www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 5 May 2017, Page No. 21513-21522
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i5.54

New Multicast Routing Protocols for Mobile Adhoc networks

Harendra Kumar, Madan Kushwaha
ITM Mahararjganj(U.P)
BIE Lucknow(U.P)
Email:- harendradeep@gmail.com
Email:- kushwaha.86@gmail.com

Abstract:

The majority of applications are in areas where quick deployment and dynamic reconfiguration are necessary and a wire network is not available for that areas. These include military battlefields, emergency search and rescue sites, classrooms, and conventions where participants share information dynamically using their wireless devices. Well established routing protocols do exist to offer efficient multicasting service in conventional wired networks. These protocols, having been designed for fixed networks, may fails to keep up with node movements and frequent topology changes in a MANET. Therefore, adapting existing wired multicast protocols as such to a MANET, which completely lacks infrastructure, appear less promising. Providing efficient multicasting over MANET faces many challenges, includes scalability, quality of service, reliable service, security, Address configuration, Applications for multicast over MANET. The existing multicast routing protocol do not addresses these issues effectively over Mobile Adhoc Networks (MANET).

Keyword:- *AODV; ODMRP; NSMP; CAMP; ADMR;MANET*

Introduction:

Add few words for introduction and change according to your wish Ad hoc routing protocols are classified in to two different categories 1) **Proactive**. Proactive routing protocol updates their information periodically. Some Proactive routing protocols are DSDV [5]

DBF [6] etc. 2) **Reactive**. Reactive routing protocol creates a route destination on demand basic. Some of reactive routing protocols are DSR [7], AODV [8] etc.

The main objective of this paper is studying of mobile ad hoc network (MANET) routing protocols in grid environment. And it makes the comparison of ODMRP and AODV routing protocols, by using performance matrices and average end to end delay, packet delivery fraction, average routing load and data packets lost. And also the study of analysis discussed that which is the better one and which is should be implemented in MANET technology.

Literature review:

In past five years lot of researches and improvement done in MAINT technology and number of routing protocols proposed so far. The performance of the ODMRP routing protocol, which is one the most famous routing protocols for multi-hop ad hoc networks, is analyzed in. [4]. Comparison of routing protocols in MANET is difficult, because its performance matrices. Without measuring performance i.e. end to end delay of ODMRP and AODV, it is not suggested to implement.

On-Demand Multicast Routing Protocol (ODMRP)

On-Demand Multicast Routing Protocol (ODMRP) [20] is a reactive mesh-based multicast routing protocol which uses the concept of forwarding group (FG) to forward the control and data packets to group members. Scoped flooding is used by forwarding group members to deliver the multicast packets to group members via shortest path. Due to reactive nature of protocol, ODMRP consumes network bandwidth very efficiently and reduces channel overhead, since route is created and maintained only when data packets are required to be sent.

Group Creation

Multicast mesh is created by the source node whenever source node send data to the members of multicast group. Mesh is created by periodically broadcasting a JOIN QUERY (JQ) message by the source node to the all the nodes of wire-less network. Whenever a node receives a correct JQ message, it re-cords the address of its upstream link (from which it received JQ message) and rebroadcasts the message. The address of upstream link will be used later to relay the JOIN REPLY (JR) messages. Whenever a multicast group member receives a JQ message, it broadcasts a JR message to all its neighbors to announce its group membership. A node upon receiving JR message checks in its local table whether any next node ID matches with its own ID. If it does, then it realizes that it has forwarded the JQ message to the sender of JR message and it lies on the path from the source to the sender of JR message and thus becomes part of forwarding group (by switching on a bit field in JR message). It then informs all its neighbor nodes by broadcasting its own JR message and this process continues until JR reaches the source node via forwarding group nodes. Each forwarding group member may receive multiple JR message but relays only once to the source node. When JR message reaches the source node a mesh has been built and forwarding group has been constructed.

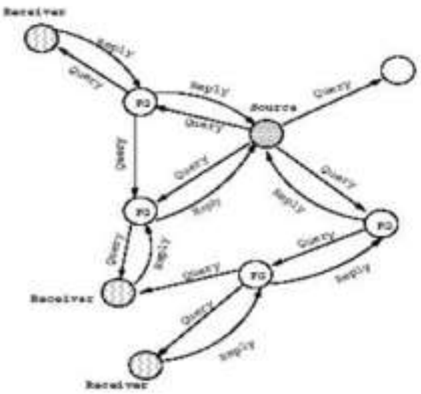


Figure 2.5 shows an example of mesh creation in ODMRP [20].

Group Maintenance

Group membership and multicast mesh is maintained in ODMRP though periodic transmission of JQ messages by the source node. All the states in ODMRP are soft states which are refreshed periodically by control message mentioned above or via transmission of data packets. Because of having soft states between the nodes, there is no need to send any special message to the member or to the neighbor nodes. If any member wants to leave a multicast group, it simply stops the generation of JR message in reply of JQ message by the source node.

Data Forwarding

Data is piggybacked by the source node in the first JQ message. All the sub-sequent data packets are sent after receiving JR messages from the group members via FG nodes. Source sends the data to receivers using the selected routes and FG nodes. Each intermediate node after receiving a non-duplicate data packet forwards it only if it is a member of FG.

ODMRP builds a group-shared forwarding mesh for each group. Each source performs periodic flood-response cycles, which create multicast forwarding state regardless of existing forwarding state. The frequent state discovery enabled the protocol to search the current shortest paths between each source node and the multicast receivers and improves the robustness of the protocol because there is exits multiple forwarding paths may exist between the members of the group. This is also why ODMRP's packet delivery ability improves as the number of sources and receivers per multicast group increases and sometimes with increased mobility: the redundant forwarding state improves ODMRP's packet delivery ability because it serves as a form of forward error correction, and makes the protocol less susceptible to mesh disconnection due to broken links. However, the frequent discovery floods and high speed number of data transmissions significantly increase network load.

Each multicast source for a group **G** in ODMRP periodically floods the network with a JOIN QUERY packet which is forwarded by all nodes in the network. This packet is sent every REFRESH INTE RVAL, e.g., every 3 seconds. Each multicast receiver responds to this flood by sending a JOIN REPLY packet which is forwarded along the shortest path back to the multicast source that originated the QUERY. Before forwarding this packet, each node waits for JOIN AGGREGATION TIMEOUT, and combines all JOIN REPLYs for the group received during this time into one JOIN REPLY. Each node that forwards the REPLY packet creates (or refreshes) forwarding state for group **G**.

Each node with forwarding state for **G** forward every data packet sent by a multicast source for **G** that it receives. A data packet thus follows the shortest paths to the multicast receivers within the forwarding mesh, though is also forwarded towards other sources for the group who may not be group members. Forwarding state is expired after a multiple of the periodic flooding interval to ensure that in the event that some number of forwarding nodes' multicast state is not refreshed due to packet loss, the forwarding state created from a previous flood would still be valid. This mechanism improves the robustness of the protocol, but may cause multiple overlapping trees to be active in the network simultaneously, each created during a subsequent JOIN QUERY flood [31].

Neighbor Supporting Ad hoc Multicast Routing Protocol (NSMP)

Neighbor Supporting Ad hoc Multicast Routing Protocol (NSMP) [32] is an-other mesh based multicast routing protocol which uses soft states approach. A multicast mesh is constructed using "forwarding nodes", which includes source node, relaying nodes and the multicast group receivers. Those nodes which are adjacent to at least one forwarding node are designated as "neighbor nodes". NSMP also designate source node as group leader as it is responsible for periodic transmission of control messages to maintain the group connectivity. Normally the node with smallest IP is elected as group leader.

Mesh Creation

In NSMP, mesh creation process is source initiated. When a source needs the route to the members of multicast group, it floods the entire wireless net-work with FLOOD_REQ messages. Each intermediate node which broadcasts the FLOOD_REQ message caches its upstream node (from which it receives FLOOD_REQ message) in its routing table. When a group member receives FLOOD_REQ message it send an REP message to its upstream node. Upstream nodes after receiving REP message, adds an entry for the multicast group in its routing table and forwards REP to its upstream node (using the cache entry) and becomes a forwarding node. Mesh creation proc-ess in NSMP is similar to that of ODMRP.

Mesh Maintenance

NSMP differs with ODMRP in its mesh maintenance phase. Unlike ODMRP, source in NSMP does not periodically flood the network with mesh creation messages to ensure the connectivity of mesh nodes. In NSMP scoped broadcasting is used and each source node periodically transmits a LOCAL_REQ message which is relayed only to the mesh nodes or neighbor nodes. REP message are transmitted in mesh maintenance phase in the same way used as are used in mesh creation phase except that forwarding nodes and neighbor nodes are updated as REP is relayed back to the source node. However, this approach of scoped broadcasting does not always re-cover from all link failures. Unrecoverable link failures and group partitioning will be recovered whenever a new source emerges and floods the network with FLOOD_REQ messages.

Group Joining & Leaving

If a node wants to join a multicast group, it sends a REP message in reply of FLOOD_REQ message, but if it is more than two hops away from the mesh nodes, it floods the network with MEM_REQ messages. MEM_REQ messages are forwarded in the same way as FLOOD_REQ message and an REP message is generated whenever it reaches one of the multicast senders or forwarding nodes which is then relayed back to the initiator of MEM_REQ message (the new node who wants to join multicast group). No special packet is required to be transmitted when a node wants to leave the group because of following soft state approach. The node who wants to leave the group just quit sending REP message in response to subsequent group connectivity messages.

Core Assisted Mesh Protocol (CAMP)

Core Assisted Mesh Protocol (CAMP) [33] is a mesh-based multicast routing protocol which uses one or more core nodes to create and maintain multicast mesh. Inspired from the basic architecture used in IP multicast, CAMP uses predefined core nodes which are known to all the nodes in the wireless network. However, these core nodes can leave the group if no node is con-nected to them. It assumes the existence of underlying unicast routing protocol which provides routing information to the mesh nodes. CAMP imposes a restriction on underlying unicast routing protocol such that it must provide correct distance from the known destination in finite amount of time. In the process of mesh creation CAMP ensures that the shortest distance to reach any particular node is included in the multicast mesh. Mesh creation process in CAMP consists of request and reply messages just like ODMRP. Cores are used to limit the control traffic overhead required for receivers to become member of multicast group, however nodes can still join the group even if all the cores becomes unavailable. To ensure that shortest path between each source and receiver is included in the multicast mesh, every entry in the packet forwarding cache is verified periodically. If number of packets coming from a reverse path falls below a certain threshold, a push join or “heart beat” message is sent to all the sources for which this reverse path is being used, thus ensuring that shortest path is always included in the multicast mesh.

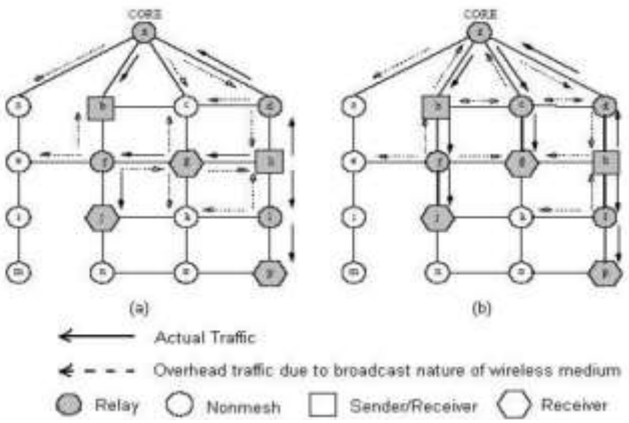


Figure (a) Traffic flow from router h in multicast mesh [33] (b) Traffic flow in equivalent multicast shared tree

ADMR (Adaptive Demand-Driven Multicast Routing Protocol)

ADMR [41] does not employ any periodic control packet exchanges, such as neighbor sensing or periodic flooding, and does not rely on lower layers within the protocol stack to perform such functions; it performs both its route discovery and route maintenance functions on demand, and automatically prunes unneeded multicast forwarding state, and expires its multicast mesh when it detects that the multicast application has become inactive. When there are no multicast sources or receivers for a given multicast group **G**, ADMR does not generate any packet transmissions. If multicast receivers and sources for **G** exist, ADMR creates a source mesh between each multicast sender **S** and the multicast receivers for the group. Source-specific forwarding enables the protocol to support source-specific multicast joins and toroute along shorter paths than protocols that use group-shared forwarding.

Packet forwarding along the ADMR source mesh does not follow any predetermined sequence of hops, but instead each non-duplicate data packet is forwarded by each mesh node, thus following the current shortest-delay paths within the mesh, to the multicast receivers. This type of forwarding increases robustness against packet loss due to collisions or broken links. The multicast sources and receivers in ADMR cooperate to create the multicast source mesh. Each source floods its first data packet for a group, and each receiver responds to that flood with a RECEIVER JOIN packet which sets up forwarding state along the shortest path back towards the source. A flood-response cycle is initiated by each receiver when it first joins the group as well. To resolve partitions, multicast sources may occasionally flood a data packet, e.g., every several tens of seconds. To detect broken links within the mesh, the ADMR routing layer at a multicast source monitors the traffic pattern of the multicast source application, and includes the expected inter-arrival time of future packets in each data packet. Each mesh node records this information upon forwarding a packet, and uses it to detect that it has become disconnected from the mesh. Once a broken link is detected, the node downstream from it (relative to the multicast source) will attempt to perform a local repair using a localized RECONNECT packet flood. Before launching the local repair, the disconnected node sends a REPAIR NOTIFICATION packet downstream, i.e., towards the multicast receivers, in order to inform downstream mesh nodes that it is going to perform the repair and they should cancel their disconnection detection timers; nodes farther away from the source than the node downstream of the broken link, would detect the broken link later, because the disconnection timer incorporates the hop count to the multicast source, which node extract from forwarded data packets. Receiver nodes that receive the REPAIR NOTIFICATION postpone their disconnection timers rather than canceling them. If no data arrives after some time, the receivers assume that the local repair has failed and re-join the group using the request-response cycle invoked during group joins.

When the application is temporarily not sending data, the routing layer generates keep alive packets to enable detection of broken links during this inactive period. If the application does not send packets in significant deviation of its sending pattern, the keep alives stop and the multicast mesh silently expires. These mechanisms allow ADMR to detect broken links without the use of periodic control packet exchanges and to make informed decisions about whether or not it is worth maintaining a multicast mesh based on an application's communication behavior and needs.

PROBLEM STATEMENT

3.1 PROBLEM DESCRIPTION

Tree-based protocols provide high data forwarding efficiency at the expense of low robustness. Their advantage is their simplicity. Their disadvantage is that until the tree is reconstructed after movement of a node, packets possibly have to be dropped. In MAODV, the tree is based on hard state and any link breakages force actions to repair the tree. A multicast group leader maintains up to date multicast tree information by sending periodic group hello messages. Link maintenance is main problem when link to group leader is break.

Mesh-based protocols perform better in high mobility situation as they provide redundant paths from source to destinations while forwarding data packets. However, mesh-based approaches sacrifice multicast efficiency in comparison to tree-based approach.

Problem statement : Mesh based protocol also suffered by creation of loops in the routing path.

3.2 PROPOSED METHODOLOGY

Like MAODV, the multicast AOMDV uses the basic MAODV route construction process. In this case, however, some extensions are made to create multiple loop-free, link-disjoint paths.

The main idea in multicast AOMDV is to compute multiple paths during route discovery. It consists of two components:

A route update rule to establish and maintain multiple loop-free paths at each node.

A distributed protocol to find link-disjoint paths.

3.2.1 Computing Multiple Loop Free Paths

Source S initiate a flood of RREQ packets, an intermediate node A rebroadcasts the RREQ, a neighbor rebroadcasts it, which in turn look at A. It received this RREQ copy to create a reverse path, this will create a loop. In order to eliminate any possibilities of the loop, we maintain a similar as in single path case.

Multicast AOMDV based on a new notion of "advertised hop-count". The advertised hop-count of a node i for a destination d represent the maximum hop-count of a multiple path for d available at i. Maximum hop-count is considered and the advertised hop-count never alter for same sequence number. It allows alternate path with lower hop-count.

When the sequence number is updated the advertised hop-count initialized each time. It is updated as follows:

$$\text{ADV hop-count}_i^d = \max_k\{\text{hopcount}_k/(\text{nexthop}, \text{hopcount}_k) \in \text{route\_list}_i^d\}$$

Multicast AOMDV Routes Update Algorithm

A node i receives a route advertisement to a destination d from a neighbor j. the variable $seqnum_i^d$, $ADV\_hopcount_i^d$ and $route\_list_i^d$, represent the sequence number, ADV_hopcount and route_list for destination d at node i repectively.

```
If ($seqnum_i^d$ <$seqnum_j^d$)
Seqnum$_i^d$=seqnum$_j^d$
If (i≠d) then
ADV_hopcount$_i^d$=NULL
Insert(j, ADV_hopcount$_j^d$+1) into route_list$_i^d$
else
ADV_hopcount$_i^d$=0
endif
else (seqnum$_i^d$=seqnum$_j^d$) and
((ADV_hopcount$_i^d$, i)>(ADV_hopcount$_j^d$,j))
then
insert(j, ADV_hopcount$_j^d$+1) into route_list$_i^d$
```

- independently .
- There are two types disjoint paths:

a) Node-disjoint

b) Link-disjoint

Node-disjoint path does not have any common node, except source and destination. In contrast, link-disjoint does not have any link in common. Link disjoin path may have common nodes. Link-disjoint is more effective than node-disjoint because later has less disjoint path from source to destination.

Link-disjoint path computation:

S, I and D represents source, intermediate and destination node repectively.si_1 and si_2 node-disjoint path from S to I and id_1 and id_2 are node-disjoint paths from I to D. There are two possible set of link-disjoint path from S to D, si_1 - $id_{1,}$ si_2 - id_2 or si_1- $id_{2,}$ si_2 - id_1.

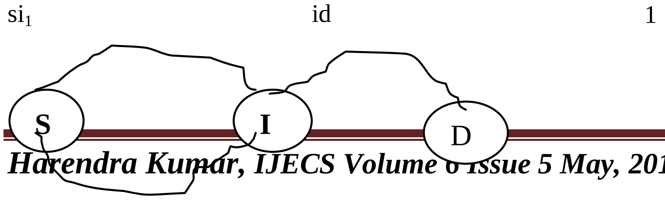


Figure 3.1 link-disjoint paths computation

Node-disjoint path computation:

A, B and C are neighbors of Source node S, floods a packet. J transmits only first arrived copy of the packet either from A or B. Two copies arrived at I one via A or B and other via C. Thus have two node-disjoint paths to S.

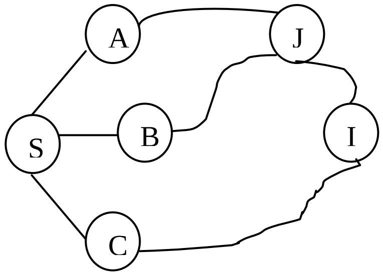


Figure 3.2 Node-disjoint paths computation

In multicast AOMDV each RREQ, RREP arriving at a node potentially defines an alternate path to the source or destination. Just accepting all such copies will lead to the formation of routing loops. In order to eliminate any possibility of loops, the "advertised hop-count" is introduced.. The protocol only accepts alternate routes with hop-count lower than the advertised hop-count, alternate routes with higher or the same hop-count are discarded.

- The advertised hop-count mechanism establishes multiple loop-free paths at every node. These paths still need to be disjoint.
- In multicast AOMDV this is used at the intermediate nodes. Duplicate copies of a RREQ are not immediately discarded. Each packet is examined to see if it provides a node-disjoint path to the source. For node-disjoint paths all RREQs need to arrive via different neighbors of the source. This is verified with the first hop field in the RREQ packet and the first hop list for the RREQ packets at the node.
- At the destination a slightly different approach is used, the paths determined there are link-disjoint, not node-disjoint. In order to do this, the destination replies up to k copies of the RREQ, regardless of the first hops. The RREQs only need to arrive via unique neighbors.

Group Creation:

Follows directly from the unicast AOMDV.

2 Discovers multicast routes on-demand using a broadcast route discovery mechanism employing the same Route Request (RREQ) and Route Reply (RREP) messages.

3 A mobile node originates a RREQ message when it wishes to join a multicast group, or when it has data to send to a multicast group but it does not have a route to that group.

4 Only a member of the desired multicast group may respond to a join RREQ.

5 If the RREQ is not a join request, any node with a fresh enough route to the multicast group may respond.

6 As the RREQ is broadcasted across the network, nodes set up pointers to establish the reverse route in their routing tables.

7 A node receiving a RREQ first, updates its route table to record the sequence number.

8 For join RREQs, an additional entry is added to the multicast route table and is not activated unless the route is selected to be a part of the multicast mesh.

Group Maintenance

Group membership and multicast mesh is maintained in multicast AOMDV though periodic transmission of RREQ messages by the source node. All the states in multicast AOMDV are soft states which are refreshed periodically by control message mentioned above or via transmission of data packets. Because of having soft states between the nodes, there is no need to send any special message to the member or to the neighbor nodes. If any member wants to leave a multicast group, it simply stops the generation of RREP message in reply of RREQ message by the source node

3.3 PERFORMANCE OF PROPOSED PROTOCOL

3.3.1 NS Simulation

Simulation environment is as follows:

1. Radio range of a node: 250m
2. Channel capacity: 2MB/sec
3. Data rate: 1 packet/sec
4. Smulation area 200m×200m
5. Traffic pattern: 50 CBR/UDP (bidirectional communication)
6. Maximum speed: 5m/sec, 10m/sec
7. Size of data packet: 512bytes
8. Number of node: 7
9. Simulation time: 200sec.

Ad hoc on demand multicast routing protocol constructs multiple routes from source to destination whenever there is a packet to transmit from source to destination. The packet transmission follows only that route which has minimal cost (minimum number of nodes in between a source to destination).

The entire topology has been divided in three groups G1, G2, G3. A node can receive or transmit packet on when it is the member of a group. For G1 node 5 is working as router. And for G2 node 3 is working as router. And for G3 node 6 is working as router.

Simulation starts at time t=10.5.

At t= 24, node 6 joins group G1. At this time node 5 updates it's routing table and constructs a route from 5 to 6 via node 4.

At time t=25 node 6 leaves group G1. And hence there is no transmission from 5 to 6.

At t=29 node 6 joins group G1, and router 5 again transmit packets to node 6 via node4. At t=40, node 6 moved out.

At t=40.5 node 6 moved-in.

At t=70 node 2 joins group G1. Router 5 forwards packet to node 2 via route 5-4-6-2.

At t=78 node 2 leaves group. At t=80.5 node 2 again join group G1, and router forwards packet to it through route 5-4-6-2.

At time t=84 node 2 leaves group G1.

At t=84.5 node2 join group G1.

At t=88 node 2 leave group G1.

At t=89 node 2 joins group G2. At this time router 3 starts transmission to node 2.

At t=90 node 2 joins group G1, at this time router 5 and router 3 both transmit packets to node 2.

At t=93 node 4 joins group G3. For G3 node 6 is working as router and transmit the packet to node 4.

At t=96.5 node 2 joins group G3. Router 6 omputes a route for transmitting packets to node 2 via node 4.

At t= 102 node 2 leaves group G3.

At t= 102 node 4 leaves group G3.

At t=106 node 2 leaves group G1.

At t=110.5 node 2 joins group G1

At t=112.5 node 2 leave group G1

and node 4 join group G3.

At t=115.5 node 3 join group G3, and router 6 updates it's routing table and reconstruct a router to node 3 via node 4 as 6-4-3.

At t=118.5 node 6 leave group G1.

At t=121 node 6 join group G1.

At t=123 node 6 leave group G1.

At t=124 node 4 leave group G1.

At t=128 node 4 join group G1.

At t=136 node 3 and 4 leave group G3.

At t=138.5 node 2 join group G3, router 6 recomputed route from 6 to 2 as 6-4-2.

At t=140.5 node 4 leave group G3, and node 2 join group G1.

At t= 157 node 2 leave group G1.

At t= 163.5 node 2 join group G1.

At t= 167.5 node 2 leaves group G1.

At t= 167.5 node 4 join group G3.

At t= 170.5 node 4 leaves group G3.

At t= 170.5 node 2 join group G1.

At t= 179 node 4 leaves group G3.

At t= 187 node 6 leaves group G1.

At t= 188 node 6 join group G1.

At t= 195.5 node 4 join group G3.

At t= 195.5 node 2 leaves group G1.

At t= 198 node 3 join group G3.

At t= 199 node 3 leaves group G3.

At t= 200 Simulation END

3.3.2 Node Structure

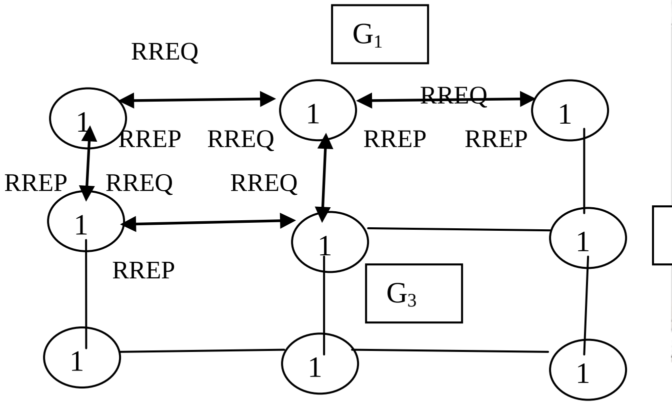


Figure 3.3 Working of multicast MAODV

RESULT

4.1 PERFORMANCE METRIC

We evaluate two key performance matrices:

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

Time: This is a total simulation time.

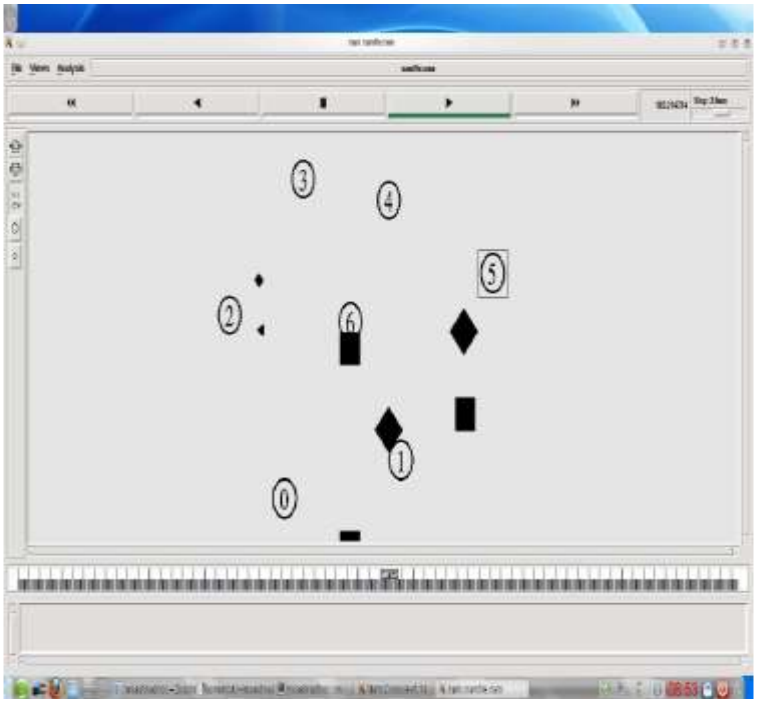


Figure 4.1 Packet transmissions in NS-2

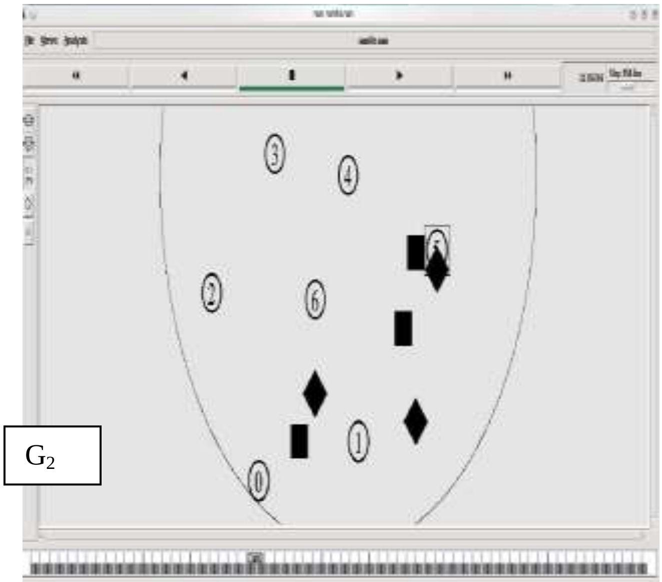
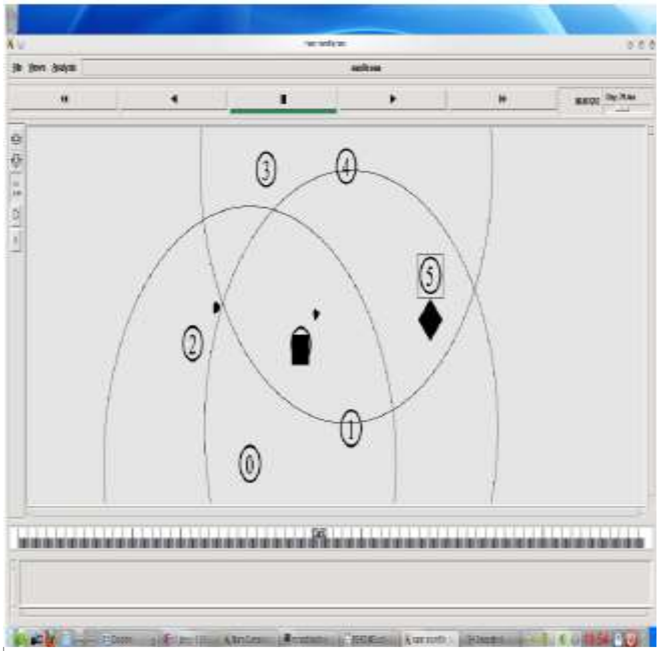


Figure 4.2

Packet transmissions in NS-2



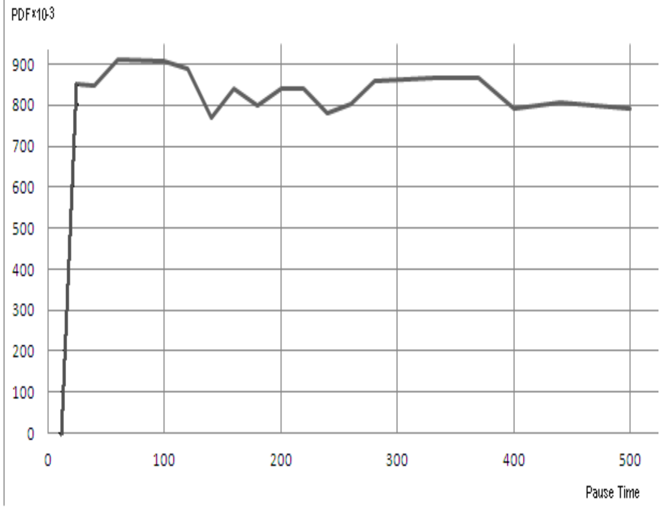


Figure 4.3 Packet transmissions in NS-2

Packet Delivery Ratio

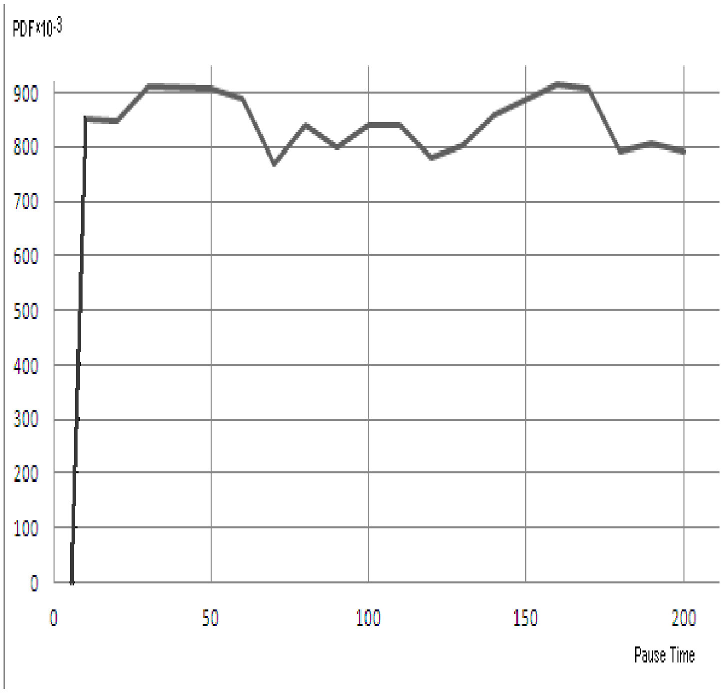


Figure 4.4 Graphical representation of result

Figure 4.5 Graphical representation of result

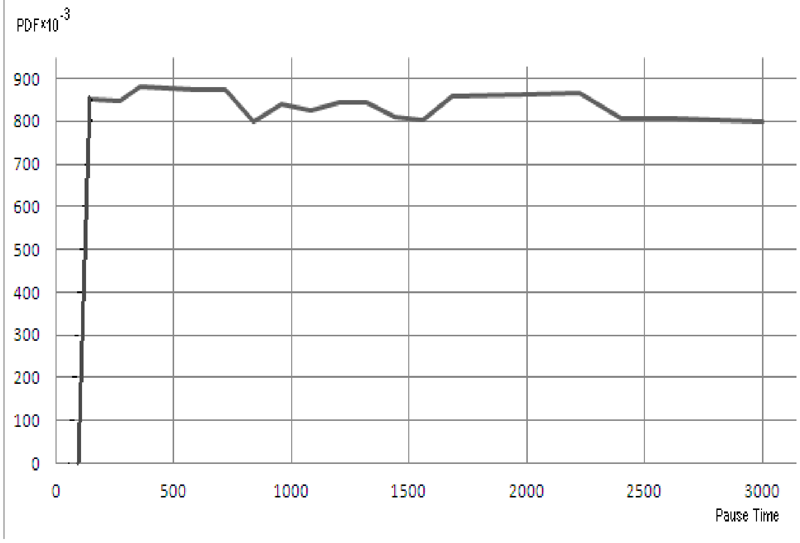


Figure 4.6 Graphical representation of result

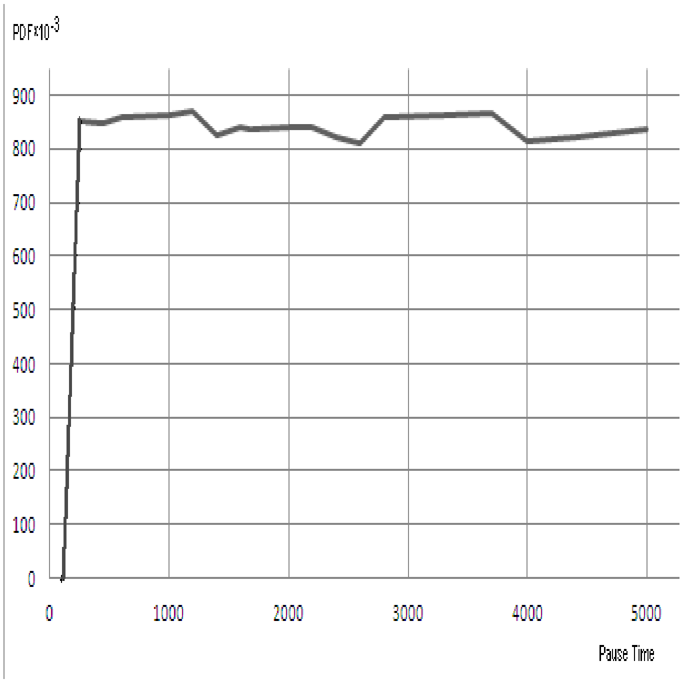


Figure 4.7 Graphical representation of result

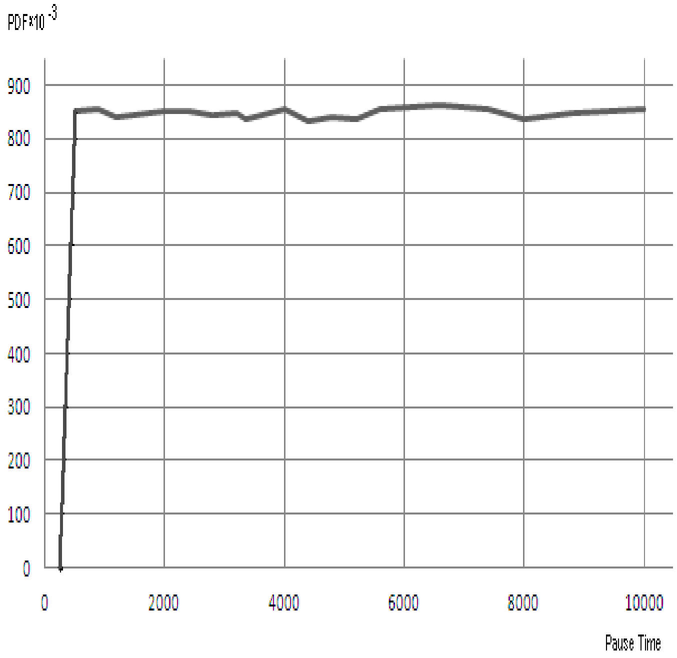


Figure 4.8 Graphical representation of result

REFERENCES

[1] http://tools.ietf.org/html/rfc2201

[37] C. K. Toh, "Ad Hoc Mobile Wireless Networks: Protocols and Systems," *Prentice Hall Englewood Cliff*, NJ 07632,2002. [36] C. Murthy and B. Manoj, "Ad hoc Wireless Networks: Architectures and Protocols," *Prentice Hall PTR*, 2005.

[27] C. K. Toh, "Ad Hoc Mobile Wireless Networks: Protocols and Systems," *Prentice Hall Englewood Cliff*, NJ 07632,2002. [36] C. Murthy and B. Manoj, "Ad hoc Wireless Networks: Architectures and Protocols," *Prentice Hall PTR*, 2005

[2]http://tools.ietf.org/html/draft-ietf-idmr-dvmrp-v3-11

[]http://technet.microsoft.com/en-us/library/bb742462.aspx

[3]R. G. T. Anderson, B. Bershad, and D. Wetherall, "A System Architecture for Pervasive Computing," *In Proc. 9th ACM SIGOPS European Workshop*, pp. 177–182, Kolding, Denmark, September 2000.

[4] I. Chlamtac and J. Redi, "Mobile Computing: Challenges and Opportunities," *In Encyclopedia of Computer Science, 4th Edn,*D. Hemmendinger, A. Ralston, and E. Reilly (Eds), International Thomson Publishing, 1998.

[5]C. K. Toh, "Ad Hoc Mobile Wireless Networks: Protocols and Systems," *Prentice Hall Englewood Cliff*, NJ 07632,2002. [36] C. Murthy and B. Manoj, "Ad hoc Wireless Networks: Architectures and Protocols," *Prentice Hall PTR*, 2005.

[6] David B. Johnson, "Validation of Wireless and Mobile Network Models and Simulation," *In Proc. DARPA/NIST Workshop on Validation of Large- Scale Network Models and Simulation* Fairfax, VA, May 1999.

[7] A. Tenenbaum, "Computer Networks," *PH PTR*, 2003.

[8]Stefano Basagni, Marco Conti, Silvia Giordano, Ivan Stojmenovic (Eds.), "Mobile Ad Hoc Networking," *IEEE Press*, Wiley Interscience. 2004.

[9] T. S. Rappaport, "Wireless Communications: Principles and Practice, 2/E", *Prentice Hall PTR*, 2002.

[10]I. Chlamtac, M. Conti, and J. Liu, "Mobile Ad Hoc Networking: Imperatives and Challenges," *J. Ad Hoc Networks*, vol. 1, no. 1, pages13 – 64, 2003.

[11] Bdale Garbee, "Thoughts on the Issues of Address Resolution and Routing in Amateur Packet Radio TCP/IP Networks," *In Proc. ARRL Amateur Radio 6th Computer Networking Conference*, pages 56–58. American Radio Relay League, August 1987.

[12]Daniel M. Frank, "Transmission of IP Datagrams over NET/ROM Networks," *In ARRL Amateur Radio 7th Computer Networking Conference*, pages 65–70. American Radio Relay League, October 1988.

[13] Ivan Howitt, Wayne Manges, Teja Kuruganti, Glenn Allgood, Jose Gutierrez, James M. Conrad, "Wireless Industrial Sensor Networks: Framework for QoS Assessment and QoS Management," *Transactions of the Instrumentation, Systems, and Automation Society (ISA)*, 45(3), July 2006.

[14] M. Weiser, "The Computer for the Twenty-First Century," *Scientific American*, pages 94–100, September 1991.

[15] C. K. Toh, "A Novel Distributed Routing Protocol to Support Ad-Hoc Mobile Computing," *In Proc. of the 1996 IEEE Fifteenth Annual International Phoenix Conference on Computers and Communications*, pages 480–486, March 1996.

[16] C. K. Toh, "Ad Hoc Mobile Wireless Networks: Protocols and Systems," *Prentice Hall Englewood Cliff*, NJ 07632,2002. [36] C. Murthy

and B. Manoj, “Ad hoc Wireless Networks: Architectures and Protocols,” *Prentice Hall PTR*, 2005.

[17] Carlos Cordeiro and Dharma P Agrawal, “Adhoc and Sensor Networks, Theory and Applications, ” *World Scientific Publishing*, 2006.

[18] S Buchegger & J-Y Le Boudec, “Nodes Bearing Grudges: Towards Security, Fairness and Robustness in Mobile Ad hoc Networks,” *In Proc. 10th IEEE Euromicro Workshop on Parallel, Distributed and Network based Processing,* pages 403-410, January 2002.

[19] Royer E. and Perkins C., “Multicast Using Ad Hoc on Demand Distance Vector Routing,”in

Proceedings of the ACM Mobicom’99,pp.207-218,1999.

[20] S.J. Lee, M. Gerla, C.C. Chiang, “On Demand Multicast Routing Protocol”, Proceedings of IEEE WCNC’99, New Orleans, pages 1298-1302, Sept 1999.

[21] R. Pandi Selvam and V. Palanisamy “A Performance Comparison of MAODV and ODMRP Routing Protocols in Mobile Ad Hoc Networks”International Journal of Research and Reviews in Ad hoc Networks (IJRRAN) Vol. 1, No. 3, September 2011, ISSN: 2046-5106.

[23] M. Liu, R. Talpade, A. McAuley, and E. Bommaiah, “AMRoute: Ad-hoc multicast routing protocol”, Technical Report, CSHCN T. R. 99-1, University of Maryland.

[24] Lusheng Ji and M. Scott Corson “Differential Destination Multicast–A MANET

Multicast Routing Protocol for Small Groups” 0-7803-7016-3/01/$10.00 ©2001 IEEE.

[25] J. LUO, P. EUGSTER, ANDJ. HUBAUX,” Progressively Adapted Sub-Tree in Dynamic Mesh” in Proc.of INFOCOM 2003, 2003.

[26] C.W. Wu, Y.C. Tay, “AMRIS: a multicast protocol for ad hoc wireless networks”, Proceedings IEEE MILCOM’99, Atlantic City, Nov. 1999.

[27] Perkins, C., Belding-Royer, E., and Das, S., “Ad hoc On-Demand Distance Vector (AODV) Routing,” rfc3561.txt (2003).

[28] Joshua Auerbach, Marc Kaplan, Shay Kutten, “Multicast Group Membership Management” IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 11, NO. 1, FEBRUARY 2003.

[29] Luo Junhai, Xue Liu, Ye Danxia “Research on multicast routing protocols for mobile

ad-hoc networks” ScienceDirect Computer Networks 52 (2008) 988–997.

[30] S. LEE, W. SU, J. HSU, M. GERLA, AND R. BAGRODIA, A performance comparison study of ad hoc wireless multicast protocols, in Proc. of INFOCOM 2000, 2000, pp. 565–574.

[31] Stefan Pleisch, Mahesh Balakrishnan, Ken Birman, Robbert van Renesse” Efficient Flooding in Mobile Ad-hoc Networks” MobiHoc’06,May 22–25, 2006, Florence, Italy,Copyright 2006 ACM 1-59593-368-9/06/0005.

[32] Seungjoon Lee,Chongkwon Kim, “Neighbor Supporting Adhoc Multicast Routing Protocol” Brain Korea21 Projectan dgrantNo.(1999-1-302-005-3).

[33] J. J. Garcia-Luna-Aceves and E.L. Madruga, “The core assisted mesh protocol,” IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, vol. 17, no. 8, pp. 1380–1394, August 1999.

[34] Ching-Chuan Chiang, Mario Gerla and Lixia Zhang” Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks” CLUSTER COMPUTING (1998).

[35] Lusheng Ji, M.Scott Corson “A Lightweight Adaptive multicast algorithm” U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-2002.

[36] Mario Gerla, Xiaoyan Hong, and Guangyu Pei, “Fisheye State Routing Protocol (FSR) for Ad Hoc Networks”, draft-ietf-manet-fsr-3.txt, June 2002.

[37] A.B.Mnaouer, L. Chen, C. H. Foh, and J.W.Tantra, “OPHMR: An Optimized Polymorphic Hybrid Multicast Routing for MANETs”, IEEE transaction on mobile computing. Vol 5, No. 6, May 2007.

[38] J.J. Garcia-Luna-Aceves, E.L. Madruga, “A Multicast Routing Protocol for Ad-Hoc Networks”, Proceedings of IEEE INFOCOM’99, New York, pages 784-792, March 1999.