# Lock and Forward Hierarchical Routing Algorithm in MANET

Vikrant Verma, Dr. Manoj Kumar

**Abstract:** ***Routing in Ad-hoc mobile network can be typical process because it is a dynamic environment. Nodes can join and leave a cluster dynamically. Traditional routing algorithm cannot be successful due to various reasons. A new "lock and forward hierarchical routing algorithm in MANET" is proposed in this paper in which so many unit clusters come together and form a logical tree structure for forwarding messages among them and also to nearby fixed cell structures. A route discovery process is initiated by a node which wants to send such message to destination. This can be done through multicast or broadcast. If finally by either method a message finds its final destination node, then it starts backward on the same path to source. This message also does a reverse tracing because it follows the same path back and locks each node in its path on its way back. Locking period is 4 times of the total traversal period from source to destination. Then finally message is routed on this locked path. Because we lock all the nodes on the way so we call it lock and forward hierarchical routing. It is hierarchical because all the cluster head make a logical tree structure in their's buffer by storing some details in routing table. This locked path can be dissolved if a node thinks that this is very much necessary for him to move to a new location. There can also be unguarnteed routing without locking the nodes on its path. This is similar to UDP protocol in TCP/IP environment and can be done by multicast or broadcast. Security enhancements can be done through Authentication and identity check of a node. Each node will provide its MAC address / IP address, digital signature and CH certificate for identification and authentication.***

**Keywords:** MANET

## 1. Introduction and Literature Review

There are three basic routing algorithms. **Dijkstra's Algorithm**[2] is shortest path finding algorithm. It maintains two sets of vertex i.e. unvisited and visited nodes and find shortest path by recursively adding and comparing the shortest path. **Prims' Algorithm**[3] works on finding the minimum spanning tree by traversing nearby nodes. It forms a minimum spanning tree by adding nodes of minimum weight to spanning tree which does not form cycle. **Kruskal's Algorithm**[4] also finds the shortest path by forming minimum spanning tree. It first sorts all the edges of graph according to its weight and than chooses directly the minimum weight edges and thus forms a minimum spanning tree.

**Traditional routing algorithms** are based on basic algorithms. In **Distance Vector (DV)**[5] each node maintains a table giving the distance from itself to all possible destinations. Periodically every node broadcasts update packets to each of the neighbors. Bellman-Ford algorithm is used for finding the shortest path to determine the correct next hop of its neighbors. When presented a packet for forwarding to some destination, each router simply forwards the packet to the correct next hop router. The main problem of Distance Vector routing is route looping and count to infinity. In **Link State (LS) Algorithm**[5] each node maintains a view of the network topology with a cost for each link. Each node periodically broadcasts the cost of its outing links to all other neighboring nodes. It uses shortest-path algorithm to choose its next hop for each destination. For **Example Bellman-Ford algorithm**[6] works on finding the shortest path with the correct next hop of its neighbors. When presented a packet for forwarding to some destination, each router simply forwards the packet to the correct next hop router. There are few problems of Bellman-Ford Algorithm i.e. route looping and count to infinity. Dijkstra's Algorithm can then be used for the shortest path.

If we use traditional routing algorithms for MANET we get few problems associated with them which are worth mentioning here. One of these problems is **Dynamism of the topology, frequent changes of connections**, connection quality and number of participants may be the reason for sudden change in topology**.** Another problem may be **limited performance of mobile systems**. The periodic updates of routing tables need energy without contributing to the transmission of user data. Sleep modes are quite difficult to realize. The Limited bandwidth of the system is reduced even more due to the exchange of routing information. Connection in wireless network may not be symmetric hence forming an Asymmetric connection.

Current Routing Protocols in Ad hoc Networks can be classified into two major categories i.e. **Table Driven and Source Initiated on demand**. **Table-driven** are proactive. These continuously evaluate the routes attempt to maintain consistent, up-to-date routing information. **Source-Initiated On-Demand Routing Protocol** can be called reactive protocols. These create routes only when it is desired by the source node. There may be a problem of longer delays. Sometimes a route may not be ready for use immediately when data packets come for routing.

In **Table-Driven Routing Protocols, Destination-Sequenced Distance-Vector Routing (DSDV)**[7] protocol is a proactive table driven algorithm based on Bellman-Ford routing. In proactive protocols, all nodes learn the network topology before a forward request comes. In DSDV protocol each node maintains routing information for all known destinations. The routing information is updated periodically. Each node maintains a table, which contains information for all available destinations, the next node to reach the destination, number of hops to reach the destination and sequence number. The nodes periodically send this table to all neighbors to maintain the topology, which adds to the network overhead. Each entry in the routing table is marked with a sequence number assigned by the destination node.

In **Wireless Routing Protocol (WRP)**[8] there is a quite complicated table structure. Each node maintains four different tables as in many other table-driven protocols only two tables are needed. These four tables are: 1) distance table, 2) routing table, 3) link cost table and 4) message retransmission list (MRL) table. In WRP nodes exchange routing-table update messages only from a node to its neighbors. An update message contains such components as an update list. An update list entry specifies a destination, a distance to the destination and a predecessor to the destination. This all complex re-computation forms overhead on processors.

**Temporally ordered routing algorithm (TORA)**[9] is a highly adaptive, efficient and scalable distributed routing algorithm based on the concept of link reversal. TORA is proposed for highly dynamic mobile, multi-hop wireless networks. It is a source-initiated on-demand routing protocol. It has a unique feature of maintaining multiple routes to the destination so that topological changes do not require any reaction at all. The protocol reacts only when all routes to the destination are lost. In the event of network partitions the protocol is able to detect the partition and erase all invalid routes. The protocol has three basic functions i.e. Route Creation, Route Maintenance Route Erasure. All the features used in this protocol are best for Ad-hoc environments and can be adapted.

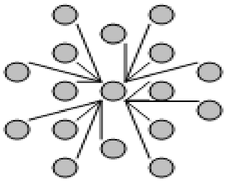
**Associativity-Based routing (ABR)**[10] this protocol does not attempt to consistently maintain routing information in every node. In an *ad-hoc* mobile network where mobile hosts (MHs) are acting as routers and where routes are made inconsistent by MHs' movement, we employ an Associativity-based routing scheme where a route is selected based on nodes having Associativity states that imply periods of stability. The association property also allows the integration of *ad-hoc* routing into a BS-oriented Wireless LAN (WLAN) environment, providing the fault tolerance in times of base stations (BSs) failures. The protocol is free from loops, deadlock and packet duplicates and has scalable memory requirements. Simulation results obtained reveal that shorter and better routes can be discovered during route re-constructions. All the characteristics of this protocol are adaptive and well suited. Associativity-based routing is main feature and can be adopted.

**Ad-hoc On-Demand Distance Vector Routing (AODV)**[11] is improved over DSDV algorithm (minimize the number of required broadcast by creating routes on a demand basis. It maintains a complete list of routes as in the DSDV algorithm. Nodes (not on a selected path) don't maintain routing information or participate in routing table exchanges. Instead of source routing, AODV relies on dynamically establishing route table entries at intermediate node. AODV use the concept of destination sequence number from DSDV. AODV only supports the use of symmetric links. AODV perform path discovery procedure using RREQ/RREP query cycles. It uses **reverse Path setup and** Forward path setup for route discovery. Reverse and forward path setup is important concept and we can adopt it for our new protocol.

**In Dynamic Source Routing (DSR)**[12] Each host maintains a route cache which contains all routes it has learnt. In source routing routes are denoted with complete information (each hop is registered). There are two major parts of source routing i.e. route discovery and route maintenance. When a host has a packet to send, it first consults its route cache. If there is an unexpired route, then it will use it otherwise, a route discovery will be performed. In route discovery there is a "route record" field in the packet. The source node will add its address to the record. On receipt of the packet, a host will add its address to the "route record" and rebroadcast the packet. Route record is one of the best concepts for route discovery and can be adopted for new protocol.

## 2. Proposed Work and Mechanism

After cluster formations through **Analyzing Secure Clustering & Maintenance (SCM) Algorithm in Mobile Ad-Hoc Network (MANET)**[1] there are three kinds of mobile stations available in an established cluster i.e. cluster members (CM), cluster heads (CH) and cluster gateways (CG). Here we are forming a unit cluster for simplicity as each member is available at one hop distance from cluster head (CH). In the same fashion in nearby environment there will be more clusters. If we consider a full cluster including its all three kinds of mobile stations as a single entity and hide its internal working for abstraction. In this fashion routing becomes an easy process in MANET environment.



Unit Cluster

After cluster formation each CH will initiate discovery process for routing purpose. While forming the cluster it already had made a list of its members and their roles. Cluster gateways will be working as routing agents to other and nearby clusters.

As soon as a gateway comes in contact to any nearby cluster, it exchanges the table of its members with it. It can only be done through mutual agreement. If two clusters are nearby in their location then it becomes possible to exchange data between them.

These nearby clusters which act as unit clusters than start forming a tree structure with the address of all the cluster heads and address of their members. The tables and the information about environment are mutually exchanged. These exchanges are performed once initially and later on at every update or change in the routing environment.

Routing in MANET can be classified as below.

1. **Within a one hop cluster:**
2. **outside one hop cluster to nearby clusters**

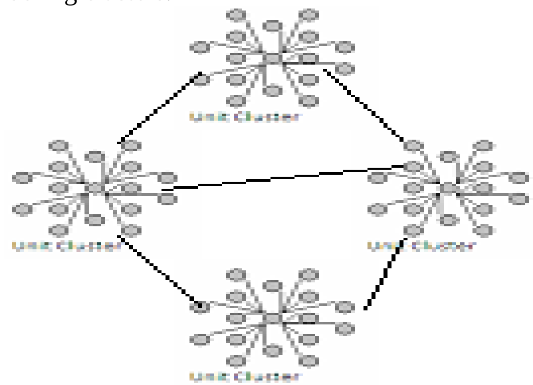
**Within a one hop cluster:**

After formation of unit cluster by **SCM Algorithm**[1] the Cluster Head (CH) is having a table of its all the members. Each member is accessible either through Cluster Head (CH) or directly if they are at accessible distance. If a member is accessible directly then there is no point of putting extra burden of routing on CH. That means there may be three kinds of routing within a cluster. First one through Cluster Head (CH) if not directly accessible, second one direct routing without intervention of CH (only inform to CH) and third one is alternatively CH uses its half of the capability for routing purpose, but if it crosses its limits than CH can choose a member to probably more capable one (Ist in the list of claimers for CH) to assist him in routing. This member will be working on behalf of CH and will be called as Co-CH of cluster.

A node calculates a nearby node by its radio range (RF). When a node sense a powerful signal in its radio range and also find this node in its own cluster than it declares this as directly accessible and mark through CH entry in table as False or '0'. After this whenever a node has to access this particular node it will send data directly to this node and in turn also send a message to Cluster Head (CH). A receiving node also sends acknowledgement to sending node as well as inform CH. What would CH do with this information? CH will store this information in his table for keeping abrupt the further routing information.

| ID | Weight | ROLE | CH-ID | Low resource | Trough CH |
|---|---|---|---|---|---|
| 192.168.2.1.0011010..<br>(Initiating node) | 9 | 1 | 192.168.2.1.0011010.. | 0 | 0 |
| 196.160.2.13.100011.. | 8 | 0 | 192.168.2.1.0011010.. | 0 | 1 |
| 192.120.22.15.110011.. | 7 | 0 | 192.168.2.1.0011010.. | 0 | 1 |
| 202.160.200.13.101010. | 6 | 2 | 192.168.2.1.0011010.. | 0 | 1 |
| .168.2.1. | 5 | 2 | 192.168.2.1.0011010.. | 0 | 0 |
| 192.168.2.1. | 7 | 2 | 192.168.2.1.0011010.. | 0 | 0 |

**Outside one hop cluster to nearby clusters:** when it is required to transfer the data to the nearby clusters. First there has to be a search process for the destination node. For the search process first the destination is searched within the cluster and for that a source node need to search its table. Source node can also judge from table itself that the destination is accessible directly or through CH. If destination node is not available inside cluster than sender node will ask the CH node for the information of nearby clusters. CH and Gateways will maintain tables of neighboring clusters.



Multiple Unit Clusters within one environment

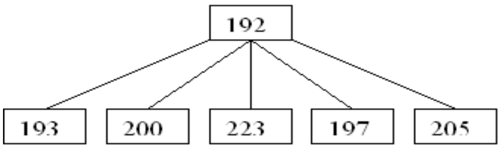
If number of clusters is present in the nearby environment then their gateways will start exchanging information regarding proximity of other clusters nearby. They shall form a logical tree structure by storing information in their catch. Information might be stored logically in a table as structure below. Main information required to store may be its MAC address, its original IP address and IP address allotted to it by the Cluster Head (CH) and the address of gateway through which these are connected.

**Table of CH 192.168.1.1 (MAC address c4:56:fe:77:12:a2)**
**Making a Tree structure in memory**

| IP of Neraby Cluster CH | Connected through Gateway | Original IP | MAC Address | Logical IP Given By cluster |
|---|---|---|---|---|
| 193.100.0.1 | 192.168.1.40 | 196.162.1.4<br>224.200.21.22<br>126.123.34.2<br>145.90.45.21<br>................. | 58:12:43:50:71:f5<br>00:1e:65:85:75:84<br>40:16:7e:50:6e:96<br>af:78:13:1a:92<br>.................... | 193.100.0.1<br>193.100.0.2<br>193.100.0.3<br>193.100.0.4<br>.................. |
| 200.110.2.10 | 200.55.3.4 | 200.162.1.4<br>222.20.21.22<br>129.12.34.2<br>144.22.45.21<br>................. | 00:73:8d:79:3d:44<br>48:d2:24:6e:da:ff<br>C4:62:ea:f5:ec:06<br>10:3b:59:0c:91:66<br>.................... | 200.110.2.10<br>200.110.2.11<br>200.110.2.12<br>200.110.2.13<br>................. |
| 223.170.1.1 | 223.170.1.10 | 20.162.1.4<br>221.100.21.22<br>128.123.34.2<br>150.90.4.21<br>.................. | 00:1d:7d:5e:b2:4b<br>28:fb:d3:3e:07:32<br>D4:93:98:Ce:79:b1<br>08 :$3^e$ :$8^e$ :2c :a5 :f9<br>.......................... | 223.170.1.1<br>223.170.1.2<br>223.170.1.3<br>223.170.1.1<br>.................. |
| 197.23.12.11 | 197.23.12.20 | 96.142.1.4<br>24.210.21.22<br>146.12.34.2<br>185.10.45.21<br>................ | d0:27:88:59:d9 :27<br>9c :d9 :17 :37 :$7^e$ :29<br>A4:81:ee:98:9a:04<br>78 :4b :87 :ca :5b :d3<br>........................ | 197.23.12.11<br>197.23.12.12<br>197.23.12.13<br>197.23.12.14<br>.................. |
| 205.121.23.45 | 205.121.23.14 | 206.12.1.4<br>241.100.21.22<br>125.13.34.2<br>149.91.45.21<br>................ | 78 :4b :87 :ca :5b :d3<br>88 :a7 :3c :d2 :ee :a3<br>00:61:71:2a:10:98<br>Dc:f1:10:82:2b:ae<br>..................... | 205.121.23.45<br>205.121.23.46<br>205.121.23.47<br>205.121.23.48<br>................ |

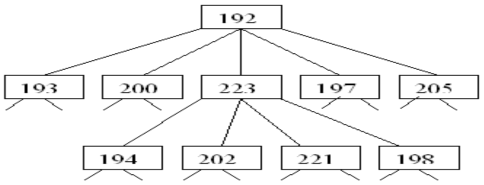
By forming the above table in memory a cluster environment will start forming the logical tree structure as below.

Logical Tree Formed



If there are so many clusters in the nearby environment then network of cluster heads (CHs) will form a logical tree at multiple levels, hence routing would become easier and straight forward.

Logical Tree Formation at level-1



**Route Discovery process:** If one mobile station in MANET wants to send a message to some destination. It must either have an unexpired route to destination. If it is so than it will simply forward the message through this route or if it does not have a route than route discovery process has to begin.

For the route discovery process there can be multicast or broadcast methods. If a cluster member wants to send a message than it has to inform its own CH for route discovery. Once again if it has the route already than there is no need of discovering it otherwise CH will start discovery process on its behalf. In another case if CH itself wants to send a message to a destination and it does not have the route, it would start the same process of route discovery in the same way.

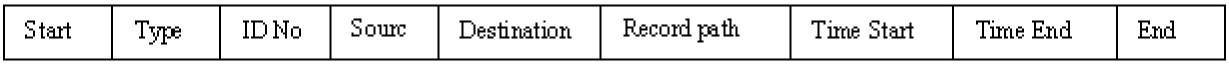
For the route discovery first of all CH will send a discovery packet to it's all the gateways. It is guaranteed that gateways would not have the routes because status of update information of CH and gateways will always be the same. In turn gateways will forward this message to gateways of nearby clusters. These gateways would inform to their CH for the same and if they have destination than the process will terminate here only. But if they do not have destination than each gateway will forward this message to CH and CH will inform to other gateways to forward it further, so that the message propagates quickly globally and will be terminated if destination is found.

Each such discovery packet will have a unique number with day/time stamp put into it while originating. By this stamp duplicate packets can be recognized.

If a particular gateway / CH see a duplicate discovery packet it will just drop it. This will eliminate problem of duplicate and delayed packets. Multiple routes also are eliminated due to multicast packet will find the shortest path automatically. A discovery packet which is coming from longer route and reaching late at a particular destination will be just dropped because it will be a duplicate packet.

In the global discovery process existing mobile cell structure can also be used if it is available nearby to some cluster. This will be called as an amphibian environment because cluster and cell of mobile phone are working together. There has to be a maximum limit of hops also for a discovery packet. Otherwise it will roam endlessly in the network and may create problem of count to infinity. Such a limit may be between 100 and 1000 according to assumptions that how far this destination might be available.

Every cluster member whichever falls in its path will suffix its address into the packet's 'Record path' field and also copy path form packet to its table in buffer of present node. This way the network will also update itself while discovering a destination.

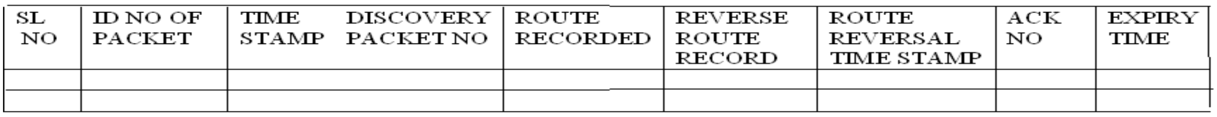


| Start | Type | ID No | Sourc | Destination | Record path | Time Start | Time End | End |
|---|---|---|---|---|---|---|---|---|

FORMAT OF ROUTE DISCOVERY PACKET

Format of path recording table at each node may be as below

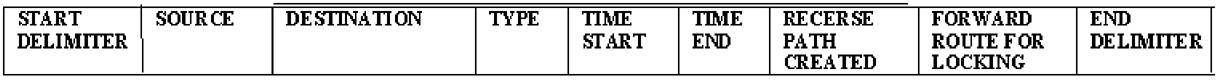
FORMAT OF PATH RECORDING TABLE AT EACH NODE



| SL NO | ID NO OF PACKET | TIME STAMP DISCOVERY PACKET NO | ROUTE RECORDED | REVERSE ROUTE RECORD | ROUTE REVERSAL TIME STAMP | ACK NO | EXPIRY TIME |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| | | | | | | | |

If a discovery packet reaches to its destination after multicasting than destination will send it back in the form of ACK (acknowledgement) by the same path. It will follow the same path back to its origin and again the same process of affixing address by each gateway/CH/member will be followed. In this way two way paths to and fro will be confirmed.

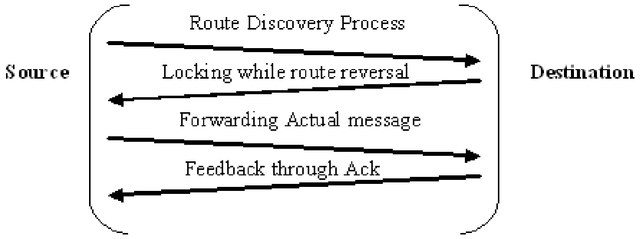
FORMAT OF ACKNOWLEDGEMENT PACKET



| START DELIMITER | SOURCE | DESTINATION | TYPE | TIME START | TIME END | RECERSE PATH CREATED | FORWARD ROUTE FOR LOCKING | END DELIMITER |
|---|---|---|---|---|---|---|---|---|

While coming back ACK will initiate the process of locking up the path and block every intermediate node for its activities. Due to this locking only, we are calling our protocol as lock and forward routing protocol. Locking period will be four times than the total span of traversal time of discovery packet one way, which was calculated while discovery packet was in forward motion.
Traverse time= destination time- start time
Locking period = traverse time x 4

Activity during Lock period



The reason to keep locking period 4 times is that because one unit time will be utilized by ACK to go back to source, in this period only he will keep locking all the nodes for further transmission then it will take 01 unit time to send the actual message to flow in locked environment, one unit time for acknowledgement, finally one unit time will be considered for delays in processing and other overheads.

Locking of a path is required while some dynamism exist in the MANET. Benefit is that routing will become assured up to some extent. After locking also if it is possible for a node to hold its activities during locking period than it would hold otherwise if it wants to move for some unavoidable reasons, it should than send message to the source station about dissolving the lock, and than it is free to move. In this way restrictions on a node during locking period for its other activities can be removed.

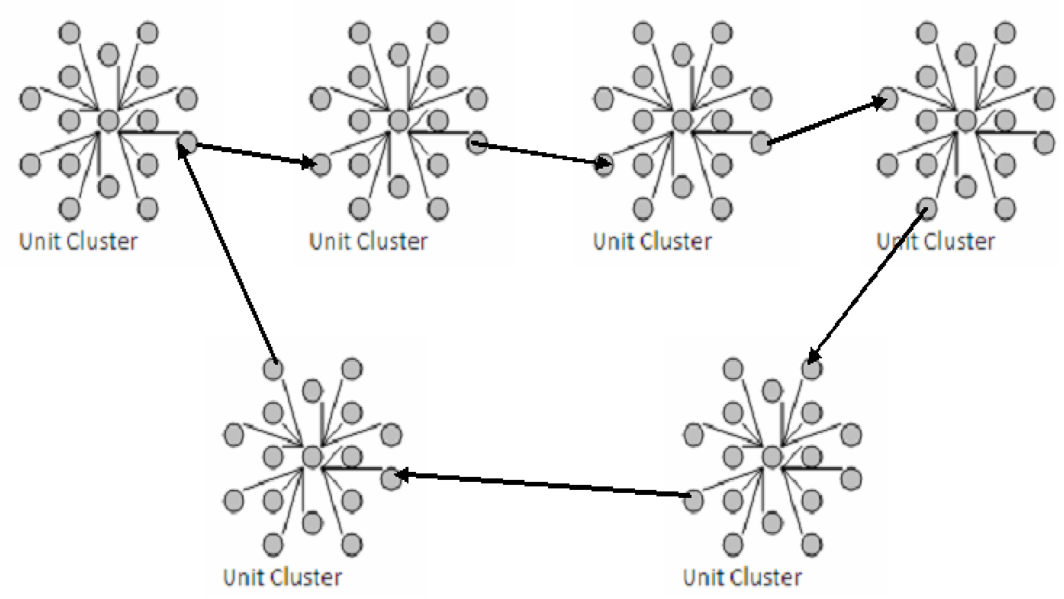
After locking the path immediately the message is routed to its destination. This packet will have its predefined path stamped in its signature, and it has to follow the same path and has to reach the destination within locking period. If the locking period expires than the same process has to start again. Confirmation of the process will be given by acknowledgement.

FORMAT OF ACTUAL DATA PACKET

| STARTING DELIMITER | SOURCE | DESTINATION | ACTUAL ROUTE | START TIME | TOTAL LOCK PERIOD | DATA | ENDING DELIMITER |
|---|---|---|---|---|---|---|---|

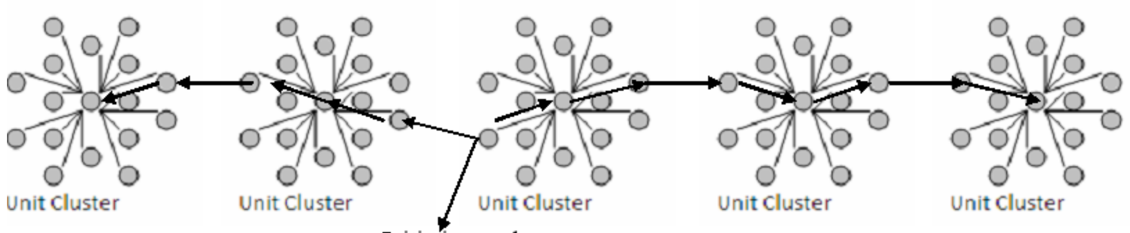
If a route reversal packet (ACK) doesn't come back than it is understood that destination is unreachable. After a maximum waiting time source may start it again the discovery process or wait in between for some random period of time. The job may be dropped for some time if five such failures occur continuously.

PROPAGATION OF MULTICAST DISCOVERY MESSAGE



If a mobile lies in a path which is not expired yet and then if this node wants to move from its location or a full cluster which is part of path wants to move to some other location has to initiate a route dissolution message, because it has to inform source and destination both. It has to send message to both the direction until it is source or destination itself. By seeing this message each node will mark this route as dissolved and delete this route after some time.

Dissolving a established path



Sends message to all the participating nodes that it is leaving from participation

When a node wants to dissolve the path, it may use the following format for the route dissolution.

FORMAT OF ROUTE DISSOLVING PACKET

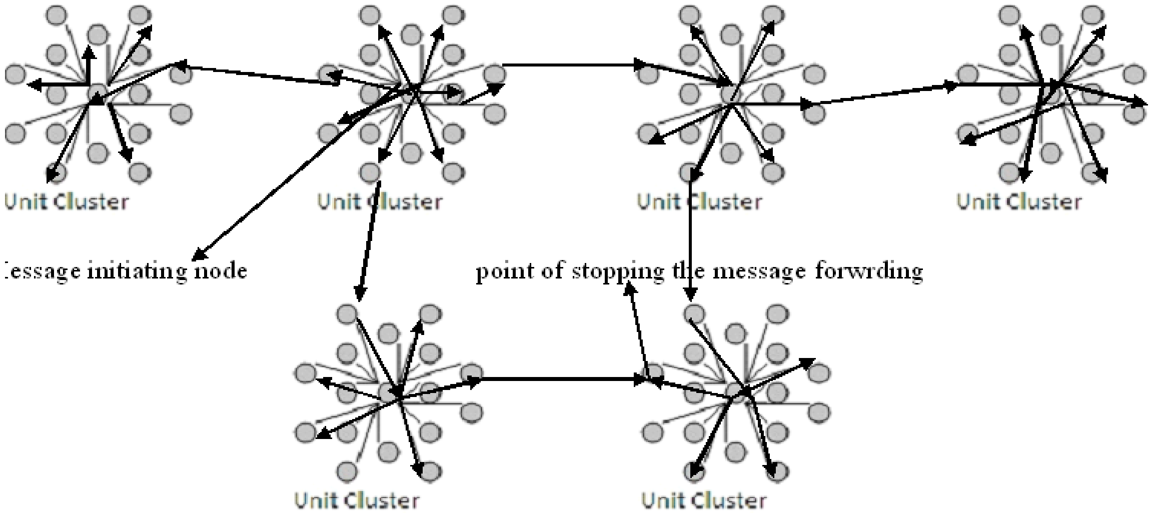
| START | TYPE | SOURCE | FULL PATH WHICH IS DISSOLVED | DESTINATION | INITIATING NODE | END |
|---|---|---|---|---|---|---|

Another type of routing can be non-guaranteed routing. If data is less important than unguarnteed routing can be used such as UDP. There can be only four fields, source address, destination address, checksum and data as a variable length field.

FORMAT OF UNGUARNTEED ROUTING PACKET

| | |
|---|---|
| SOURCE ADDRRESS | DESTINATION ADDRESS |
| MAX HOP | CHECKSUM |
| DATA | |

This type of routing can be done by multicast or broadcast. This is the same process which we have used for discovery of path. Source node will initiate its message in all the directions and further the CH and Gateways will multicast / broadcast it on the multiple paths by sending its copies on the paths originating from them. It would send the message in opposite direction of receiving it. It means that they will not send message to the direction from where it has received it. There is a slight difference between multicast and broadcast. If a copy of message is send to only few selected nodes than it is called multicast and if the message is forwarded to all the connected and near by nodes than it is called broadcast. One method out of these two can be chosen as per situation. In this process at least a message will reach to its destination and then this whole process will stop. Its format is having max hop field. This field will be decremented on every hop and at last it will be destroyed when it will hit zero. Another way of stopping it is if destination finds a duplicate message than it will also be destroyed immediately. There would not be any locking of nodes before sending message and also there would not be any acknowledgement in the process hence important data can not be routed through this protocol.



UNGUARNTEED ROUTING BY BROADCAST OR MULTICAST

Authentication check while routing is most important for security reasons. To check weather a node is authentic or not a node should have something to prove its identity. It will provide following things while routing to other clusters. Its physical address (MAC Address), Its logical address in cluster (IP Address), Its digital signature, Its CH address (logical numbering ) and A certificate issued by its CH with all those information it will form as unique node and authentic one.

Each node getting a message from one direction will forward this into opposite direction of getting and will never forward message into same direction from where it is getting the message. A node getting a same message again will destroy it and do not forward it. This will abstain to form the count to infinity problem and a packet will not form a cycle. To stop a message to roam endlessly there has to be a max hop counter built in its architecture, so that after max hops it will be destroyed. Before sending it has to be initialized at a particular counter i.e. may be 10,100 or 1000 etc or in between some other value. It has to be decided by the max size of the network. While routing this counter will be decremented at every hop until it hits zero. When it hits zero it will be destroyed.

## References

[1] Analyzing Secure Clustering & Maintenance (SCM) Algorithm in Mobile Ad-Hoc Network (MANET) by Dr Manoj Kumar Associate Prof RK PG College Shmali (UP), Mr Vikrant Verma Research Scholar Mewar University- Vasundhara campus Ghaziabad, Mr Sumit Chaudhary Asst Prof Shri Ram College, MZN

[2] Improve on dijkstra shortest path algorithm for huge data Fuhao ZHANG*, Ageng QIU, Qingyuan LI - Chinese Academy of Surveying and Mapping, Beijing, China, 100039, Zhangfh@casm.ac.cn

[3] Prim Algorithm Approach to Improving Local Access Network in Rural Areas - Arogundade O. T., Sobowale B., and Akinwale A. T.

[4] Playing with Kruskal: algorithms for morphological trees in edge-weighted graphs Laurent Najman, Jean Cousty, Benjamin Perret & - Comparing minimum spanning tree algorithms- Igor Podsechin, Tampereen, lyseon lukio, Tietotekniikka

[5] Performance Analysis Between Distance Vector Algorithm (DVA) & Link State Algorithm (LSA ) For Routing Network - Asmaa Shaker Ashoor.( INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 4, ISSUE 02, FEBRUARY 2015, ISSN 2277-8616)

[6] A Survey Paper of Bellman- Ford Algorithm and Dijkstra Algorithm for Finding Shortest Path in GIS Application - Vaibhavi Patel, Prof.Chitra Baggar (International Journal of P2P Network Trends and Technology (IJPTT) – Volume 5 – February 2014)

[7] Destination-Sequenced Distance-Vector Routing (DSDV), C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computer," Comp. Commun. Rev., Oct. 1994.

[8] Wireless Routing Protocol (WRP), S. Murthy and J. J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks, Oct. 1996, pp.

[9] Temporally ordered routing algorithm (TORA), V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *Proc. INFOCOM '97*, Apr. 1997.

[10] Associativity-Based routing (ABR), C-K. Toh, "A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing," *Proc. 1996 IEEE 15th Annual Int'l. Phoenix Conf. Comp. and Commun.*, Mar. 1996.

[11] Adhoc on demand distance vector routing(AODV) Networking working group request for comments:3561 - C.Perkins, Nokia Research Center, University of California, Santa Barbara, University of Cincinnati July 2003

[12] DSR, The dynamic source routing protocol(DSR) for mobile Adhoc Network for IPv4, D.Johnson, Rice University Y.HU UIUC D.Maltz, Microsoft Research Feb 2007.