEEE SICON '97, Apr. 1997, pp. 197–211. http://www.ics.uci.edu/~atm/adhoc/paper-collection/gerla-routing-clustered-sicon97.pdf

[6] [online] The Secan Lab, University of Luxembourg, Luxembourg.http://wiki.uni.lu/secan-ab/Distributed+Bellman-Ford.html

[7] [online] The Secan Lab, University of Luxembourg, Luxembourg.http://wiki.uni.lu/secan-lab/Count-To-Infinity+Problem.html

[8] D B. Johnson, D A. Maltz, and Y. Hu, "*The dynamic source routing protocol for mobile ad hoc network*," Internet-Draft,April-2003. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-09.txt

[9] C.E. Perkins, E. Royer, and S.R. Das, "*Ad hoc on demand distance vector (AODV) routing*," Internet Draft, March2000. http://www.ietf.org/internetdrafts /draft-ietf-manet-aodv-05.txt

[10] Samir R. Das, Charles E. Perkins, Elizabeth M. Royer and Mahesh K. Marina. *"Performance Comparison of Two On-demand Routing Protocols for Ad hoc Networks."* IEEE Personal Communications Magazine special issue on Ad hoc Networking,February-2001,p.16-28. http://www.ronai.hu/././library/Performance_comparison_of_AODV_and_DSR-Perkins.pdf

[11] David B Johnson and David A Maltz. "*Dynamic source routing in ad hoc wireless networks"*. In Imielinski and Korth, editors, Mobile Computing, volume 353. Kluwer Academic Publishers, 1996.

[12] Haas Z.J, " *A new routing protocol for the reconfigurable wireless network"*. In Proceedings of the 1997 IEEE 6th International Conference on Universal Personal Communications, ICUPC '97, San Diego, CA, October 1997; pp.562--566.http://www.ics.uci.edu/~atm/adhoc/paper-collection/haas-routing-protocol-icupc97.ps.gz

[13] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields and Elizabeth M. Belding royer. *"A Secure Routing Protocol for Ad Hoc Networks" (ARAN)* In International Conference on Network Protocols (ICNP), Paris, France,November,2002. www.cs.ucsb.edu/~kimaya/icnp2002.pdf

[14] Stefano Basagni, Marco Conti, Silvia Giordano, Ivan Stojmenovic, "*Mobile Ad Hoc Networking*", ISBN: 0-471-37313-3, Wiley-IEEE Press: *Chapter 12: Ad hoc networks Security* Pietro Michiardi, Refik Molva http://www.eurecom.fr/~michiard/pub/michiardi-adhoc.pdf

[15] Hongmei Deng, Wei Li, and Dharma P. Agrawal, "*Routing Security in Wireless Ad Hoc Network*," IEEE Communications Magzine, vol. 40, no. 10, October 2002.

[16] Yih-Chun Hu, David B. Johnson, Adrian Perrig. *"SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks"*, Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '02), pp: 3-13,Jun2002. http://www.cs.colorado.edu/~rhan/CSCI_7143_001_Fall_2002/Papers/Perrig2002_wmcsa02.pdf

[17] Yih-Chun Hu, Adrian Perrig, David B. Johnson. *"Ariadne: A secure On-Demand Routing Protocol for Ad hoc Networks"* MobiCom 2002, September 23-28, 2002, Atlanta, Georgia,USA. http://lambda.cs.yale.edu/cs425/doc/ariadne.pdf

[18] A. Perrig, R. Canetti, D. Tygar, and D. Song, "*The TESLA Broadcast Authentication Protocol*," Cryptobytes,, Volume 5, No. 2 (RSA Laboratories, Summer/Fall 2002), pp. 2-13.http://www.rsasecurity.com/rsalabs/cryptobytes/

[19] P. Papadimitratos and Z. Haas. *"Secure routing for mobile ad hoc networks" (SRP)* SCS Communication Networks and Distributed Systems Modeling and Simulation Conference, pp. 27--31, January 2002. http://wnl.ece.cornell.edu/Publications/cnds02.pdf

