Journal of Mobile Communication 2 (1): 18-26, 2008
ISSN: 1990-794X

Performance Comparison of MANET Protocols Based on Manhattan Grid Mobility Model

G. Jayakumar and G. Gopinath
Department of Computer Science, Bharathidasan University, Tiruchirapalli-620 023, India

Abstract: Mobile ad hoc network is a collection of mobile nodes communicating through wireless channels without any existing network infrastructure or centralized administration. Because of the limited transmission range of wireless network interfaces, multiple "hops" may be needed to exchange data across the network. In order to facilitate communication within the network, a routing protocol is used to discover routes between nodes. The primary goal of such an ad hoc network routing protocol is to find an efficient route establishment between a pair of nodes so that messages may be delivered in a timely manner. Route construction should be done with a minimum of overhead and bandwidth consumption. This study examines two on demand routing protocols for mobile ad hoc networks based on manhattan grid mobility model-the Dynamic Source Routing (DSR), Ad hoc On-demand Distance Vector routing (AODV) and evaluates both protocols based on packet delivery fraction, normalized routing load, average delay and normalized medium access load by varying number of nodes, speed and connection rate.

Key words: MANET, routing protocol, on-demand, DSR, AODV, Manhattan

INTRODUCTION

Mobile ad hoc networks are formed by autonomous system of mobile nodes connected by wireless links without any preexisting communication infrastructure or centralized administration. Communication is directly between nodes or through intermediate nodes acting as routers. The advantages of such a network are rapid deployment, robustness, flexibility and inherent support for mobility. Due to their Minimal configuration, quick deployment and absence of a central governing authority make Ad Hoc networks suitable for emergency situations like natural disasters, military conflicts, emergency medical situations etc (Corson and Macker, 1999; Carlo, 1999). Many previous studies have used Random Waypoint as reference model (Guolong *et al*., 2004; Tracy *et al*., 2002). In this study we will make a detailed study based on Manhattan mobility model. Due to node mobility, routes between two nodes may change. Therefore, it is not possible to establish fixed paths for delivery between networks. Because of this, routing is the most studied problem in mobile ad hoc networks and a number of routing protocols have been proposed.

The ad-hoc routing protocols are divided into 2 categories.

Table-driven routing protocols: In table driven routing protocols, consistent and up-to-date routing information to all nodes is maintained at each node.

On-demand routing protocols: In On-Demand routing protocols, the routes are created as and when required. When a source wants to send to a destination, it invokes the route discovery m echanisms to find the path to the destination. Once a Route has been established, it is maintained until either the destination becomes inaccessible (along every path from the source), or until the route is no longer used, or expired (Elizabeth, 2003).

In recent years, a variety of new routing protocols targeted specifically at this environment have been developed. We consider two wireless on demand ad hoc network routing protocols that cover a range of design choices.

DYNAMIC SOURCE ROUTING PROTOCOL (DSR)

DSR is a uniform, topology based, reactive protocol. It emphasizes aggressive caching and deduction of topology information extracted from source routing headers. When using source routing, each packet to be routed carries in its header the complete list of nodes through which the packet must pass. An advantage of source routing is that intermediate hops need not maintain routing information in order to route the packets they receive, since the packet they carry the necessary routing information. DSR adapts quickly to routing changes when host movement is frequent, it requires less or no overhead during periods when the host moves less frequently. This protocol performs well in environmental

Corresponding Author: G. Jayakumar, Department of Computer Science, Bharathidasan University, Tiruchirapalli-620 023, India

conditions such as host density and movement rates. DSR (Johnson and Maltz, 1996) is an on-demand routing protocol, which is based on the concept of source routing. Mobile nodes should maintain route caches, which specify the source route, which the mobile should be aware. The entries in the route caches must be continually updated as new routes are learned. This protocol is of 2 phases: Route maintenance and Route discovery.

When a source node has a packet to be send to the destination it first checks the route caches whether it has a route to the destination. If it has an unexpired route to the destination then it uses this route to send the packet. If there is no route then the source node initiates a Route discovery by broadcasting a route request packet. This request packet contains the address of the source, the address of the destination and a unique identification number. When a node receives the packet it checks whether it has a route to the destination. If not it adds its own address to the route record of the packet and forward the packets to the outgoing links. To restrict the number of route requests being forwarded nodes forward the route request packet if the mobile has not seen the request and if the mobile address does not appear in the route record.

A route reply is generated when it reaches the destination. When the packet reaches the destination the route record specifies the number of hops it has taken. When a node generating route reply is a destination, it places the route record contained in the route request into route reply (Royer and Toh, 1999). If the responding node is an intermediate node then it appends its cached route-to-route record and generates the route reply. To return the route reply the responding node must have a route to the initiator. If it has a route in the route cache then it uses that route. If symmetric links are provided the node reverse the route in route record. If symmetric links are not supported the node initiates its own route discovery and piggyback the route reply on the new route request.

Rout maintenance (David *et al.*, 2004) is made through the use of route error packets and acknowledgements. Route error packets are transmitted at a node when the data link layer encounters a fatal transmission problem. When a route error packet is received the hop in error is removed from the nodes, route cache and nodes containing that hop are truncated at that point. Acknowledgements are used to verify the correct operation of links. Such acknowledgments are passive acknowledgement where the mobile is able to hear the next hop forwarding the packet along the route. Route maintenance can be performed using end-to-end acknowledgement rather than hop-by-hop acknowledgement. As long as route exists between two end-to-end hosts which can communicate route maintenance is possible. With hop-by-hop acknowledgement, the particular hop error can be indicated in the route error packet but with end to end acknowledgement the sender assumes that error has occurred in the last hop of the route to the destination is in error.

Route optimizations: Based on the study of Sung and Gerla (1999) on the simulation study of table driven and on-demand routing protocols for mobile networks, explains the following optimization schemes, to improve performance and reduce overhead can be made in DSR. Some of the optimization are:

- Non propagating route request.
- Piggybacking on route discoveries.
- Gratuitous route replies.
- Gratuitous route errors.
- Salvaging.
- Snooping.

Non-propagating route request: When the sender originates a request, it sets the time to limit to zero hop, thus allowing the neighbor only to receive the packet. If a neighbor is the destination or has route information to the destination in its cache, it sends a reply to the originator. If no reply is received within the time out period, then the sender floods a route request.

Piggybacking on route discoveries: To eliminate the route acquisition latency, data can be piggybacked on route request packets. If the intermediate node, which has route information to the destination in its cache, replies a route then that node has to construct a data packet and forward to the destination node in order not to lose any data.

Gratuitous route replies: When receiving a packet not address to itself, then the node refers to the listed source route that has not been traversed yet. If unprocessed part contains identification of the node, it realizes that shorter route can be achieved by not visiting the preceding hops in the source route.

Gratuitous route errors: When the source of the broken route receives route error, it piggybacks the received route error on the next route request packet for route discovery. This piggybacking prevents nodes from replying with stale routes.

Salvaging: If an intermediate node of a route detects that next hop cannot be reached; it searches the route cache for an alternate route. If such a route is found, it substitutes the available route for the stale route in the data header and forwards it. The intermediate node is still required to send a route error back to the sender.

Snooping: When processing data, a node examines the unvisited node in the source route and inserts those routes into its route cache. This snooping enables nodes to have multiple alternate routes for each destination.

Two sources of bandwidth overhead in DSR are route discovery and route maintenance. These occur when new routes are to be discovered, or when network topology changes. Caching techniques in each node, at the expense of memory and resources, can reduce this overhead. The remaining source of bandwidth overhead is source route header included in every packet.

Advantages: Dynamic source routing protocol performs well in environmental conditions such as host density and movement rates. DSR can adapt itself to routing changes when host movement is frequent It requires less or no overhead during periods when host moves less frequently. DSR does not require periodic transmission of router advertisements or link status packets, thereby reducing the overhead of DSR.

AD-HOC ON DEMAND DISTANCE VECTOR ROUTING (AODV)

AODV (Perkins and Royer, 1999) initiates routing activities only in the presence of data packets in need of route. AODV is a uniform, destination based, reactive protocol. It uses table driven routing framework and destination sequence numbers into an on demand protocol. AODV is an improvement of DSDV and it minimizes the number of required broadcasts, by creating routes on demand basis, as opposed to maintain a complete lists of routes in DSDV algorithm. AODV can be classified as pure on demand acquisition system. The nodes, which are not selected in the path, do not maintain any routing information or do not take part in routing table exchanges. AODV prepares loop-free routes for both unicast and multicast even while repairing loop broken links. AODV is capable of unicast, broadcast and multicast communication. Unicast and multicast routes are discovered on demand and use a broadcast route discovery mechanism. There are numerous advantages of combining both unicast and multicast communication ability in the same protocol. A protocol, which specifies that route information can be obtained when searching a multicast route, can also increase unicast routing knowledge.

AODV primary objectives: To provide unicast, broadcast and multicast capabilities to all nodes in ad-hoc network; To minimize the broadcast of control packets; To disseminate information about link breakages to neighboring nodes that utilizes the link.

When a source node has to send some message to the destination and if there is not any valid path then it initiates a path discovery process to locate the other node. It broadcasts a route request packet to its neighbors, which then forwards the requests to its neighbors and so on until it reaches the destination or any intermediate node with a fresh enough route to the destination node is located. AODV uses destination sequence numbers to ensure that all routes are loop free and contains the most recent information. Each node maintains its sequence number as well as broadcast id, whenever the node initiates a route request the broadcast id is incremented and together with nodes IP address identifies a route request. Along with its own sequence number and broad casts id the source node includes in the route request the most recent sequence number for the destination. Intermediate nodes do reply only if they have a route to the destination, whose corresponding destination numbers has equal to or greater than that contained in the route request.

While forwarding the route request the intermediate nodes record in the route table the address of the neighbor from which the copy of the broad cast is received, thereby establishing a reverse path. If additional copies of same route request are received then these packets are discarded. Once the route request reaches the destination the destination responds by sending a route reply packet back to the neighbor from where it received the route request. As the route reply is routed back along the reverse path the nodes in this path set forward route entries in their route tables which point to the node from where the route reply had come. These forward entries indicate active forward route. Along with the route entry a timer is also set which causes deletion of the entry if it is not used within the specified lifetime. Route reply is forwarded along the path established by the route request. AODV supports only symmetric links.

Route maintenance: If a source node moves it is able to reinitiate a route discovery protocol to find a new route to the destination. If anode along the route moves its upstream neighbor notices it and propagates a link failure notification message to each of the active stream neighbors to inform them the erasure of the part of the route. These nodes propagate the link failure notification to their upstream neighbors until the source node is reached. The source then reinitiates a route discovery for that destination if a route is still is desired.

Hello messages are broad casted by a node to inform each other neighborhood and they are used to maintain the local connectivity of the node. When a retransmission of packet has to be made the nodes listen to ensure that next hop is within the reach. For these purposes hello messages are used to ensure that next hop is within the communication range. The hello messages are used to specify greater knowledge of network connectivity.

Advantages: There are numerous advantages of combining both unicast and multicast communication ability in the same protocol. A protocol, which specifies that route information can be obtained when searching a multicast route, can also increase unicast routing knowledge. If a node returns a route for a multicast group to some source node, that source node in addition to learning how to reach a multicast group will also have learned a route to the node returning the information. AODV has the advantage of enhanced general routing knowledge. Combining both the protocol simplifies coding.

Disadvantages: AODV does not provide guaranteed delivery of data packets. To enhance this service multicasting algorithm can be used known as Scalable Reliable Multicast (SRM) (Floyd *et al*., 1997).

MANHATTAN MOBILITY MODEL

We introduce the Manhattan model to emulate the movement pattern of mobile nodes on streets defined by maps. The Manhattan map used in our study is shown in Fig. 1.

Applications: It can be useful in modeling movement in an urban area where a pervasive computing service between portable devices is provided.

Important characteristics: Maps are used in this model too. The map is composed of a number of horizontal and vertical streets. Each street has 2 lanes for each direction (north and south direction for vertical streets, east and west for horizontal streets). The mobile node is allowed to move along the grid of horizontal and vertical streets on the map. At an intersection of a horizontal and a vertical street, the mobile node can turn left, right or go straight. This choice is probabilistic: the probability of moving on the same street is 0.5, the probability of turning left is 0.25 and the probability of turning right is 0.25. The velocity of a mobile node at a time slot is dependent on its velocity at the previous time slot. Also, a nodes velocity is restricted by the velocity of the node preceding it on the same lane of the street. The inter-node and intra-node relationships involved are the same as in the Freeway model. Thus, the Manhattan mobility model is also expected to have high spatial

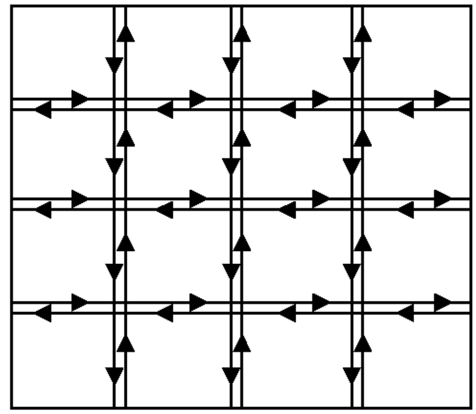

Fig. 1: Manhattan map

dependence and high temporal dependence. It too imposes geographic restrictions on node mobility (http:// nile.usc.edu/ important/software.htm). However, it differs from the Freeway model in giving a node some freedom to change its direction. Most of the mobility models mentioned above are parameterized. E.g. SDR and ADR (Biao *et al*., 1994) are some of the parameters used in RPGM, while maps are important parameters in the Freeway and Manhattan models. Using a parameterized approach, we aim to get a good coverage of the design space of the proposed mobility metrics by producing a rich set of mobility patterns that can be used as a ''test-suite'' for further research.

This study present an overview of 2 main categories of mobile ad-hoc routing protocols and a general comparison of both the protocols based on Manhattan Ggrid model. We present a detailed simulation of 2 routing protocols focussing on their differences in their dynamic behaviours that can lead to performance differences. In the Manhattan Grid Model a bigger map area size is considered because we need to divide the map areas into multiple grids (10 by10) for a 1000×1000 m area. In the first set of experiments the mean of the speed is varied from 5-25 m s^{-1}. The figure show the performance comparison of two On Demand Routing protocols.

SIMULATION MODEL

In this study for the Traffic and Mobility Models generation Constant Bit Rate (CBR) traffic sources are used. The source-destination pairs are spread randomly over the network. Only 512-byte data packets are used. The number of source-destination pairs and the packet sending rate in each pair is varied to change the offered load in the network.

GENERATING TRAFFIC AND MOBILITY MODELS

Traffic models: Random traffic connections of TCP and CBR can be setup between mobile nodes using a traffic-scenario generator script. This traffic generator script is available under ~ns/indep-utils/cmu-scen-gen and is called cbrgen.tcl. It can be used to create CBR and TCP traffics connections between wireless mobile nodes. So the command line looks like the following:

ns cbrgen.tcl [-type cbr|tcp] [-nn nodes] [-seed seed] [-mc connections][-rate rate]

For the simulations carried out, traffic models were generated for 20 nodes with cbr traffic sources, with maximum connections of 5 at a rate of 10 packets sec^{-1} and the packet size is 512 bytes.

Mobility models: The Bonn motion tool which is a Mobility Scenario Generation and Analysis Tool (Christain and Michael, 2007) is used for generating different movement patterns.

Performance metrics: The following 4 important performance metrics are considered for evaluation of these two on demand routing protocols:

Packet delivery fraction: The ratio of the data packets delivered to the destinations to those generated by the CBR sources.

Average end-to-end delay of data packets: This includes all possible delays caused by buffering during route discovery latency, queuing at the interface queue, retransmission delays at the MAC and propagation and transfer times.

Normalized routing load: The number of routing packets transmitted per data packet delivered at the destination. Each hop-wise transmission of a routing packet is counted as one transmission

Normalized MAC load: The number of routing, Address Resolution Protocol (ARP) and control,e.g., Request To Send (RTS), Clear To Send (CTS), Acknowledge (ACK) packets transmitted by the Medium access Control (MAC) layer for each delivered data packet. Essentially, it considers both routing overhead and the MAC control overhead. Like normalized routing load, this metric also accounts for transmission at every hop.

The first two metrics are the most important for best effort traffic. The routing load metric evaluates the efficiency of the routing protocol. Finally the MAC load is a measure of effective utilization of the wireless medium by data traffic.

Implementation: For implementation we install Cygwin in windows and this can be done from the website www.cygwin.com and the installation of the ns-allinone-2.28 that is the Ns-2 network simulator (ns-allinone-2.28) was installed from the website www.isi.edu/nsnam/ns/.

It is found that installation of ns-2 could be a bit lengthy and a time-consuming process. It involved downloading and setting up a 250 MB package. However, getting the simulator to work was the first step involved in carrying out the simulations.

The wireless extension to ns-2 (incorporated in the current release ns-2.28 under cygwin) provides the implementation of the DSR, AODV routing protocols. Nam is the basic visualization tool used for ns-2 simulations. Using the perl script analyze the trace file which are generated and using MSExcel we draw the graph.

To Parsing the Simulation trace files what we do is after each simulation, the trace files recording the traffic and node movements are generated. These files need to be parsed in order to extract the information needed to measure the performance metrics. The old trace format was used for parsing (k-lug.org/~griswold/NS2/ns-trace-formats.html).

RESULTS AND DISCUSSION

The simulation parameters considered for doing the performance comparison of two routing protocols is given in Table 1.

The analysis of the Fig. 2a shows that AODV has less delivery ratio when compared to DSR. In the Manhattan Grid models, nodes do not move diagonally instead they travel either forward or backwards, right or left and all directions are orthogonal to each other. The chance of a traffic signal breaking up increases as 2 nodes diverges. Hence, there are more route error messages due to disconnections. From the Fig. 2b we infer that routing

Table 1: The performance comparison of two routing protocols

Protocols	AODV, DSR
Simulation time	200s
#of nodes	20
Map size	1000×1000 m
Mean Speed	5-10-15-20-25 m s^{-1}
Mobility model	Manhattan Grid
Traffic type	CBR
Packet size	512 bytes
Connection rate	10 pkts sec^{-1}
Pause time	10 s
#of connections	5

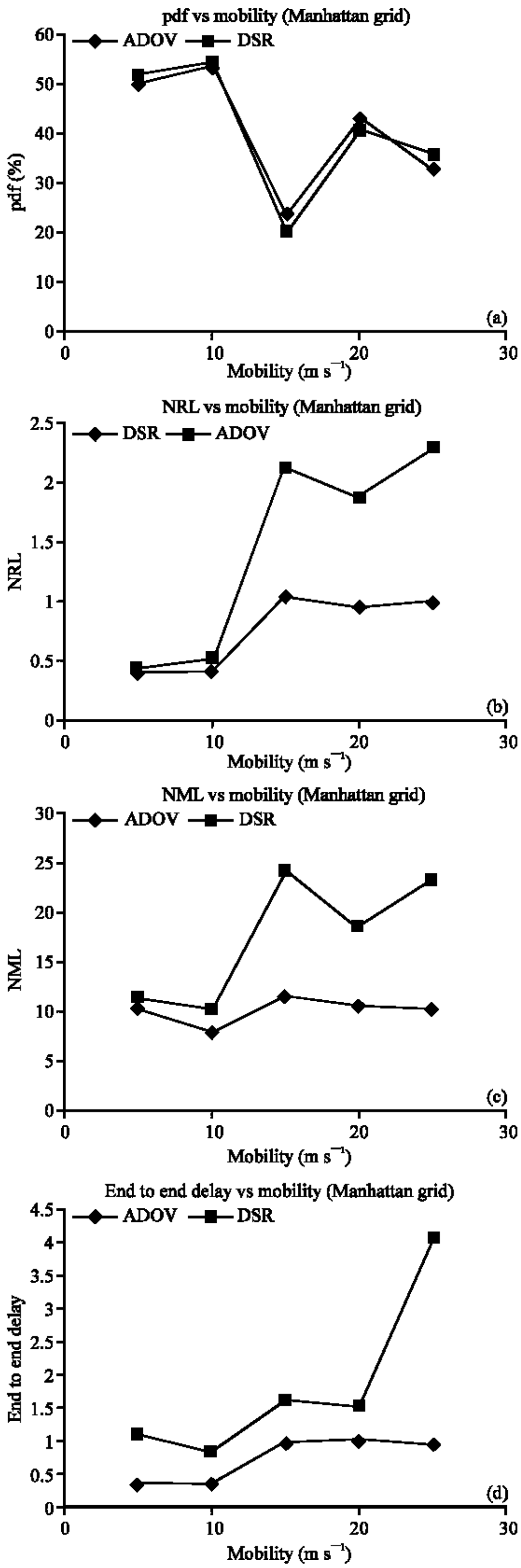

Fig. 2: Mobility results for Manhatten grid models

overheads generated by AODV is greater than DSR. This can be attributed due to more routing updates needed in AODV. DSR uses source routing and also caches some routing entries. The normalized MAC load for AODV is found to be less when compared to DSR. This is because Rout Error (RERRs) (Fig. 2c) is handled different in each protocol. RERR are unicast in DSR and therefore contribute to additional MAC overhead like Route Reply (RREPs). In AODV, RERRs are broadcast like Route Request (RREQs) and hence are less expensive. Consequently when the MAC overhead is factored DSR is found to generate higher overall network load than AODV in all scenarios despite having less routing overhead. The end to end delay of AODV is less when compared to DSR (Fig. 2d).

Node density results: In the second experiment on Manhattan Grid model, we try to find the performance aspect when the node density is varied within a fixed map area. The node density is made to increase from 10-50 nodes within the same map area of 1000×1000 m. The simulation parameters are shown in the Table 2.

From the Fig. 3a we infer that DSR actually starts degrading the PDF performance on approaching a higher node density. The routing overhead remains relatively low for DSR whereas for AODV it increases tremendously. The network delay of DSR when compared with the rest has a significant order of magnitude difference of almost 5-6 times more. This is highly undesirable for delay sensitive applications. Hence, for scenarios similar to Manhattan Grid, it would be advisable not to use DSR for routing delay sensitive applications. We see from the figure the mobility of the nodes affects the number of

Table 2: Simulation parameters

Protocols	AODV, DSR
Simulation time	200 s
#of nodes	10-20-30-40-50
Map size	1000×1000 m
Speed	10 m s^{-1}
Mobility model	Manhattan grid
Traffic type	CBR
Packet size	512 bytes
Connection rate	20 pkts sec^{-1}
Pause time	10 s
#of connections	5

Table 3: Simulation parameters for networking loading results

Protocols	AODV, DSR
Simulation time	200s
#of nodes	20
Map Size	1000×1000 m
Speed	10 m s^{-1}
Mobility model	Manhattan grid
Traffic type	CBR
Packet size	512 bytes
Connection rate	10-20-30-40-50 pkts sec^{-1}
Pause time	10s
3of connections	5

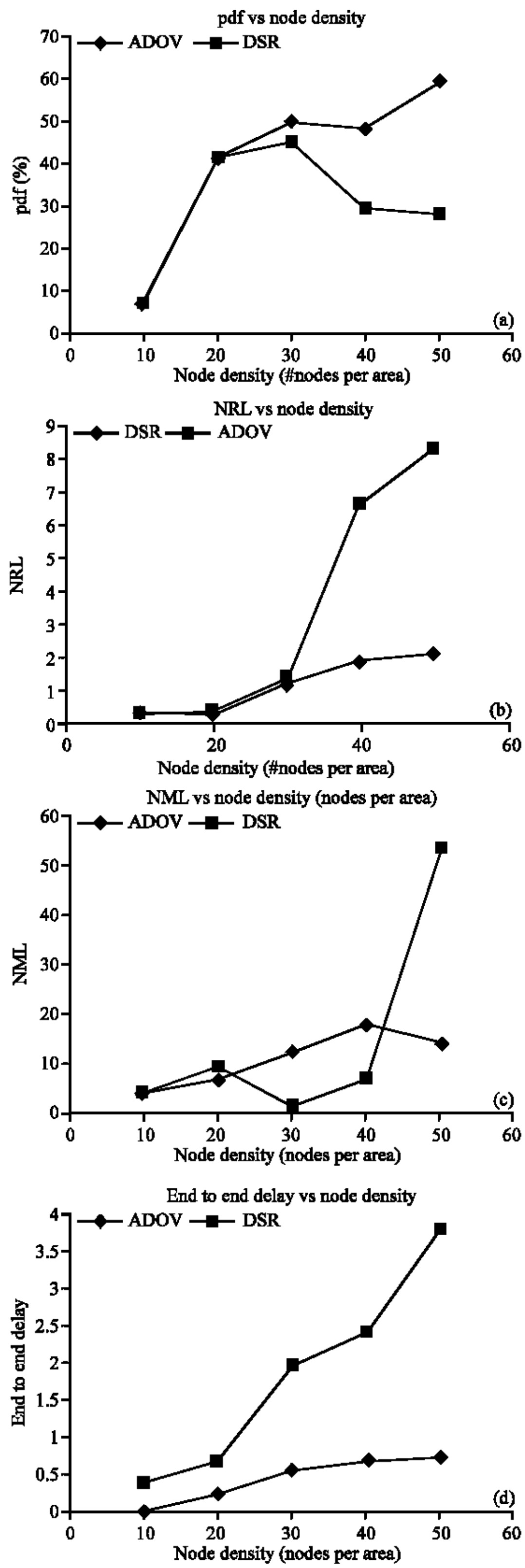

Fig. 3: Node density results

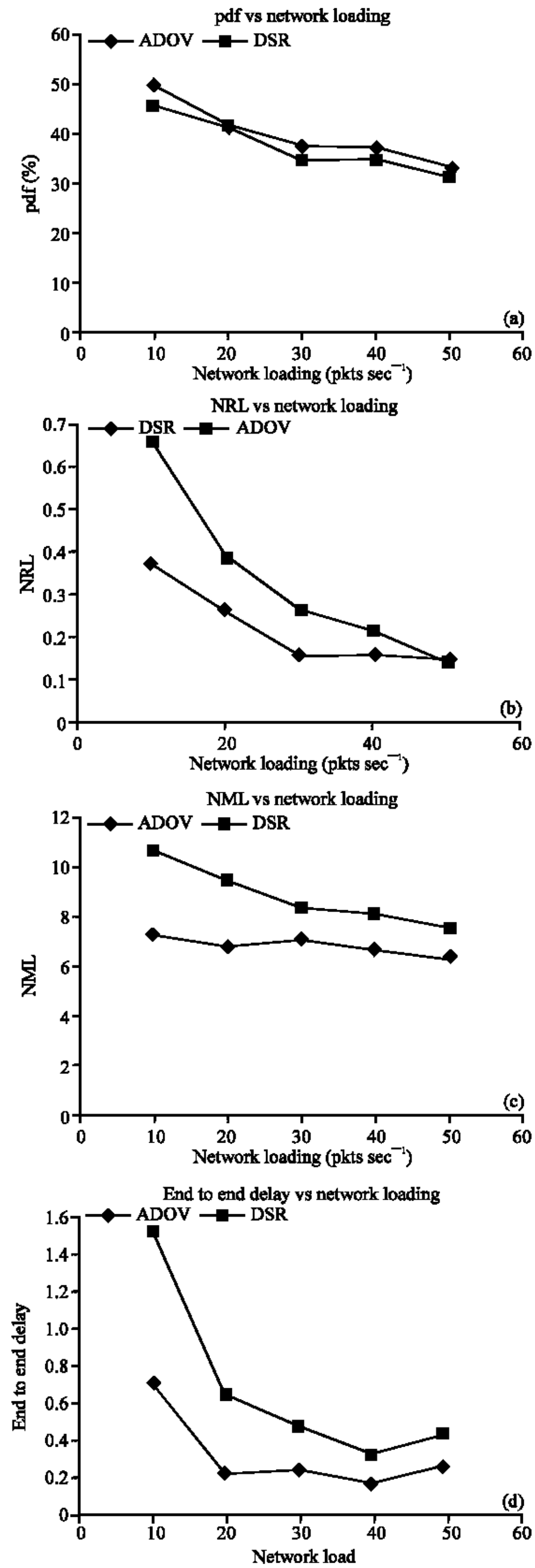

Fig. 4: Networking loa results

average connected paths, which in turn affect the performance of the routing algorithm. We have also studied the impact of node density on routing performance. With very sparsely populated network the number of possible connection between any two nodes is very less and hence the performance is poor (Fig. 3 b-c). It is expected that if the node density is increased the throughput of the network shall increase, but beyond a certain level if density is increased the performance degrades in some protocol (Bai and Helmy, 2004; Bai *et al*., 2004, 2003).

Networking loading results: In the third set of experiments of Manhattan Grid Model we try to find the routing performance aspect when the offered load increases. To do this we increase the average connection load offered by each connection starting at 10-50 pkts sec^{-1}. This study of increase in network loading helps to find the impact of bandwidth intensive applications such as video streaming etc (Table 3).

The Fig. 4a shows that the PDF of routing protocols have very close performance results when the network loading is high. The figure are shown for the parameters which have been simulated. The Fig. 4b shows that DSR has a less routing overhead when compared to AODV. The Fig. 4c shows that DSR has a high normalized MAC load when compared to AODV. The Fig. 4d shows that DSR has a high delay when compared to AODV.

CONCLUSION

In this study, we have analyzed the impact of mobility pattern on routing performance of mobile ad hoc network in a systematic manner. In our study, we observe that the mobility pattern does influence the performance of MANET routing protocols.

One striking difference in Manhattan Grid model simulation is for varying speed of nodes is that AODV protocol has the worst performance for packet delivery ratio despite being having the best results in Random way point model. In densely networks of the Manhattan Grid model the packet delivery ratio for AODV, DSR are relatively near to one another. DSR, however has very bad results in network latency making AODV favorable choices in more dense networks. There is seemingly no difference in the choice of routing protocols for systems having increasing network load when simulated in a Manhattan Grid model. All protocols have almost similar packet delivery ratio performance.

We observe a very clear trend between mobility metric, connectivity and performance. With similar average spatial dependency we find that when relative Speed increases the Link Duration decreases and the Routing Overhead increases and throughput decreases with similar average relative speed we find that Spatial Dependence increase, the Link Duration increases and the Throughput increases and routing overhead decreases. Thus we see that Mobility Metrics influence Connectivity Metrics which in turn influence protocol performance metrics.

Thus, we conclude that relative rankings of protocols may vary with the mobility model used. In mobility models with very high relative speed like Manhattan mobility model AODV seems to achieve as good a throughput as DSR (and sometimes better). AODV does not use aggressive caching, thus the ratio of the number of route replies coming from the cache to the total number of route replies is lesser for AODV than DSR. Thus, the likelihood of getting invalid routes from the cache is lesser for AODV than for DSR. This may explain why AODV outperforms DSR in Manhattan models with high mobility.

REFERENCES

Bai, F. and A. Helmy, 2004. A Survey of Mobility Modeling and Analysis in Wireless Adhoc Networks. In Wireless Ad Hoc and Sensor Networks, Kluwer Academic Publishers.

Bai, F. and A. Helmy, 2004. The IMPORTANT Framework for Analyzing and Modeling the Impact of Mobility in Wireless Adhoc Networks. In Wireless Ad Hoc and Sensor Networks, Kluwer Academic Publishers.

Bai, F., G. Bhaskara and A. Helmy, 2004. Building the Blocks of Protocol Design and Analysis-Challenges and Lessons Learned from Case Studies on Mobile Adhoc Routing and Micro-Mobility Protocols. ACM Comp. Commun. Rev., 34: 57-70.

Bai, F., N. Sadagopan and A. Helmy, 2003. Important: A framework to systematically analyze the Impact of Mobility on Performance of Routing protocols for Adhoc Networks. IEEE INFOCOM, pp: 825-835.

Biao, Z., K. Xu and M. Gerla, 1994. Group and Swarm Mobility Models for Ad Hoc Network Scenarios Using Virtual Tracks. In Proc. MILCOM., 1: 289-294.

Carlo, K., 1999. Ad Hoc Networking, Systems J., pp: 33-40.

Christain de Waal and M. Gerharz, 2007. BonnMotion: A Mobility Scenario Generation and Analysis Tool. Communication Systems Group, Institute of computer Science IV, University of Bonn, Germany. http://web.informatik.uni-bonn.de/IV/Mitarbeiter/BonnMotion/.

Corson, S. and J. Macker, 1999. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC: 2501.

David, B.J., D.A. Maltz and Y.C. Hu, 2004. The Dynamic Source Routing (DSR) Protocol for Mobile Ad Hoc Networks. draft-ietf-manet-dsr-10.txt.

Elizabeth, B.R., 2003. Routing Approaches In Mobile Ad Hoc Networks. In: S. Basagni, M. Conti, S. Giordano, I. Stojemenvoic (Eds.). Ad Hoc Networking. IEEE Press Wiley, New York.

Floyd, S., V. Jacobson, C.G. Liu, S. McCanne and L. Zhang, 1997. A Reliable Multicast Frame work for Light-weight Sessions and Application Level Framing. IEEE/ACM Transaction on Networking, 5: 784-803.

Guolong, L., G. Noubir and R. Rajaraman, 2004. Mobility Models for Ad hoc Network Simulation. In Proceedings of IEEE. INFOCOM., 1: 7-11.

Johnson, D.B. and D.A. Maltz, 1999. Dynamic Source Routing. In Ad-Hoc Wireless Networks, Mobile Computing, T. Imielinski and H. Korth (Eds.). Kluwer, pp: 153-181.

Old and New Trace Format for NS2 Logged Trace Files, 2007. http://k-lug.org/~griswold/NS2/ns-trace-formats.html.

Perkins, C.E. and E.M. Royer, 1999. Ad-hoc on-Demand Distance Vector Routing. Proceeding 2nd IEEE Workshop, Mobile Computing. Sys. Apps., pp: 90-100.

Royer, E.M. and C.K. Toh, 1999. A Review of Current Protocols for Ad Hoc Mobile Networks. IEEE Pers. Commun., 6: 46-55.

Sung, J.L. and M. Gerla, 1999. A Simulation Study of Table-Driven and On-Demand Routing Protocols for Mobile Ad Hoc Nteworks. IEEE Network, pp: 48-54.

Tracy, C., J. Boleng and V. Davies, 2002. A Survey of Mobility Models for Ad Hoc Network, Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, 2: 483-502.

User Manual for IMPORTANT Mobility Tool Generator in NS-2 Simulator, 2004. http://nile.usc.edu/important/software.htm.