Partition in mobile adhoc network with mobility-A new approach for effective use of fast IP address autoconfiguration

Mukesh Chand *, HL Mandoria[#]
*Assistant Professor, Gurukul Kangri University, Haridwar
mukesh.geu@gmail.com
#Prof. & Head, IT Dept, College Of Technology, Pantnagar University
drmandoria@gmail.com

***Abstract*-A mobile ad hoc network is a collection of mobile nodes in which a pair of nodes communicates by sending messages either over a direct wireless link, or over a sequence of wireless links including one or more intermediate nodes. Only pairs of nodes that lie within one another's transmission radius can directly communicate with each other. Wireless link failures occur when previously communicating nodes move such that they are no longer within transmission range of each other. Further when the speed increases route failures also occurs rapidly and also affects the performance of the system, such as route discovery time increases, the number of packets dropped increases and also the delay increases. In this paper a new approach is proposed to improve the performance of adhoc network when nodes move with different mobility.**

Keywords: Mobility, Adhoc network, Mobile nodes

I-Introduction

The mobility is the most important attribute of the MANETs. Mobility dictates network [4] and application level protocols. Mobility may be in some cases a challenge for the designer, and may become part of the solution in other cases. We can have several types of mobility models such as: Individual random mobility: When a mobile node moves with random mobility [2].

1. Group mobility: When a group is sharing the same profile of mobility.
2. Motion along replanted routes: When a mobile node follows the predefined path trajectories.

The mobility model can have major impact on the selection of a routing scheme and can thus influence performance [3]. Some time mobility is harmful in Adhoc Network .Some main challenges are identified caused by mobility such as

- Fast auto IP address configuration of mobile nodes when a mobile Adhoc Network merges with other mobile Adhoc Network.
- Fast recovery of IP addresses of mobile nodes when an Adhoc Network is partitioned into sub networks.
- Path breakage - we must prevent packet loss, for instance presetting backup paths etc
- Topology control traffic overhead - one approach to combat path breakage is to “update" the topology very frequently. But, this can have dangerous side effects.
- Long lasting disconnections - One approach will be to design delay and disruption tolerant network protocols.

II-Related work

Nodes in Adhoc networks are mobile and due to this mobile nature of nodes it becomes very difficult to maintain the performance of an adhoc network. A large number of research paper available in the literature have proposed scheme for the mobility management but the solutions are not as efficient as they should be as a scheme for Mobility management with OLSR protocol for fourth generation (4G) mobile networks [1].In this approach the author has described the issues related to mobility but no solution is given to overcome the impact of mobility. In this paper we proposed algorithms to maintain the performance of an adhoc network [4] when some of mobile nodes in the network go out of range which results a partitioned network. As for as mobility is concerned the proposed algorithm is designed in such a way that helps to maintain the performance of an adhoc network.

III-Proposed Algorithm

In this chapter simulation is carried out for Partition of the network [5], in which initially there is one independent MANET is present. Due to mobility [6] the network is partitioned into two Adhoc networks. By considering the proposed algorithm the effect of partition of a network into two or more sub network has been described. Partition can be created due to:

c) Graceless leaving

d) Graceful leaving

In Graceless leaving mobile nodes in the Adhoc network leaves the network without giving information of its departure from the existing Adhoc network. Such partition is known as graceless partition. The result of such departure decreases the performance and increases the unnecessary traffic due to repetition of request. Also when the IP addresses [7] assigned to these mobile nodes that depart abruptly are useless and cannot be allocated until prior intimation of their departure but in the case of graceful leaving IP addresses allocated to these mobile nodes can be reused without conflict.

Graceful leaving of node

In this case nodes leave the network by informing the cluster head of the network.

1. Send Message (Kill) // the leaving node sends a kill message to cluster head $CH_{ACK} \rightarrow$ Node // the cluster head ask the node for confirmation as
2. $Node_{ACK} \rightarrow$ CH // the leaving node sends confirmation
3. CH(Restore IP) // the cluster head restore the node's IP address as
4. $CH_{broadcast} \rightarrow$ node(s) // the cluster head broadcast updated information to all the participating nodes in the network

Graceless Leaving

When nodes are leaving the network without informing then the cluster head [8] check for the existence of all the nodes in the network .It sends a beacon signal to other nodes in the network and waits for response from all the nodes. The algorithm proceeds as follows:

1. Initially there is no beckon signal used in the network
   Beckon_signal=null;
2. All the nodes are idle and then cluster head sends a beckon signal to all the nodes in the network (CH→ Node) in Idle Mode.
   Set
   Beckon _Timer =5;
3. After sending beacon signal the cluster head sets the beckon timer value and waits for Beckon signal from the node (Waits for 5 times Beckon Update Time).
   If (Beackon_Timer==0)
   Update Cluster_Head_table;
   Else
   Send Beckon_signal to node;
   Beckon_Timer =Beckon_Timer-1;
4. If the cluster head does not receive beckon signal from the node then cluster head identify that the node is not present in the network.
   If (Beckon_signal =null)
   Update Cluster_Head_table;
5. Cluster head then flush the entries of the leaving node and mark its IP address as unallocated for reuse by other joining node in the network.
   Cluster_Head_table = Previous (Cluster_Head_table) +free (node_entry)
   IP_List=IP_List + free (IP (node));
6. Broadcast Updated information to all nodes
   Broadcast (IP_List)
   CH ----------------> node (s)

Process model for partition in adhoc network: These process models are used when some of the nodes in the adhoc network go out of range with the prior information of leaving or leaving abruptly.

Process model shown in Fig. 1 is applied on the cluster head .To complete the process five states have been used .Each state process the incoming information and send to the next state by identifying the value of a particular state variable. The state description is as follows:

Send_kill_mesg: In this state the leaving node sends a kill message to the cluster head.

Check_ack: In this state cluster head asks the node for the confirmation and waits for the reply.

Kill_confirm: In this state leaving node sends confirmation message to the cluster head.

Update_IPtab: In this state after receiving the confirmation message from the leaving node the cluster head update the IP table

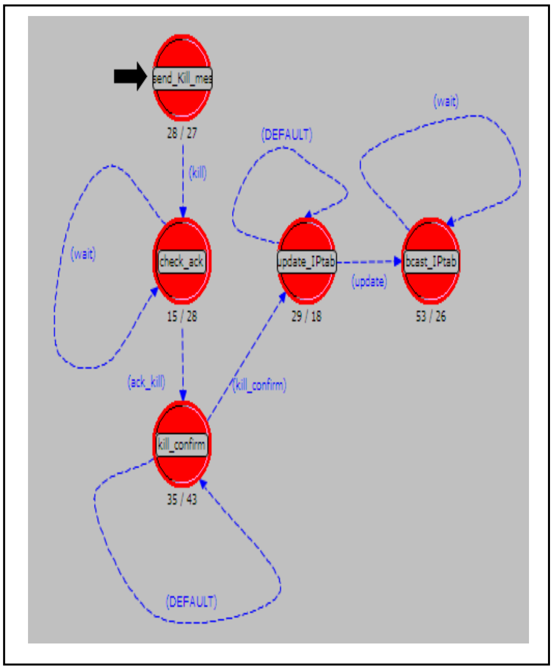

Figure 1: Process model for graceful partition

Bcast_IPtab: In this state the cluster head save the IP address of a leaving node and broadcast the updated IP address table the nearby nodes.

Transition from one state to other state occurs which is based on the value of a state variable. As from Table 3.18 F1, F2, F3, F4 and F5 are the function which are used in a particular state, V1, V2, V3, V4, V5 and V6 are the state variables used for a particular variable in the state.

Process model shown in Fig. 2 is used when some of the nodes in the adhoc network leave their network abruptly without prior information. In this model we have used five states to complete the process. Each state in the process model has different functionalities. G1, G2, G3, G4 and G5 are the key function associated with a particular state, variable W1, W2 ,W3, W4 and W5 are the state variables .Each function process the value of these variables and pass the value of these variables to the next state and causes a state transition. The description of the state is as follows:

Init: In this state each node checks for the updated information at a random time or regular time interval.

Receive_CH: In this state the cluster head checks for the beckon value received from all the nodes.

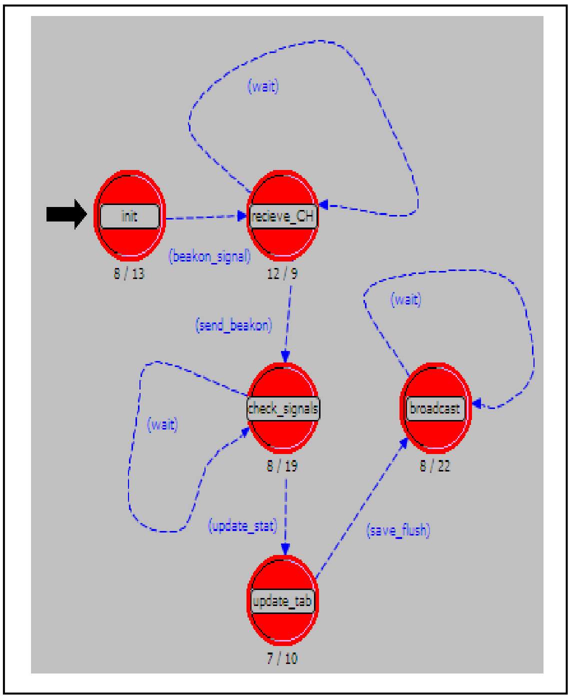

Fig. 2: Process model for graceless partition

Check_signals: In this state the cluster head updates the cluster head table if the value of beckon timer is zero otherwise the cluster head keeps on sending and receiving until the value of beckon timer becomes zero.

Update_tab: In this state when no signal is received by the cluster head then it updates the cluster head table and save the IP address of the node which is not present in the network or just leaved.

Broadcast: In this state after updating the table the cluster head broadcast the updated table to the nearby nodes.

After deploying these two process model on the MANET node model the results are taken.

Simulation & result analysis

In this section we are using random way point model to check the performance of proposed algorithm. In this model, each node selects a random point in the simulation area as its destination, and a speed v from an input range [v_{min}, v_{max}]. The node then moves to its destination at its chosen speed. When the node reaches its destination, it rests for some pause time. If the destination is the next hop then pause time does not affect the performance but if the destination is after some intermediate nodes then the pause time is considered may be considered. At the end of this pause time, it selects a new destination and speed and resumes movement. In the simulation environment we have taken a mobility scenario for partition as shown in Fig. 3. In this scenario two MANETs are deployed in an area of 10 Km * 10 Km .MANET1 covers the area of 5Km * 5Km and consist of 20 mobile nodes which are in the range of each other either directly or indirectly and MANET2 covers the area of 3Km * 3Km. Mobile nodes 10,11,12,13 and 14 are moving with the same speed as shown in Table 1.

Route Discovery Time:

Whenever a node (source) tries to send data to another node (destination) it will first look into its ROUTING TABLE. If source doesn't find entry for destination it constructs a special ROUTE REQUEST packet and broadcast it. The format of ROUTE REQUEST packet is as follows, here Source Address, Request ID used to identify the request uniquely. As source broadcasts the ROUTE REQUEST message to the network, so it is possible that same request may come to a node by different path. Here we have slightly modified the AODV algorithm. When a node receives the same ROUTE REQUEST message by more than one path, unlike AODV it will not simply discard the massage which has came later. It keeps track all the neighbours node by which the ROUTE REQUEST message appear, though the root may be a longer one. Initially when the mobile nodes 10,11,12,13 and 14 depart from the MANET1 the route discovery time is low but when these mobile nodes goes out of MANET1 then the route discovery time increases because of exchanging routing table to update the information about the existance of each mobile nodes in the MANET.Mobile nodes starts moving with zero pause time and finally departs from the MANET1.

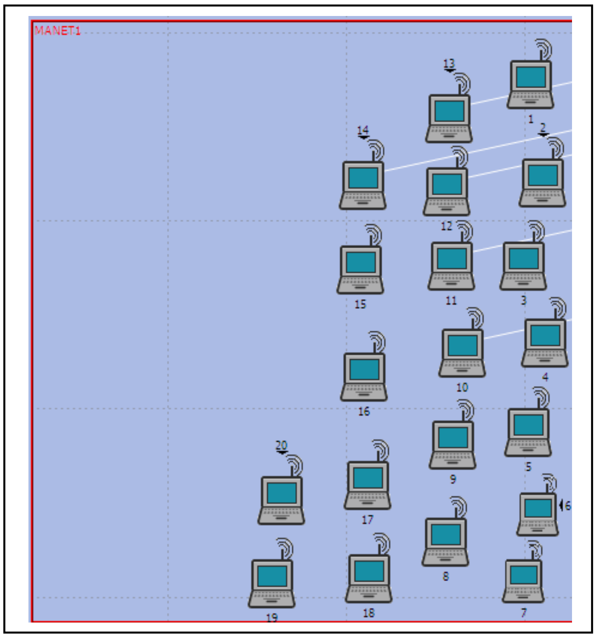

Fig. 3: Mobility scenario for partition

Initally the route discovery time is 0.10 sec and after some time route discovery time increases to 0.18 sec .The total route discovery time increases to 72 % and which is further reduced with the help of proposed algorithm as decribed in chapter 5.When the speed is 30 Km/Hr the total route discovery time decreses from 66% to 51%.When the speed of mobile nodes 10,11,12,13 and 14 is 50 Km/Hr then the total route discovery time decreases from 88% to 63%.When the speed of the mobile nodes is 70 Km/Hr then the total route discovery time decreases from 30% to 18%.The route discovery time is shown in Fig. 4.

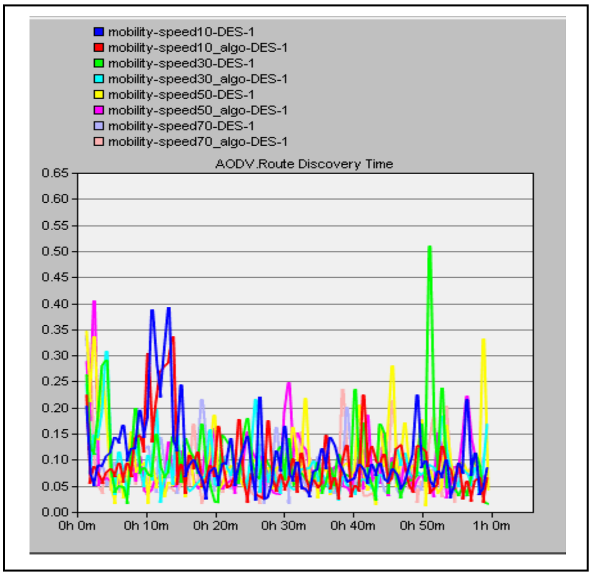

Fig. 4: Route discovery time

Total Packets dropped:

The mobility of nodes makes fixed multi-hop routing unlikely in MANETs. In a dynamically changing environment, the position of the nodes changes over a period of time. As a result, a route determined between any pair of nodes breaks frequently and has to be re-discovered.Untill unless the status of the existence of mobile node is known, the packets are dropped. Initially when the speed of the mobile nodes 10,11,12,13 and 14 is 10 Km/Hr then the total packets dropped are measured as 65 packets but after some time packets dropped reaches to 250 packets. So the total number of packets dropped is further reduced by applying proposed algorithm as described in chapter 5 from 93 % to 78 %.When the speed of mobile nodes 10,11,12,13 and 14 is 30 Km/Hr then the total number of packets dropped is reduced from 80% to 68 %. When the speed of the mobile nodes is 50 Km/Hr then the total number of packets dropped is reduced from 83 % to 60%.When the speed of mobile nodes is 70 Km/Hr then the total number of packets dropped is reduced from 86 % to 68 %.The total packets dropped is shown in Fig. 5.

Delay (sec): Whenever the mobile nodes departs from the MANET then services provided by or to these mobile nodes is also affected .Suppose if a mobile node which is fixed is downloading a file from other mobile node which is moving with some speed then latency will increase to download a file and results in incomplete download after waiting for certain amount of time.Initally when the speed of mobile nodes 10,11,12,13 and 14 then the delay is measured as 0.14 sec and after some time the delay increases to 0.26 sec.The total delay increases to 80% and which is reduced to 43 % when we apply proposed algorithm to the mobile nodes. When the speed is 30 Km/Hr then the total delay is reduced from 83 % to 33 %.When the speed of mobile nodes 10,11,12,13 and 14 is 50 Km/Hr then the total delay is reduced from 52% to 16%.When the speed of mobile nodes is 70 Km/Hr then the total delay is reduced from 98% to 28%.The total delay is shown in Fig. 6.

Table 1: Speed and trajectory

Name of the Node	Speed (Km/Hr)		Trajectory Name	Mobility Model
	Min	Max		
10	0	10	Traj10_Manet10_Partition	Random Way point
	0	30	Traj30_Manet10_Partition	Random Way point
	0	50	Traj50_Manet10_Partition	Random Way point
	0	70	Traj70_Manet10_Partition	Random Way point
11	0	10	Traj10_Manet11_Partition	Random Way point
	0	30	Traj30_Manet11_Partition	Random Way point
	0	50	Traj50_Manet11_Partition	Random Way point
	0	70	Traj70_Manet11_Partition	Random Way point
12	0	10	Traj10_Manet12_Partition	Random Way point
	0	30	Traj30_Manet12_Partition	Random Way point
	0	50	Traj50_Manet12_Partition	Random Way point
	0	70	Traj70_Manet12_Partition	Random Way point
13	0	10	Traj10_Manet13_Partition	Random Way point
	0	30	Traj30_Manet13_Partition	Random Way point
	0	50	Traj50_Manet13_Partition	Random Way point
	0	70	Traj70_Manet13_Partition	Random Way point
14	0	10	Traj10_Manet14_Partition	Random Way point
	0	30	Traj30_Manet14_Partition	Random Way point
	0	50	Traj50_Manet14_Partition	Random Way point
	0	70	Traj70_Manet14_Partition	Random Way point

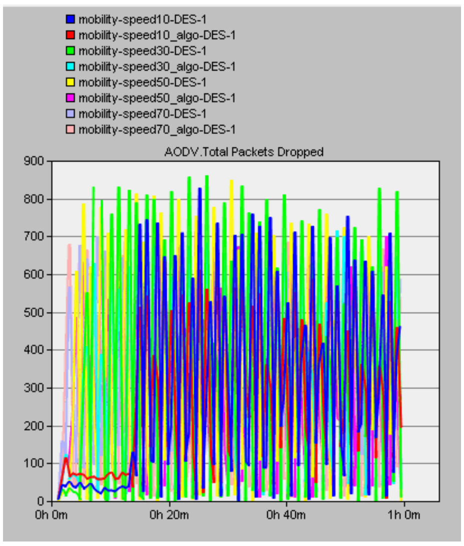

Fig. 5 Total packets dropped

Validation of Proposed Algorithm in case of partition with mobility:

Performance of the system is evaluated for node density 20 and speeds 10 Km/Hr, 30 Km/Hr, 50 Km/Hr and 70 Km/Hr. Performance of the proposed Algorithm in case of partition with mobility is shown in Table 6.3.When the speed of the node 10,11,12,13 and 14 is 10 Km/Hr then the overall performances including parameters route discovery, total packets dropped and delay is improved to 27%. When the speed of the node 10,11,12,13 and 14 is 30 Km/Hr then the overall performance is improved to 33%. When the speed of the node 10,11,12,13 and 14 is 50 Km/Hr then the overall performance is improved to 38%.

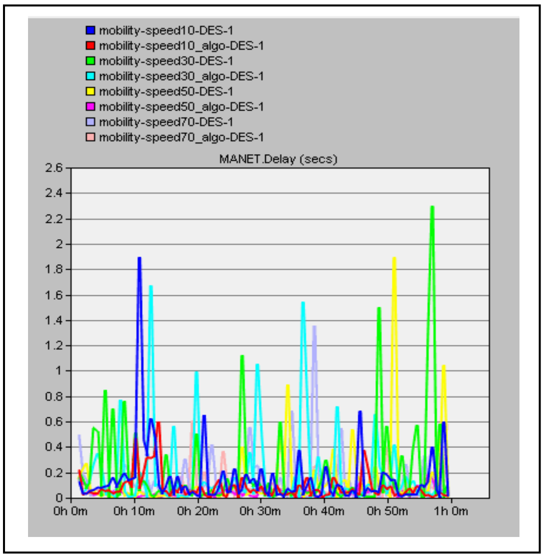

Fig. 6: Delay (sec)

When the speed of the node 10,11,12,13 and 14 is 70 Km/Hr then the overall performance is improved to 46.Though in this simulation node's speed is increased up to70 Km/Hr and overall performance of the system is increasing gradually. The loss rate is 56 % when the proposed algorithm is used and which is slightly efficient than the loss rate calculated from Shabnam Jazayeri[1].

Conclusion

In this chapter the proposed algorithm for partition is applied to check the performance in the case of mobility. In the first scenario when the two MANETs are merged with different speed then the performance of proposed algorithm is increasing gradually .That means the proposed algorithm supports mobility when the different MANETs are merged with different speed. In the second scenario only the proposed algorithm for graceless leaving of nodes is applied to the mobile nodes and the performance is increasing gradually. This means that the proposed algorithm support mobility in the case of partition.

References

[1] Jazayeri, S.; , "Mobility Management with OLSR Protocol for Fourth Generation (4G) Mobile Networks," *Next Generation Mobile Applications, Services and Technologies, 2008. NGMAST '08. The Second International Conference on*, vol., no., pp.598-603, 16-19 Sept. 2008doi:10.1109/NGMAST.2008.12URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4756493&isnumber=4756389

[2] Sourav Saha; Mukherjee, M.; Neogy, S.; , "Mobile-autoconf: Mobility management with autoconfiguration in mobile ad-hoc networks," *TENCON 2008 - 2008 IEEE Region 10 Conference* , vol., no., pp.1-6, 19-21 Nov. 2008doi:10.1109/TENCON.2008.4766496URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4766496&isnumber=4766377.

[3] Rohankar, R.; Bhatia, R.; Shrivastava, V.; Sharma, D.K.; , "Performance analysis of various routing protocols (proactive and reactive) for random mobility models of Adhoc networks," *Recent Advances in Information Technology (RAIT), 2012 1st International Conference on* , vol., no., pp.331-335, 15-17 March 2012doi:10.1109/RAIT.2012.6194441URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6194441&isnumber=6194433

[4] Bai, F.; Narayanan Sadagopan; Helmy, A.; , "IMPORTANT: a framework to systematically analyze the Impact of Mobility on Performance of Routing Protocols for Adhoc Networks," *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies* , vol.2, no., pp. 825- 835 vol.2, 30 March-3 April 2003doi:10.1109/INFCOM.2003.1208920URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1208920&isnumber=27206

[5] Ze Li; Haiying Shen; , "A mobility and congestion resilient data management system for distributed mobile networks," *Mobile Adhoc and Sensor Systems, 2009. MASS '09. IEEE 6th International Conference on* , vol., no., pp.60-69, 12-15 Oct. 2009doi:10.1109/MOBHOC.2009.5336942URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5336942&isnumber=5336910

[6] Ting Wang; Chor Ping Low; , "Dynamic Message Ferry Route (dMFR) for Partitioned MANETs," *Communications and Mobile Computing (CMC), 2010 International Conference on* , vol.3, no., pp.447-451, 12-14 April 2010doi:10.1109/CMC.2010.125 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5471612&isnumber=5471305

[7] Thoppian, M.R.; Prakash, R.; , "A distributed protocol for dynamic address assignment in mobile ad hoc networks," *Mobile Computing, IEEE Transactions on* , vol.5, no.1, pp. 4- 19, Jan. 2006doi:10.1109/TMC.2006.2

[8] Garcia-Palacios, E.; Mehallegue, N.; Safdar, G.A.; , "Providing security and energy efficiency in wireless ad-hoc sensor networks through Secure Cluster-Head Election (SEC-CH-E)," *Telecommunications and Signal Processing (TSP), 2012 35th International Conference on* , vol., no., pp.47-51, 3-4 July 2012doi:10.1109/TSP.2012.6256250URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6256250&isnumber=6256179