MRA: Multi-level Routing Algorithm to Balance the Traffic Load in Wireless Ad hoc Network

Kevin Joy Dsouza
Dept. of CSE
St. Joseph Engineering College, Mangalore
joydsouza33@gmail.com

Sujatha M
Dept. of CSE
St. Joseph Engineering College, Mangalore
sujatha_msk@yahoo.co.in

***Abstract*— A critical requirement in the design of Ad hoc networks is the development of an efficient routing protocol which provides efficient communication. The node's in MANET have limited communication resources such as bandwidth, buffer space, battery power etc. The resource constraints in MANET require the traffic to be fairly distributed among the mobile host. Multi-level routing algorithm proposed in this paper facilitates an efficient method for communication with mobile node's. It reduces the communication overhead by fairly distributing the traffic among the wireless nodes. It selects the intermediate node's which have the enough resources and capability to handle the information transmitted in order to reach the destination node. The routing algorithm becomes more suitable as the mobile nodes move, it reduces the average end to end delay and increases the number of control packets sent with respect to increase in the number of node's.**

Keywords*— *MANET, Ad hoc, Multi-level;

I. Introduction

In MANETs communication between nodes is done through the wireless medium. Nodes are mobile and may join or leave the network any time, MANETs have a dynamic topology. The Nodes which are in transmission range of each other are called neighbors. Information can be sent directly to each other with respect to neighbors. But, when a node needs to send data to another non-neighboring node, the data is routed through a sequence of multiple hops, with intermediate nodes acting as routers.

- **Unpredictability of environment:** Ad hoc networks may be deployed in unknown terrains, hazardous conditions, and even hostile environments where tampering or the actual destruction of a node may be imminent. Depending on the environment, node failures may occur frequently.
- **Unreliability of wireless medium:** Communication through the wireless medium is unreliable and subject to errors. Also, due to varying environmental conditions such as high levels of electro-magnetic interference (EMI) or inclement weather, the quality of the wireless link may be unpredictable. Furthermore, in some applications, nodes may be resource-constrained and thus would not be able to support transport protocols necessary to ensure reliable communication on a lossy link. Thus, link quality may fluctuate in a MANET.

The topology in an Ad hoc network may change constantly due to the mobility of nodes. As nodes move in and out of range of each other, some links break while new links between nodes are created.

Nodes in a MANET are typically battery powered as well as limited in storage and processing capabilities. Moreover, they may be situated in areas where it is not possible to re-charge and thus have limited lifetimes. Because of these limitations, they must have algorithms which are energy-efficient as well as operating with limited processing and memory resources. The available bandwidth of the wireless medium may also be limited because nodes may not be able to sacrifice the energy consumed by operating at full link speed.

As a result of these issues, MANETs are prone to numerous types of faults including,

- Transmission errors: The unreliability of the wireless medium and the unpredictability of the environment may lead to transmitted packets being garbled and thus received in error.
- Node failures: Nodes may fail at any time due to different types of hazardous conditions in the environment. They may also drop out of the network either voluntarily or when their energy supply is depleted.
- Link failures: Node failures as well as changing environmental conditions may cause links between nodes to break.
- Route breakages: When the network topology changes due to node/link failures and/or addition or deletion of node/link to the network, routes become out of date and thus incorrect. Depending upon the network transport protocol, packets forwarded through stale routes may either eventually be dropped or be delayed; packets may take a circuitous route before eventually arriving at the destination node.
- Congested nodes or links: Due to the topology of the network and the nature of the routing protocol, certain nodes or links may become over utilized, i.e., congested. This will lead to either larger delays or packet loss. Routing protocols for MANETs must deal with these issues to be effective. In the remainder of this section, we present an overview of some of the key uni-path routing protocols for MANETs.

II. RELATED WORK

The paper [1] gives a brief idea about unit disk graphs. This graph provides a graph-theoretic model for broadcast networks (cellular networks) and for some problems in computational geometry and knowledge about the nodes interaction in the network. By referring to this it is understood that many standard graph theoretic problems remain NP-complete on unit disk graphs, including, independent set, domination, coloring, independent domination, and connected domination. NP-completeness for the domination problem is shown to hold even for grid graphs, a subclass of unit disk graphs.

The paper [2] explains a protocol that is used for routing in Ad hoc networks that uses dynamic source routing. To send a packet to a different host, the sender constructs a source route within the packet's header, giving the address of every host within the network through which the packet should be forwarded so as to reach the destination host. The sender then transmits the packet over its wireless network interface to the primary hop identified within the supply route. Once a host receives a packet, if this host is not the final destination of the packet, it merely transmits the packet to subsequent hop known within the source route within the packet's header. Once the packet reaches its final destination, the packet is delivered to the network layer code on that host.

In this paper [3] the author explains regarding Ad hoc On Demand Distance Vector Routing (AODV), an algorithmic program that is employed for the operation of such Ad hoc networks. In this protocol every Mobile Host operates as a specialized router, and routes are obtained as needed (i.e., on-demand) with very little or no reliance on periodic advertisements. This is often quite appropriate for a dynamic self- beginning network, as required by users desire to utilize Ad hoc networks. AODV provides loop-free routes even while repairing broken links within the network. As a result, this protocol does not need global periodic routing advertisements, the demand on the general information measure available to the mobile nodes is substantially but in those protocols that do necessitate such advertisements.

In this paper [4] information about power-aware metrics for determining routes in wireless Ad hoc networks is presented and different metrics based on battery power consumption at nodes. It is observed that using these metrics in a shortest-cost routing algorithm reduces the cost/packet of routing packets by 5-30% over shortest-hop routing, this cost reduction is on top of a 40-70% reduction in energy consumption obtained. These metrics ensures that the mean time to node failure is increased significantly. By using shortest-cost routing is that packet delays do not increase.

In this paper [5] information about oblivious routing algorithms in which packet paths are constructed independently described. It gives a simple oblivious routing algorithm for geometric networks (networks which are embedded in the Euclidean plane) where random intermediate node has to be chosen in the space between the source and destination, and then send the packet to its destination through the intermediate node. This algorithm can be analyzed in terms of stretch and congestion. It says that the stretch is constant, and the congestion is near optimal when the network paths can be chosen to be close to the geodesics.

The paper [6] presents a general framework for analyzing the traffic load resulting from a given set of paths and traffic demands are presented. The load balancing problem is a minmax problem and gives two lower bounds for the achievable minimal maximum traffic load. The framework is illustrated by giving an example of uniformly distributed traffic demands in a unit disk with a few families of paths given in advance. By using these paths it is possible to decrease the maximum traffic load by factor of 33 − 40% depending on the assumptions. It says that obtained traffic load level also comes quite near the tightest lower bound.

This paper [7] gives the solution for the problem when routing along shortest paths. In this approach nodes that are centrally located forward a disproportionate amount of traffic. This results in increased congestion and energy consumption. But the maximum load can be decreased if the packets follow curved paths by using such routing scheme. This can be expressed in terms of geometric optics and computed by linear programming.

III. PROPOSED ALGORITHM

Here network is divided into disjoint regions. If suppose the source and destination nodes lie in the same region then the nearest neighbor node having enough resources is used as intermediate node to reach destination. In the condition when source and destination node lie in the different region, condition is used to check weather intersection of source and destination nodes lie on the network center path. If intersection of source and destination does not lie on path to network Centre then nearest neighbor node is used to transmit the information. If intersection of source and destination lie on the path to network Centre then route goes through an intermediate point to avoid the network Centre. The technique is as follows let l be the line connecting source and destination. Also let l' denote line that goes through network origin and is perpendicular to l. to be able to select nodes away from the network center we select the node which is at distance r_{intP} which lie on line l'.

We use two sets of parameters $r = [r1, r2, ..., r_k]$, where $0 \leq r1 \leq r2 \leq ... \leq r_k \leq R$ and $f = [f1, f2, ..., f_k]$. The parameters r_i are used to divide the network area into disjoint regions. The functions $fi: R \times R \rightarrow R$ are used to assign the intermediate points. Here network area is divided into two regions namely the inner region with distance r from the network centre and the outer region outside it. The decision of the function to use is made based on the regions where the source node and the destination node are located. In the first stage of the algorithm the number of intermediate points is determined. We find the largest i such that at least one of the source or the destination nodes is at distance greater than r_i to the network centre. The number of intermediate points is then calculated as 2^{i-1}.The coordinates of the intermediate points are calculated and stored in an array *intP* of size$2^i - 1$ using a recursive algorithm. First the middle element (i.e. *intP*[2^{i-1}]) is calculated as in Algorithm 3 using the given source and destination. Next the algorithm recursively fills in the first half of the *intP* array using the given source and *intP*[2^{i-1}] as destination, and similarly recursively fills in the second half of the *intP* array using *intP*[2^{i-1}] as source and the given destination.

ALGORITHM 1 : MLR Routing

---

function multilevel(*src, dest, r[], f[])*

path ← (*src*)

if dist(*src, dest*) ≤ Tr **then**

  path ← (*path, dest*)

  return *path*

end if

{else}

curr ← *src*

intPoints ← assignIntP(*src, dest, r, f*)

for i = 1 to size(*intPoints*) **do**

 {use greedy path to a node within the transmission range of intP[i]}

  while dist(*curr, intP[i]) > Tr* **do**

   curr ← greedy(*curr, intP[i])*

   path ← (path, curr)

  end while

end for

 {use greedy path to a node within the transmission range of dest}

while dist(*curr, dest*) > *Tr* **do**

curr ← greedy(*curr, dest*)

path ← (*path, curr*)

end while

path ← (*path, dest*)

return *path*

ALGORITHM 2: Assigns Intermediate Points

---

function assignIntP(*src, dest, r*[], *f*[])

k ← size(*r*)

level ← 0

numberIntP ← 0

for *i* = *k* downto 1 **do**

if dist(*src, origin*) > *r*[*i*] or dist(*dest, origin*) > *r*[*i*]

then

level ← *i*

numberIntP ← $2^i - 1$

break;

end if

end for

if *level* == 0 **then**

intP← an empty array of size 1

intP[1] ← *dest*

else

intP ← an empty array of size *numberIntP*

calcIntPArray(*src, dest, f*[*level*], *intP*)

end if

return *intP*

procedure calcIntPointsArray(*p*1*, p*2*, f,A*[])

  midIndex ← size(*subArray*)/2

  calcIntPoint(*p*1*, p*2*, f,A*[*midIndex*])

  if size(*A*) > 1 **then**

  *p*3 ← *A*[*midIndex*]

  calcIntPArray(*p*1*, p*3*, f,A*[1..*midIndex* − 1])

  calcIntPArray(*p*3*, p*2*, f,A*[*midIndex* + 1..*end*])

end if

ALGORITHM 3: calculate intermediate points

procedure calcIntP($p1$, $p2$, f, $intP$)

($x1$, $y1$) ← cartesianCoordinates($p1$)

($x3$, $y3$) ← cartesianCoordinates($p2$)

l ← line going through ($x1$, $y1$) and ($x3$, $y3$)

l' ← line going through *origin* and perpendicular to l

($x2$, $y2$) ← intersect(l, l')

if ($x1 \le x2 \le x3$ and $y1 \le y2 \le y3$) or ($x1 \ge x2 \ge x3$ and $y1 \ge y2 \ge y3$) **then**

($r1$, $\theta1$) ← polarCoordinates($x1$, $y1$)

($r2$, $\theta2$) ← polarCoordinates($x2$, $y2$)

($r3$, $\theta3$) ← polarCoordinates($x3$, $y3$)

r ← f($r1$, $r3$)

θ ← $\theta2$

intP oint ← polar2cartesian(r, θ)

distance(*x,y,intP)*

else

intP ← *p2*

distance(*x,y,intP)*

end if

IV. SIMULATION AND RESULTS

The proposed routing protocol is simulated using network simulator 2(NS 2.34) [11]. The simulation environment is summarized in the Table I. As indicated in the TABLE I, the simulation was carried out for duration of seconds over an area of 1000 sq. meters with varying traffic from 5,10… to 25 connections.

TABLE I: SIMULATION PARAMETER

Simulation area(m x m)	1000 x1000
Simulation time(s)	100
Number of traffics	5,10,15,20,25
MAC layer protocol	IEEE 802.11
Transmission range (m)	250
Maximum velocity(m/s)	100
Hello interval	1ms

A. AVERAGE END TO END DELAY WITH NUMBER OF THE NODES

Results obtained for average end to end delay with respect to number of nodes can be observed from the graph. Average end to end delay of nodes is measured in time (ms) with respect to number of node's.

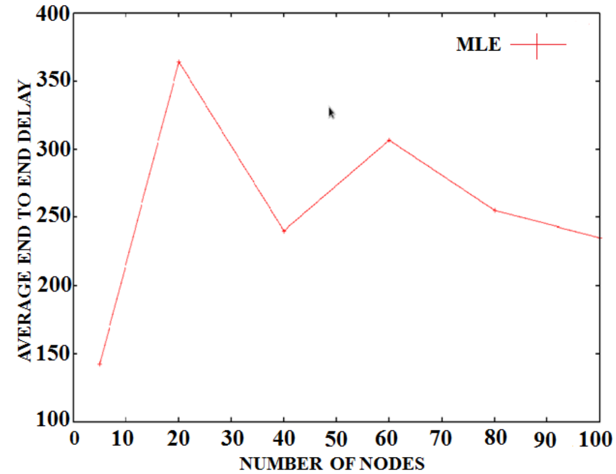

Fig. 1: Average End to End delay w.r.t Number of Node's

.

B. AVERAGE END TO END DELAY WITH RESPECT TO SPEED OF THE NODES

It is observed from the graph that the delay becomes comparatively less as the mobility of the nodes increases. This makes the protocol more suitable for Ad hoc network as the nodes are free to move randomly.

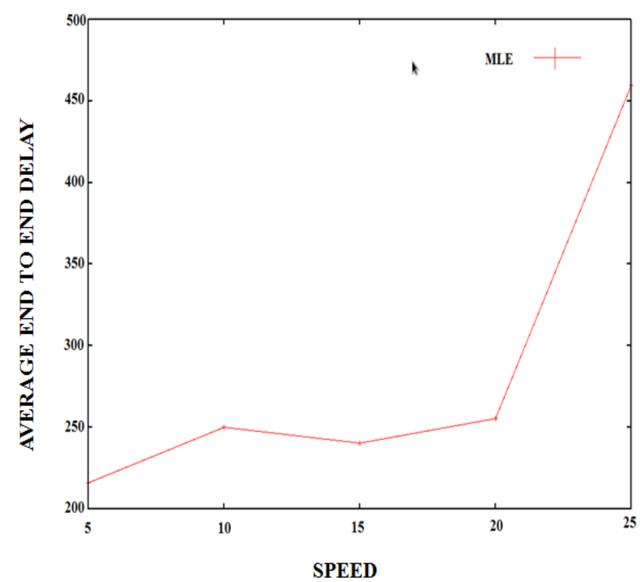

Fig. 2: Average End to End Delay w.r.t Speed of the Node's

C. CONTROL PACKETS SENT WITH RESPECT TO NUMBER OF NODES

The number of control packets sent with respect to the number of nodes is shown in the graph. The frequency of connection establishment increases with increase in the number of node's making the protocol more suitable for the Ad hoc network.

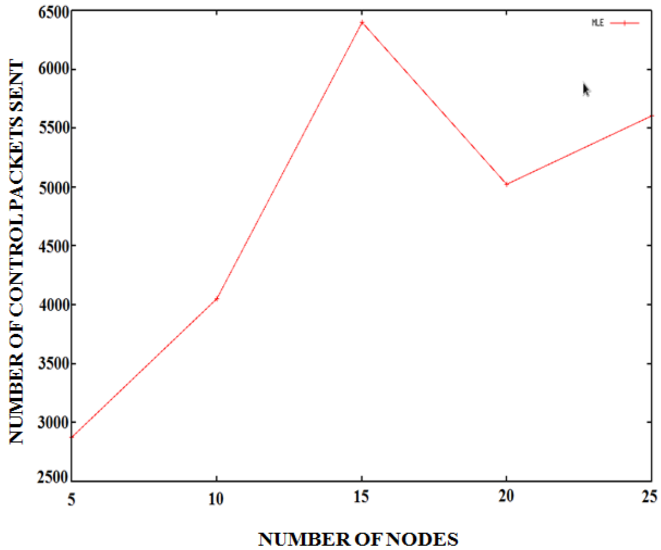

Fig. 3: Number of Control Packets Sent w.r.t Number of Node's

D. CONTROL PACKETS SENT WITH RESPECT TO SPEED OF THE NODES

The following graph shows results obtained for the number of control packets sent with respect to speed of the node's.

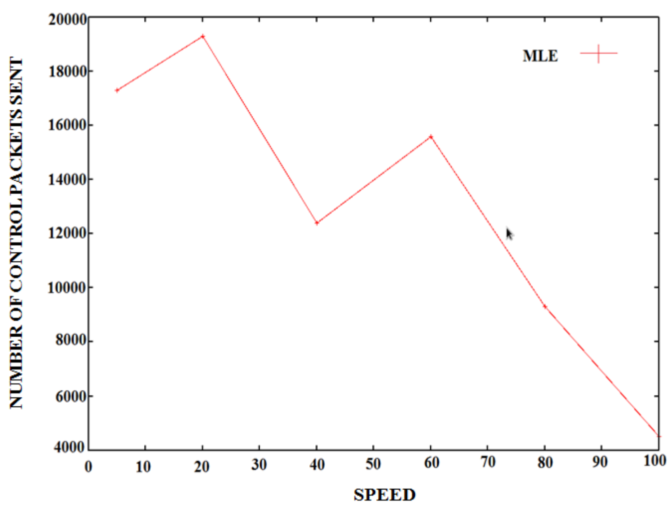

Fig. 4: Number of Control Packets Sent w.r.t speed of the Node's

From the following graph it is understood that protocol performs well, as the mobility of the nodes increases and making it more and more suitable to work in Ad hoc network.

V. CONCLUSION

The multi-level routing protocol proposed here uses an efficient method of selecting the intermediate nodes to reach the destination. The main requirement in wireless Ad hoc network is to use network resources efficiently such as bandwidth, power etc. Meanwhile it is always important to have the persistent data about the current network scenario of node statistics such as node position, signal strength, power. The routing protocol here incorporates an efficient method of broadcasting hello message. Whenever there is a change in the network, nodes broadcast hello message, accordingly all nodes update their routing table. The fundamental parameters of hello message are node position, signal strength and destination coordinates. Experimental results are evident to prove that the average end to end delay is reduced and connection establishment is very responsive, making it more suitable for Ad hoc network.

REFERENCES

[1] B. N. Clark, C. J. Colbourn, and D. S. Johnson, "Unit disk graphs" *Discrete Math.*, vol. 86, no. 1-3, pp. 165–177, 1990.

[2] D. Johnson and D. Maltz, "Dynamic source routing in Ad Hoc wireless networks," in *Kluwer Academic Publishers*, 1996, pp. 153–181.

[3] C. Perkins and E. Royer, "Ad Hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1997, pp. 90–100.

[4] S. Singh, M. Woo, and C. Raghavendra, "Power-aware routing in mobile Ad Hoc networks," in *Proceedings of MobiCom '98*, 1998, pp. 181–190.

[5] C. Busch, M. Magdon-Ismail, and J. Xi, "Oblivious routing on geometric networks," in *Proceedings of the 17th annual ACM symposium on Parallelism in algorithms and architectures*, 2005, pp. 316–324.

[6] E. Hyyti¨a and J. Virtamo, "On load balancing in a dense wireless multihop network," in *Proceedings of the 2nd Conference on Next Generation Internet Design and Engineering*, 2006, pp. 71–79.

[7] L.Popa, A.Rostamizadeh, R.Karp, C.Papadimitriou, and I. Stoica, "Balancing traffic load in wireless networks with curveball routing," in *Proceedings of MobiHoc '07*, 2007, pp. 170–179.

[8] Surabhi Jain1, Pankaj Dev Chadda,"A Dynamic Approach Of Location Based Routing In Mobile Ad Hoc Network

[9] Mohsen EftekhariHesari, Lata Narayanan, Jaroslav Opatrny, New Routing Algorithms to Balance Traffic Load , *IEEE WCNC 2011-Network,*pp 968-973.

[10] B. Karp and H. T. Kung, "Gpsr: greedy perimeter stateless routing for wireless networks," MobiCom 00: in Proc. of the 6th annual international conference on Mobile computing and networking, New York, NY, USA, pp. 243–254, 2000.

[11] Network simulator 2 (Ns2):http://www.isi.edu/nsnam/ns/

[12] http://ramakrishnamundugar.blogspot.in/