Volume 3, Issue 10, October 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Improved Scalable Multicasting over Ad Hoc Networks in Parallel Environment

D.Adithya Chandra Varma
School of Computing Science
India

Praveen Kumar Reddy.M
School of Computing Science
India

Prof.VarunKumar.M
School of Information Technology
India

Abstract— For group communication, one of the most efficient methods is multicasting. Over a dynamic topology it is difficult to implement group membership management and multicast package forwarding. So a new algorithm in Economic Spatial Group Multicast protocol(ESGMP) which uses a virtual-zone-based structure which was proposed to provide multicasting environment, in which nodes are divided into Zones. A multicast tree is constructed and a cost efficient multicast packet forwarding procedure is obtained using ESGMP and parallel algorithm. We ensured that when the nodes move from one zone to another zone EGMP maintains an efficient multi cast tree. We also have parallelized the mobility of nodes among the zones which gives us a scalable multicast protocol.

Keywords— Multicasting, Ad Hoc Networks, Zones, parallel algorithm

I. INTRODUCTION

Group communications over Mobile Ad Hoc Networks (MANETs) have caught increasing importance in recent years. Some of the practical implementations could be found among group of soldiers in a battlefield, among the relief workers working in groups during natural calamities. To realize such group communications multicasting is one of the best method .With multicast method we can operate on one-to-many or many-to-many systems. But implementing efficient multicasting over a MANET is not easy when the topology constantly keeps on changing [2]. Traditional MANET multicast protocols can be described in two major categories where one is tree-based and the other is mesh based. However the problem lies in maintaining the tree structure for constantly moving nodes using conventional protocols. The remedy for the problem was found when few years back a unicast routing, geographic routing protocol was proposed. Still the entire problem couldn't be solved using unicast protocol. There were many disadvantages of using the protocol. Unicast model impose constrains over the application of geographic multicasting within a small group. In order to overcome such problems we have proposed a new category of multicasting protocol called Economical Group Spatial Multicasting protocol (EGSMP).Here we can scale large group and network size. More appropriately we worked on tree structure. In this paper we have focused on following schemes [1].

1. We have divided a geographical area into certain number of zones and zones may contain any number of nodes ranging from zero to a finite number.
2. We determined zone leaders for each and every zone.
3. We kept track of the groups formed with the node leaving and joining zones.
4. We formed multicast tree.
5. We determined the most economical path of communication using parallel code.

II. RELATED WORK

First we discuss conventional multicast protocols and the basic procedures applied on it. And then we give existing geographical multicast algorithms in the literature [2][3]. Tree based protocols and mess-based protocols are examples of conventional topology-based multicast protocols**.** Efficient forwarding of packets to the group members can be done by using tree based protocol. When some of the links break the packets can be forwarded by using mesh based protocol. Since the maintenance of conventional mess or tree structure requires dynamic network, the topological based protocols are difficult to scale to the big networks size. The algorithm provides the efficient stateless multicast protocol, which gives a good scalability for both network size and group size [5]. As in this paper we develop the efficient protocol for scaling location-based multicast, the topology-based protocols will not be considered any further. Just for the record similar portability and system setup, the delivery ratio. A network zone bidirectional tree was used in order to maintain the packet delivery among the nodes. Here we maintain the position of the node and based on it the multicast tree is constructed and then the packet is forwarded [6]. This technique reduces the overhead and the time in route searching.

But when the network size is large the mobility of nodes among the zones is very frequent so the membership management and the effective route optimization is difficult [7].

III. APPROACH

ESGMP provides multicasting using zones. Here the nodes self-organize themselves. All the data transmission among the nodes and network management in the system is done by Zone leader's .Later in the paper we come to know on what bases a zone leader is elected. A multicast tree has to be maintained by the zone leader in order to manage the flow of data efficiently. This paper organizes as follows. Initially we have to construct a tree in order to identify the group members and to maintain their flow .this tree is the base for zone leader to identify which nodes are joining the new group and which nodes are leaving the group [1].

A. **Zone leader:** It is elected based on some rules. Before electing, the node which come in to the network has to send its existence in the network .It sends a message NEW_ SESSION to all nodes in the network indicating its arrival. It removes from the network by sending END_SESSION indicating its departure.

The main task in order to ensure multicasting is the electing of Zone leader which is done based on the following rules

1. If the neighbor table contains no nodes in the same zone, the node itself announces as the zone leader
2. Initially all the flags of the nodes in the zone are empty, that means there is no leader in the zone.

For this problem we find out the node that is colder to the center of the zone and announce it as the zone leader and its leader flag is set. In our approach we have find out the node which is center to the zone and we use this as a support to send the packets to different nodes in different zones .to explain this suppose node 16 wants to send data to node 7 of zone(0,1).so it sends the data to node 18 of its own zone and then it tries to find out the center to the zone (0,1) so that it can send the message to the node 7.since there is no node closer to the zone(0,1) there is chance of data lost in this situation. To overcome this problem we introduced Zone mode so that the message transfer is guaranteed [1].

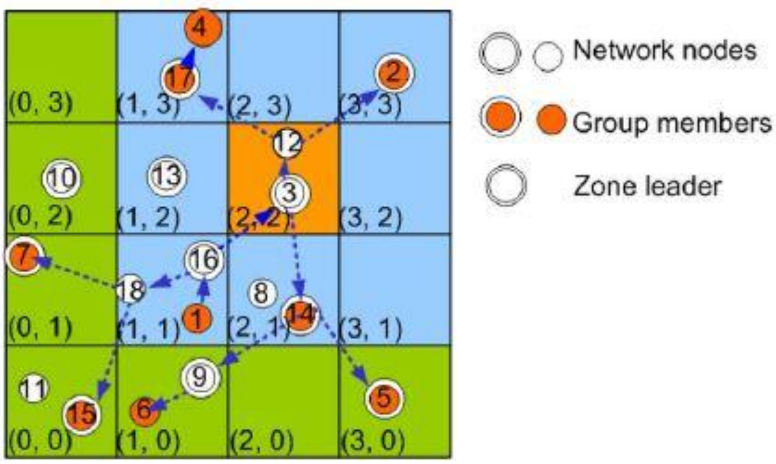

Fig1: Zone Structure and multicast session example.

The major part of this paper is to construct the multicast tree. Whenever the node of one zone wants to join with other zone or leave from the existing zone the construction of efficient multicast tree has to be ensured. The multicast tree maintaince is the responsible of zone leader of the corresponding zone [1].

B. **New group join:** When a node wants to join in to new group a JOIN_REQUEST message is sent to its zone leader where the JOIN_REQUEST contains node address, position and address of zone to which it wants to join. The zone leader verifies the multicast table and it communicates with the other zone leader to which the node wants to join. Then this zone leader sends JOIN_REPLY to the node and hence the node joins in to a new zone.

Table1: Neighbour table of node 18 in fig1

Node id	Position	Flag	Zone id
16	(x16,y16)	1	(1,1)
1	(x1,y1)	0	(1,1)
7	(x7,y7)	1	(0,1)
3	(x3,y3)	1	(1,2)

C. **Group leave:** Whenever a node wants to leave from the group it sends LEAVE message to the zone leader which contains the group id .The zone leader of the corresponding node empties its entry from the multicast tree.

D. **Packet delivery and forwarding to the nodes:** Whenever a node wants to transfer message it sends it to the zone leader. The zone leader checks for the acknowledgement flag in the multicast tree .If it is unset which means it is not a leader the packet is forwarded to the root node where the root zone is responsible for sending the packets to the corresponding nodes.

Entry of Group G in Multicast table of Node 16 in the fig1.

Group id	G
Root zone id	(2,2)
Upstream zone id	(2,2)
Downstream zone list	(0,1)(0,0)
Downstream node list	1

E. **Path maintenance in the network:** Since the network is dynamic and it changes periodically the connection among the nodes and the maintained of multicast tree and its path maintained is a major concern in the MANET. In the network the nodes move from one zone to another zone and hence there is a possible of zones becoming empty. Here to solve such a problem the zones are divided into many clusters.

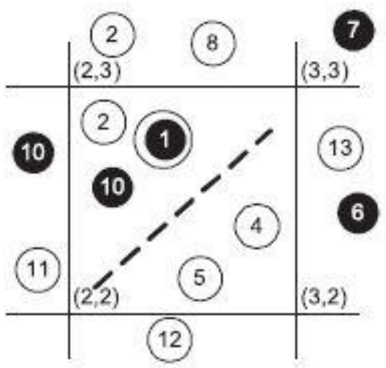

Fig 2 Multiple Clusters in Zone1 for fig1.

F. **Empty zone problem:** Whenever all nodes in the zone move away from the zone it becomes empty. Assuming the range and distance separated by nodes in the zone we can estimate the probability that a zone can become empty. Based on the probability we can adjust the multicast tree and make sure the tree is connected and packet is delivered among the nodes. Whenever the leader is moving away from the zone and due to this it is becoming empty it sends its multicast table to the zone which is at its upstream. Then the leader their at upstream takes the responsibility and delete the zone from the network. Then the zone leader sends a message to all the nodes in its zone to make sure that a new node is entered and if there are any changes it can change. The nodes sends REPLY message to the zone leader after updating.

G. **Path optimization:** There is possible of receiving a duplicate packet from the nodes in the network. This situation arises where the zone leader sends a message to its upper zones to indicate the arrival of new node to its zone and keeps a time to receive acknowledgement from the nodes. If it does not receive any ack message there is a possible of packets to lose the message. In such case the root zone will take care of this problem.

IV. IMPLEMENTATION

In this paper we focused on several numbers of nodes. Next we have divided the nodes into zoned or more accurately Meta nodes. Each zone is visualized as a branch node. We mainly emphasized a protocol and cost of communication in this paper. We already discussed about the protocols in the previous sections. So for determining the cost of communication and the shortest possible path for transmission between individual nodes in the same zone or different zone and among different zones we have implemented shortest path algorithm in and outside zones. As the messages come in the domain of cost we have determined the efficiency of communication based on the number messages communicated between different zones and among the nodes in the zones. More the numbers of messages will be exchanged more will be the cost efficiency. For each passing message we have determined a fractional constant that will be multiplied to the number of messages passed to determine the cost of transmission. The entire methodology is discussed more appropriately in the following section [6]. Suppose there are N number nodes. We have divided the nodes into certain zones. Assume the number of zones to be Z where Z<N always. Let us take that entire number of nodes is divided as{1,2…,a},{a+1,a+2,……b},…….,{k,k+1,…N}. We used a shortest path algorithm to determine minimum cost path in each zone as well as among the zones. Again the efficiency of the system is directly proportional to the number of messages exchanged. So we incorporate this idea by reducing the cost on messages passed between nodes. In this paper we have parallelized the way in which the nodes communicate with each other. Also the mobility of nodes is parallelized. By using this algorithm we can implement a efficient multicast protocol. In general there is a possible of weights going negative. To handle this we need a dynamic programming which solves the problem [2][4].

```
1)  Procedure  path construction()
2)  for  i=1 to x
3)  for j=1 to x
4)  for k=1 to x
5)  if path[j][k] > path[j][i]+path[i][k]
6)  {
7)  path[j][k]=   path[j][i] +path[i][k]
8)  next[j][k]=i;
9)  }
10) path(j,k)
11) if there is no path i.e path[j][k]==infinity then
12) return NULL;
13) else   intermidiate= next[j][k];
14) if intermidiate=NULL then there exists an edge from
    j,k with no vertices.
15) else
16) return
    path(j,intermidiate)+intermidiate+path(intermidiate,k
    );
```

V. PERFORMANCE ANALAYSIS

In this paper we have implemented a code to speed up communication between nodes in the zones and nodes in different zones. We have showed that with increase in number of nodes the communication time is maintained round a steady value .We particularly noticed that using paralyzed code we can slow down the time of communication between the nodes which is a generally overhead using serial code.

Table showing nodes and its execution time

No of nodes	Serial time(μs)	Parallel time(μs)
100	1.5023	0.5003
500	2.1237	0.5123
1000	2.6732	0.5274
2000	3.1298	0.5298

We have also computed speed up factor by using equation

$$\text{Speed Up} = (\text{Parallel execution time} / \text{Serial execution time}) \quad \rightarrow \quad \textbf{(1)}$$

A. For Two Cores:

No of Nodes	Serial Execution Time in μs	Parallel Execution Time in μs	Speed Up In μs
100	1.5023	0.5003	3.0027
500	2.1237	0.5123	4.1454
1000	2.6732	0.5274	5.0686
2000	3.1298	0.5298	5.9075

B. For Four Cores:

No of Nodes	Serial Execution Time in μs	Parallel Execution Time in μs	Speed Up In μs
100	1.5023	0.3721	4.0373
500	2.1237	0.3870	5.4875
1000	2.6732	0.3983	6.7115
2000	3.1298	0.3996	7.8323

C. **For Eight Cores:**

No of Nodes	Serial Execution Time in µs	Parallel Execution Time in µs	Speed Up In µs
100	1.5023	0.2771	5.4215
500	2.1237	0.2832	7.4989
1000	2.6732	0.2826	9.4593
2000	3.1298	0.2932	10.6746

D. **Speed Up Comparison:**

No of Nodes	Two core	Four core	Eight core
500	3.0027	4.0373	5.4215
1000	4.1454	5.4875	7.4989
1500	5.0686	6.7115	9.4593
2000	5.9075	7.8323	10.6746

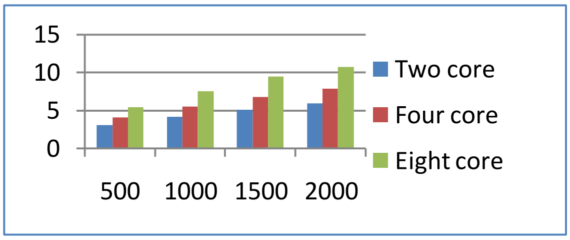

Fig 3 Graphical representation for speed up

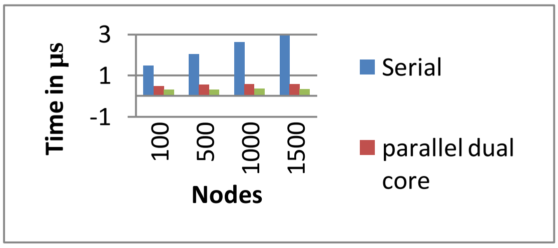

Fig 4: Comparision graph

VI. Conclussions

In this paper we have come across all the possible methods and procedures to handle group communications over mobile ad hoc networks and to determine a efficient protocol to minimize the cost of communication. The mobility of the nodes between the inter zones and intra zones is parallelized. The inter zone mobility is handled by adding an extra overhead in between the respective zones. This results into constructing of efficient multicast protocol.

Acknowledgment

We would like to thank the School of Computer Science and Engineering, VIT University, for giving us such an opportunity to carry out this research work and also for providing us the requisite resources and infrastructure for carrying out the research.

References

[1] X. Xiang, X. Wang, and Y. Yang, IEEE Transaction Supporting Efficient and Scalable Multicasting over Mobile Ad Hoc Networks, transactions on mobile computing, vol. 10, no. 5, April 2011.

[2] L. Ji and M.S. Corson, "Differential Destination Multicast: A MANET Multicast Routing Protocol for Small Groups," Proc. IEEE INFOCOM, Apr. 2001.

[3] E.M. Royer and C.E. Perkins, "Multicast Operation of the Ad Hoc On-Demand Distance Vector Routing Protocol," Proc. ACM/IEEE MobiCom, pp. 207-218, Aug. 2006.

[4] C. Wu, Y. Tay, and C.-K. Toh, "Ad Hoc Multicast Routing Protocol Utilizing Increasing Id-Numbers (AMRIS) Functional Specification," Internet draft, Nov. 2001.

[5] X. Zhang and L. Jacob, "Multicast Zone Routing Protocol in Mobile Ad Hoc Wireless Networks," Proc. Local Computer Networks (LCN '03), Oct. 2008.

[6] C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for Multihop Mobile Wireless Networks," ACM J. Cluster Computing, special issue on Mobile computing, vol. 1, no. 2, pp. 187-196, 1998.

[7] J.J. Garcia-Luna-Aceves and E. Madruga, "The Core-Assisted Mesh Protocol," IEEE J. Selected Areas in Comm., vol. 17, no. 8, pp. 1380-1394, Aug. 2009.

[8] M. Gerla, S.J. Lee, and W. Su, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," Internet draft, draftietf- manet-odmrp-02.txt, 2010.

