Effect of Mobility on Reactive Protocols Using Genetic Algorithm

Arun Biradar, Dr. Ravindra C. Thool

Abstract— Ad- hoc routing in wireless scenarios is very interesting subject for research because without base station it is very difficult to maintain mobile nodes to connect and communicate to each other so there is need of wireless protocol to make work easy. For such type routing protocols is main area of research since packets are transmitted by hop by hop, therefore various performances varies from protocol to protocol, their usefulness is major concern. The goal of this paper is to find the shortest path from source to destination node in AODV, DSR and AOMDV using genetic algorithm on different metric.

Index Terms— Ad Hoc On-Demand Distance Vector (AODV), Dynamic Source Routing (DSR), Ad Hoc On-Demand Multipath Distance Vector (AOMDV), Mobile Ad Hoc Networks (MANETs), Genetic Algorithm (GA).

◆

1 INTRODUCTION

THIS article examines routing protocols designed for these d hoc networks by first describing the operation of each of the protocols and then comparing their various characteristics. Our goal is to carry out a systematic performance study of dynamic routing protocols for ad hoc networks, the Dynamic Source Routing protocol (DSR) , Ad Hoc On-Demand Distance Vector protocol (AODV), Ad Hoc On-Demand Multipath Distance Vector protocol (AOMDV). DSR, AODV and AOMDV share an interesting common characteristic they both initiate routing activities on an on demand basis. The key motivation behind the design of on-demand protocols is the reduction of the routing load. High routing load usually has a significant performance impact in low bandwidth wireless links. While DSR and AODV share the on-demand behavior in that they initiate routing activities only in the presence of data packets in need of a route, many of their routing mechanics are very different. In particular, DSR uses source routing, whereas AODV use a table-driven routing framework and destinations sequence numbers. DSR does not rely on any timer based activities, while AODV does to a certain extent.

AOMDV shares several characteristics with AODV. It is based on the distance vector concept and uses hop-by-hop routing approach. Our main objective is to evaluate the effectiveness of AOMDV relative to AODV in the presence of mobility-related route failures. AOMDV always drops fewer packets with improvements. Smaller packet loss with AOMDV is because of the availability of alternate paths to forward the packets when one path fails.

2 SOURCE INITIATED ON-DEMAND ROUTING

A different approach from table-driven routing is source-initiated on-demand routing. This type of routing creates routes only when desired by the source node. When a node requires a route to a destination, it initiates a route discovery process within the network.

This process is completed once a route is found or all possible route permutations have been examined.

Once a route has been established, it is maintained by a route maintenance procedure until either the destination becomes inaccessible along every path from the source or until the route is no longer desired.

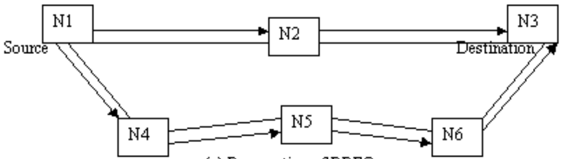
2.1 Ad Hoc On-Demand Distance Vector Routing (AODV)

Detailed submission guidelines can be found on the author resources Web pages. The Ad Hoc On-Demand Distance Vector (AODV) routing protocol described in [1] builds on the DSDV algorithm previously described. AODV is an improvement on DSDV because it typically minimizes the number of required broadcasts by creating routes on a demand basis, as opposed to maintaining a complete list of routes as in the DSDV algorithm. The authors of AODV classify it as a pure on-demand route acquisition system, since nodes that are not on a selected path do not maintain routing information or participate in routing table exchanges [1]. When a source node desires to send a message to some destination node and does not already have a valid route to that destination, it initiates a path discovery process to locate the other node. It broadcasts a route request (RREQ) packet to its neighbors, which then forward the request to their neighbors, and so on, until either the destination or an intermediate node with a fresh enough routes to the destination is located. Figure 1a illustrates the propagation of the broadcast RREQs across the network.

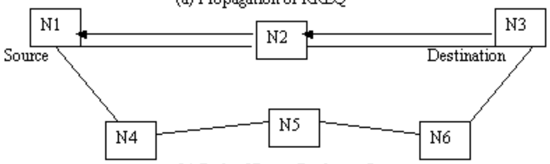
- *Arun Biradar is pursuing Ph.D. degree e in Computer Science & Engineering in SGGS Institute of Engineering & Technology, Nanded, Maharashtra, India. E-mail: ambiradar@yahoo.com*
- *Dr.Ravindra C. Thool is working as a Professor & Head, Department of Information Technology, SGGS Institute of Engineering & Technology, Nanded, Maharashtra, India. E-mail: rcthool@yahoo.com*

AODV utilizes destination sequence numbers to ensure all routes are loop-free and contain the most recent route information. Each node maintains its own sequence number, as well as a broadcast ID. The broadcast ID is incremented for every RREQ the node initiates, and together with the nodes IP address, uniquely identifies an RREQ. Along with its own sequence number and the broadcast ID, the source node includes in the RREQ the most recent sequence number it has for the destination. Intermediate nodes can reply to the RREQ only if they have a route to the destination whose corresponding destination sequence number is greater than or equal to that contained in the RREQ. During the process of forwarding the RREQ, intermediate nodes record in their route tables the address of the neighbor from which the first copy of the broadcast packet is received, thereby establishing a reverse path. If additional copies of the same RREQ are later received, these packets are discarded. Once the RREQ reaches the destination or an intermediate node with a fresh enough route, the destination/intermediate node responds by unicasting a route reply (RREP) packet back to the neighbor from which it first received the RREQ fig. 1b.

As the RREP is routed back along the reverse path, nodes along this path set up forward route entries in their route tables, which point to the node from which the RREP came. These forward route entries indicate the active forward route. Associated with each route entry is a route timer which will cause the deletion of the entry if it is not used within the specified lifetime. Because the RREP is forwarded along the des listen for retransmission of data packets to ensure that the next hop is still within reach. If such a retransmission is not heard, the node may use any one of a number of techniques, including the reception of hello messages, to determine whether the next hop is within communication range. The hello messages may list the other nodes from which a mobile has heard, thereby yielding greater knowledge of network connectivity.



(a) Propagation of RREQ



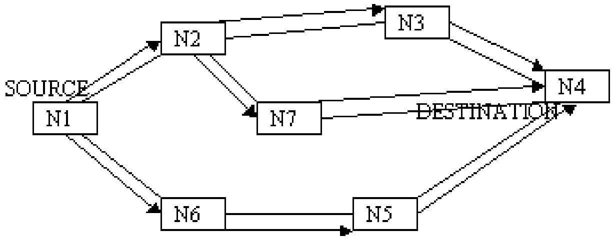
(b) Path of Route Replay to Source

Figure 1. AODV Route Discovery

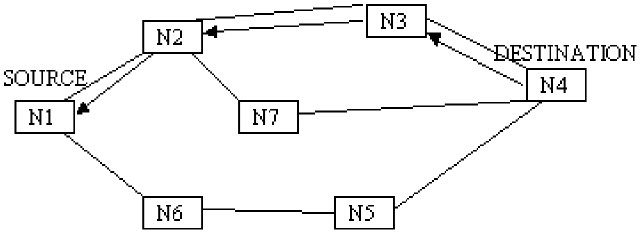
2.2 Dynamic Source Routing (DSR)

The Dynamic Source Routing (DSR) protocol presented in[2] is an on-demand routing protocol that is based on the concept of source routing. Mobile nodes are required to maintain route caches that contain the source routes of which the mobile is aware. Entries in the route cache are continually updated as new routes are learned. The protocol consists of two major phases: route discovery and route maintenance. When a mobile node has a packet to send to some destination, it first consults its route cache to determine whether it already has a route to the destination. If it has an unexpired route to the destination, it will use this route to send the packet. On the other hand, if the node does not have such a route, it initiates route discovery by broadcasting a route request packet. This route request contains the address of the destination, along with the source node s address and a unique identification number. Each node receiving the packet checks whether it knows of a route to the destination. If it does not, it adds its own address to the route record of the packet and then forwards the packet along its outgoing links. To limit the number of route requests propagated on the outgoing links of a node, a mobile only forwards the route request if the mobile has not yet seen the request and if the mobile s address does not already appear in the route record. A route reply is generated when the route request reaches either the destination itself, or an intermediate node, which contains in its route cache an unexpired route to the destination. By the time the packet reaches either the destination or such an intermediate node, it contains a route record yielding the sequence of hops taken.

Figure 2a illustrates the formation of the route record as the route request propagates through the network. If the node generating the route reply is the destination, it places the route record contained in the route request into the route reply. If the responding node is an intermediate node, it will append its cached route to the route record and then generate the route reply. To return the route reply, the responding node must have a route to the initiator. If it has a route to the initiator in its route cache, it may use that route. Otherwise, if symmetric links are supported, the node may reverse the route in the route record. If symmetric links are not supported, the node may initiate its own route discovery and piggyback the route reply on the new route request.



(a)Building of route record during route discovery



(b)Propagation of the route replay with the route record

Figure 2.Creation of Route record in DSR

Figure 2b shows the transmission of the route reply with its associated route record back to the source node. Route maintenance is accomplished through the use of route error packets and acknowledgments. Route error packets are generated 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 all routes containing the hop are truncated at that point. In addition to route error messages, acknowledgments are used to verify the correct operation of the route links. Such acknowledgments include passive acknowledgments, where a mobile is able to hear the next hop forwarding the packet along the route.

2.3 Ad-Hoc On-Demand Multipath Distance Vector (AOMDV)

AOMDV extends the AODV protocol to discover multiple paths between the source and the destination in every route discovery. Multiple paths so computed are guaranteed to be loop-free and disjoint. In AOMDV, RREQ propagation from the source towards the destination establishes multiple reverse paths both at intermediate nodes as well as the destination. Multiple RREPs traverse these reverse paths back to form multiple forward paths to the destination at the source and intermediate nodes. AOMDV also provides intermediate nodes with alternate paths as they are found to be useful in reducing route discovery frequency.This mechanism reduces route discovery latency and the routing overheads.

Single path routing protocols have been heavily discussed and examined in the past. More recent research topics for MANETs are multi-path routing protocols. Multi-path routing protocols establish multiple disjoint paths from a source to a destination and are thereby improving resilience to network failures and allow for network load balancing. These effects are particularly interesting in networks with high node density (and the corresponding larger choice of disjoint paths) and high network load (due to the ability to load balance the traffic around congested networks). A comparison of multiple multipath protocols is therefore particularly interesting in scenarios of highly congested and dense networks.

AOMDV outperforms the other protocols in highly mobile networks. Multi-path routing is advantageous in networks of high node density.

3 AD HOC NETWOK WITH GENETIC ALGORITHM

Ad hoc network can represented as a connected graph with N nodes. The optimization metric is cost of path between the source and destination. The sum of cost of individual hops is total cost.

The goal is to find the path with minimum total cost between source and destination node using genetic algorithm with simple and effective manner.

The steps need to be followed are as below:

3.1 Representation of Chromosome

The path from source to destination node is a feasible solution. Optimal solution is the shortest one. Beginning a random population of strings is generated, which represents feasible or unfeasible solutions. Unfeasible solutions are strings that can't reach the destination. A Chromosome corresponds to possible solution of the optimization problem. Thus each chromosome represents a path which consists of sequences of positive integers that represent IDs of nodes through which a routing path passes with source node followed by intermediate node to destination node. The default maximum chromosome length is equal to the number of nodes.

3.2 Evalutaion of Fitness Function

Fitness function is defined as follows:

$$f_i = \frac{1}{\sum_{j=1}^{l_i-1} C_{g_i(j)g_i(j+1)}}$$

Where f_i represents the fitness value of the i^{th} chromosome l_i is the length of the i^{th} chromosome, g(j) represents the gene (node) of the j^{th} locus in the i^{th} chromosome and C is the link cost between nodes[1]. In the proposed algorithm the link cost are considered to be equal to each other and to 1. This means the cost which represents the shortest distance is the hop count.

3.3 Selection of Best Fit

Selection process of the best fit is done to improve the average quality of the population. Process gives the better chance to the best chromosome to survive. Two types of selection process: Proportionate and Ordinal-based selection. Proportionate selection picks out chromosomes based on their fitness values relative to the fitness of the other chromosome in the population. This selection includes roulette wheel selection, stochastic remainder selection and stochastic universal selection[1]. This paper use the roulette wheel concept, the values providing the best fit being given a higher percentage on the wheel area, so that values providing a better fit have higher probability of producing an offspring.

3.4 Crossover Operator

Crossover selects genes from parent chromosomes and creates a new offspring. Crossover is performed on strings using midpoint crossover. Midpoint crossover divides the parent's chromosomes into two from the midpoint. Crossover provides incorporation of extra characteristics in the off spring produced.

3.5 MutationOperator

Mutation operator randomly alters genes to partially shift the search to new locations in the solution pace. Mutation is done if consecutive iteration values are the same.

4 SIMULATION MODEL

We use a detailed simulation model based on ns-2 in our evaluation. The Distributed Coordination Function (DCF) of IEEE 802.11 [3] for wireless LANs is used as the MAC layer protocol. The 802.11 DCF uses Request-To-Send (RTS) and Clear-To-Send (CTS) control packets [4] for unicast data transmission to a neighboring node. The RTS/CTS exchange precedes data packet transmission and implements a form of virtual carrier sensing and channel reservation to reduce the impact of the well-known *hidden terminal problem*. Data packet transmission is followed by an ACK. Broadcast data packets and the RTS control packets are sent using physical carrier sensing. An unslotted carrier sense multiple access (CSMA) technique with collision avoidance (CSMA/CA) is used to transmit these packets. The radio model uses characteristics similar to a commercial radio interface, lucent S wave LAN. Wave LAN is modeled as a shared-media radio with a nominal bit rate of 2 Mb/s and a nominal radio range of 250 m. A detailed description of the simulation environment and the models is available in[5 6]. The implementations of AODV and DSR in our simulation environment closely match their specifications[5]. The routing protocol model detects all data packets transmitted or forwarded, and responds by invoking routing activities as appropriate. The RREQ packets are treated as broadcast packets in the MAC. RREP and data packets are all unicast packets with a specified neighbor as the MAC destination. RERR packets are treated differently in the two protocols. They are broadcast in AODV and use unicast transmissions in DSR. Both protocols detect link breaks using feedback from the MAC layer. A signal is sent to the routing layer when the MAC layer fails to deliver a unicast packet to the next hop. This is indicated, for example, by the failure to receive a CTS after a specified number of such as hello messages is used. Both protocols maintain a send RTS retransmissions, or the absence of an ACK following data transmission. No additional network layer mechanism buffer of 64 packets. It contains all data packets waiting for a route, such as packets for which route discovery has started, but no reply has arrived yet. To prevent buffering of packets indefinitely, packets are dropped if they wait in the send buffer for more than 30 s. All packets (both data and routing) sent by the routing layer are queued at the interface queue until the MAC layer can transmit them. The interface queue has a maximum size of 50 packets and is maintained as a priority queue with two priorities each served in FIFO order. Routing packets get higher priority than data packets.

A. Traffic and Mobility Models

We use traffic and mobility models similar to those previously reported using the same simulator. Traffic sources are continuous bit rate (CBR). 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. The mobility model uses the random waypoint model in a rectangular field. Two field configurations are used:
1500 m X 300 m and 50 nodes generated mobility scenarios. Identical mobility and traffic scenarios are used across protocols.

5 PERFORMANCE RESULT

A. Performance Metrics

Important performance metrics are evaluated:

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: Average end to end delay is an average end to end delay of data packets that is average time needed to transfer a data packet from source to destination. Once the difference between every CBR packets sent and received is found, it is divided by the total number of CBR packets received. This gives the average end to end delay for received packets. The lower is the end to end delay, the better the application performs.

e2edelay = $\sum$ (CBRrcvTime-CBRsentTime)/$\sum$ CBRrcv

Normalized routing load: It is defined as the total number of routing packets transmitted per data packet. It is calculated by dividing the total number of routing packets sent (includes forwarded routing packets as well) by the total number of data packets received.

nrl = $\sum$ RTRpkts/$\sum$ CBRrcv

6 COMPARISION

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 on 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 a 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 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 replay 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.

7. CONCLUSION

The first experiment uses 10 number of sources with varying pause time (see fig 3), we observed the there is large variation in DSR with respect to AODV arc. This is due to the property, that explained in the section 4, because for the large mobility (small pause time) AODV and DSR both are busy in maintaining communication link and therefore lot of control packets are needed. Large packets drops, therefore we got variation in both protocol but in DSR, it is much more due to its route caching aggressiveness causing network busy too much that is why more collision occurs and therefore much packet drop occurs. Near 300 s pause time we observe that DSR has much better performance than AODV. So we conclude that for the high mobility AODV gives better performance with the quicker changes.

Here the utility of AODV is in the area where speed is more desirable and communication also for example Highway, Aero dram, shipyard etc. DSR is most preferable to those area where the speed is not so serious matter for example Institution, Industries, Organization etc.

In the Figure 4 we observed that delay for DSR is decreasing with the increase of pause time because the overhead of route discovery is reduces with lowering mobility and due to which extra time in retransmission of packets and wasting in congested traffic is reduced. Delay for 10 sources with higher mobility the AODV has better response then DSR. Here we also observed that AODV produces result of increasing trend of delay (Fig. 2d). This is due to a high level of network, congestion and multiple access interferences at certain regions of the ad hoc network. Neither protocol has any mechanism for load balancing, that is, choosing routes in such a way that data traffic can be more evenly distributed in the network.

This phenomenon is less visible with higher mobility where traffic automatically gets more evenly distributed due to source movements. Since DSR gives delay so we deploy this protocol in the Institutions, Industries and Organizations etc where delay is considerable due to slow speed region. Where AODV gives low delay at high mobility which causing it best suitable for Highway, aero dram etc.

In case of routing overload Figure 5, DSR gives better performance than AODV. Since In high-mobility scenarios, nodes the routing load of AODV is about twice as much as mobile DSR with 10. For both protocols, routing load drops with increase in pause time (decrease in mobility). Since routing overhead of DSR is less then AODV throughout the variable pause time because in DSR no routing update information is send for small interval and in this protocol uses cache information of routes are used for route maintenance which it gathered while route discovery process. But in AODV for specific interval of time every nodes send route update information, which makes control packets wandering all the time in the network.

Routing overhead is not needed to much in the area where speed is not very serious thing therefore DSR is best for those area like Industries etc. But AODV better for High speed area as we got our in two previous results therefore in the routing overhead case we ignore it as comparison to good communication with less packet drop.

In Figure 6, the MAC Load is showing something similar trend the as routing overhead is showing, but with higher mobility , there is very less difference between DSR & AODV which is going to be more & more at lower mobility, Utility of DSR protocol in the area of lower mobility zone is also verifies by this result.

This paper presented a genetic algorithm for solving the shortest path (SP) routing problem and performs better and effectively even to the changes in the network due to node mobility and topology changes.

In summary for the high mobility DSR has low packet delivery ratio and normalized routing load but high delay.

AOMDV achieves the best performance in scenarios with high node mobility.

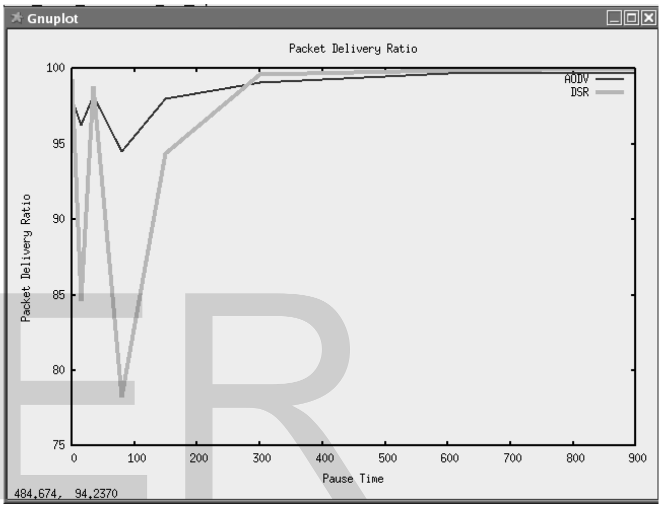


Figure 3. Packet delivery ratio for 10 sources with 50 nodes

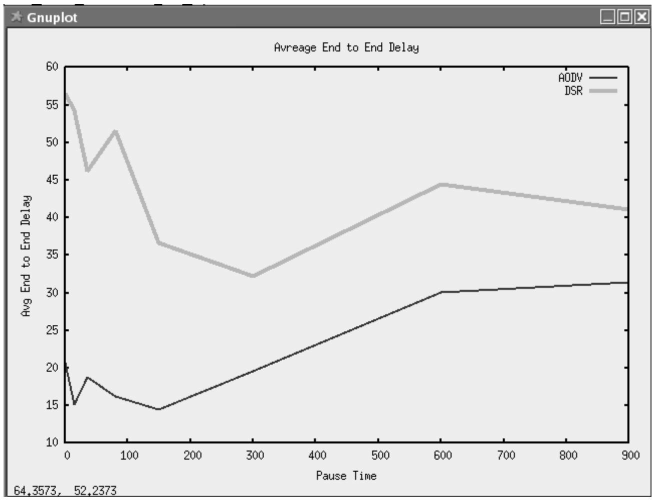


Figure 4. Average packet delay for 50 nodes and 10 sources nodes

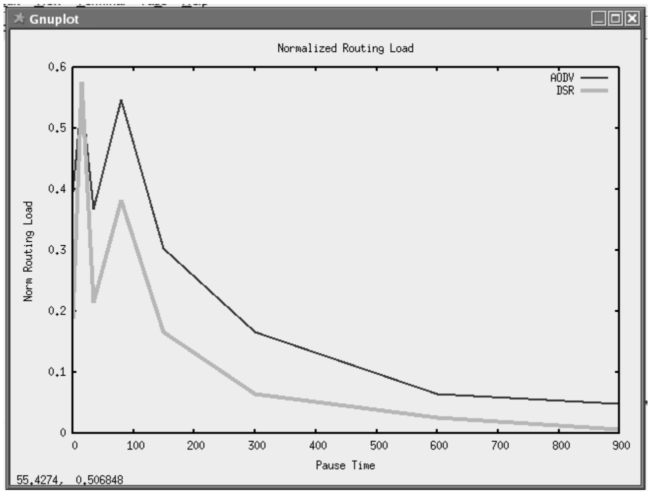


Figure 5. Normalized routing load for 50 nodes with 10 source

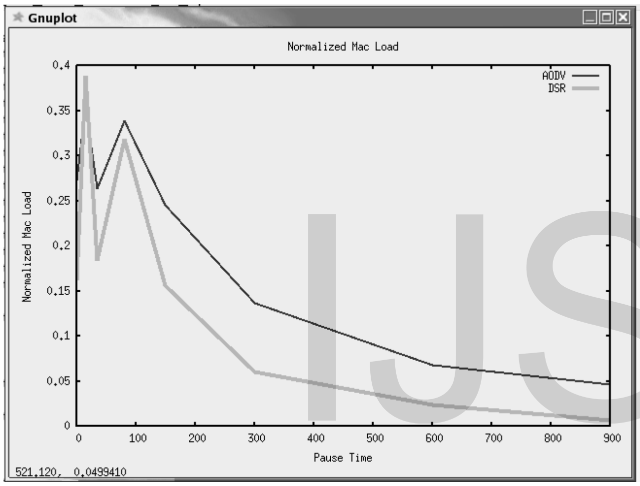


Figure 6 Normalized MAC load for 50 nodes with 10 source

ACKNOWLEDGMENT

We are thankful to the Department of Computer Science & Engineering and Information Technology of Shri Guru Govind Singhji Institute of Engineering and Technology (SGGSIE&T), Nanded, Maharashtra, India.

REFERENCES

[1] C. E. Perkins and E. M. Royer, Ah-Hoc On Demand Distance Vector Routing, Proc. 2nd IEEE Workshop Mobile Comp. Sys. And Apps., Feb. 1999, pp 90-100.

[2] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," T. Imielinski and H. Korth, Eds. Mobile Computing, Ch. 5,Kluwer, 1996.

[3] C. W. Ahn and R. S. Ramakrishna, A Genetic algorithm for shortest path routing problem and the sizingof population, IEEE Transactions on Evalutionsry computation, vol 6, no. 6, Dec. 2002.

[4] Samir R. Das, Charles E. Perkins & Elizabeth M. Royer Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks" http:// www.ietf.org/ internet-drafts/ draft- ietf-manet-aodv- 03.txt, June 1999. IETF Internet Draft.

[5] Tsirigos, A., Haas, Z.: Multipath routing in the presence of frequent topological changes. IEEE Communications Magazine 39 (2001)

[6] D. E. Goldberg, genetic Algorithms in Search, Optimization and machine Learning, Addtion-Wesley Publishing Company, 1989.

[7] S. mardle S. Pascoe, An overview of genetic algorithm for the solution of optimization problems, Computers in Higher Education economics Rivew (CHEER), Vol. 13, pages 16-20, 1999.

[8] J. Broch et al., "A Performance Comparison of Multihop Wireless AdHoc Network Routing Protocols," Proc. IEEE/ACM MOBICOM '98, Oct. 1998, pp. 85–97.

[9] V. Bharghavan et al., MACAW: A Media Access Protocol for Wireless LANs, Proc. ACM SIGCOMM '94, Aug.1994,pp. 212– 25.

[10] IEEE, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11 1997

[11] ns 2: Network Simulator: (http://www.isi.edu/nsnam/ns/.)

[12] Wang, L., Shu, Y., Dong, M., Zhang, L., Yang, O.: Adaptive multipath source routing in ad hoc networks. IEEE ICC 3 (2001) 867–871

[13] Ganjali, Y., Keshavarzian, A.: Load balancing in ad hoc networks: Single-path routing vs. multi-path routing. IEEE INFOCOM (2004)

[14] S. Mueller, R. Tsang, D. Ghosal, "Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges", Lecture Notes in Computer Science (LNCS 2965), pp. 209-234, 2004.

[15] C. Siva Ram Murthy and B.S. Manoj, "Ad hoc Wireless Networks Architecture and protocols", Prentice Hall, 2004

ABOUT AUTHOR'S

Arun Biradar has received B.E, M.Tech. degree in Computer Science & Engineering and Pursuing Ph.D. in Computer Science and Engineering at Shri Guru Gobind Singhji Institute of Engineering and Technology (SGGSIE&T), Nanded, Maharashtra, India. His main research interests are Wireless Ad-Hoc Networks and Genetic Algorithms.

Dr. Ravindra C. Thool received his Ph. D. degree in Electronics and Computer Engineering. He is working as a Professor and Head, Department of Information Technology, Shri Guru Gobind Singhji Institute of Engineering and Technology (SGGSIE&T), Nanded, Maharashtra, India. He is a member of IEEE, ISTE and ASABE (USA). His main research interests are Wireless Ad-Hoc Networks, Computer Networks, Genetic Algorithms and Computer Vision.