Location-Based Multicast Routing Protocol for Mobile Ad Hoc Networks

TZAY-FARN SHIH[1†] , CHAO-CHENG SHIH[2], and CHIN-LING CHEN[1]
[1]Department of Computer Science and Information Engineering
Chaoyang University of Technology
Taichung, Taiwan 41349, R.O.C
[2]Department of Information Management
Shih Hsin University
Taipei, Taiwan 116, R. O. C
tfshih@cyut.edu.tw, ccshih@cc.shu.edu.tw, clc@mail.cyut.edu.tw

Abstract: - Wireless network offers freedom moving around the effective transmission area and the flexibility and easy to use function for Internet application. Many applications of computer network involve multiple users that will rely on the ability of the network to provide multicast services. Thus, multicasting will be concerned as an essential part of Ad Hoc networks. Some of the proposed routing algorithms require maintaining a global network state at each node, the imprecision of global state and the large amount of storage and communication overhead induce poor scalability. In this paper, we propose a distributed cluster-based QoS multicast routing algorithm which only requires maintaining a local state at each node. The location information provided by positioning device is aided in route discovery and route maintenance procedure. Our protocol partitions the network into square clusters. In each cluster, a *cluster head* and *gateways* are selected by a *cluster head selection algorithm* and a *gateway selection algorithm* respectively. After the construction of cluster heads and gateway nodes, a distributed computation collectively utilizes the local state information to construct multicast tree in a hop-by-hop basis. Simulations are conducted to evaluate the performance of our algorithm. As it turns out, our protocol has better performance and lower routing overhead than the non-cluster based algorithm.

Key-Words: - Mobile ad hoc networks, Multicasting, QoS, GPS, Proactive routing, Reactive routing

1 Introduction

The advancement in wireless communication and portable computing devices has made mobile computing possible. Without any typical wiring requirements, wireless networking offers freedom moving around the effective area. Temporary workgroups can also be assembled, conference rooms made network ready without extensive rewiring of corporate offices. There are currently two variations of mobile wireless networks. The first is known as the infrastructure networks that have fixed base stations. A mobile unit within these networks connects to, and communicates with, the nearest base station that is within its communication range. The second type of mobile wireless network is the infrastructureless mobile network, commonly known as a mobile ad hoc network. An ad hoc mobile network is a collection of mobile nodes that are dynamically and arbitrarily located in such a manner that the interconnections between nodes are capable of changing on a continual basis. Unlike conventional wireless networks, ad hoc networks are wireless network with no fixed routers, hosts, or wireless base stations. Nodes of these networks function as routers, which discover and maintain routes to other nodes in the network [1]. According to how route information is collected, ad hoc network routing protocols can be classified as proactive and reactive [2-8].

Many applications of computer network such as videoconferencing will involve multiple users that will rely on the ability of the network to provide multicast services. Thus, multicasting will likely be an essential part of networks. In multicast

[†] Corresponding author.

communication, messages are concurrently send to multiple destinations, all members of the same multicast group. One of the core issues that need to be addressed as part of providing such mechanisms is multicast routing. It has recently attracted a lot of attention in the design of multicast routing protocol for ad hoc mobile network [9-23]. Recently, QoS in ad hoc networks has also received more attention. QoS guarantee to satisfy a set of predefined performance requirements for the users while the transmissions of packets [24]. QoS routing is the process of choosing the routes to be used by the flow of packets of a logical connection in attaining the associated QoS constraints. There are two QoS multicast routing strategies [25-27], source routing and distributed routing, classified according to the way of how the search of feasible paths is carried out and how the state information is maintained. In the source routing, a feasible path is locally computed at the source node that induce the scalability problem in large networks. In the distributed routing, the path computation is distributed among the intermediate nodes so it is more scalable. The availability of small, inexpensive low-power GPS receiver and techniques for calculating relative coordinates based on signal strengths make it possible to apply position-based routing algorithm in ad hoc mobile network. There are some position-based routing protocols were proposed recently [28-32].

In this paper, we propose a scalable and loop-free distributed cluster multicast QoS routing protocol, which requires every node to maintain only its local state and uses physical location information provided by positioning devices [33, 34] in route discovery and route maintenance. In our protocol, the whole network is partitioned into several square zones called zones or clusters. In each cluster, we first select a cluster head by a cluster head selection algorithm and then use a gateway selection algorithm to select gateways of neighbor cluster heads. After the construction of cluster heads and gateway nodes, it uses a distributed computation to collectively utilize the most up-to-date local state information to find multicast tree in a hop-by-hop basis. Our cluster-based routing algorithm only use source, destination, cluster heads and gateway nodes for route probing, so that the route probing packets can be reduced significantly. Our algorithm can be applied to solve both unicast and multicast routing problem. The performance of our algorithm was studied through extensive simulation. The simulation results reveal that it has much better performance than non-cluster mesh based algorithm ODMRP.

The rest of the paper is organized as follows. Our protocol is described in Section 2. Section 3 presents the simulation model and the simulation results. Finally, we give a conclusion in Section 4.

2 Location-Aware Cluster Multicast QoS Routing protocol

In this section, we describe the operation of our distributed location-aware cluster multicast QoS routing protocol (LACMQR).

Cluster construction principle: Using R and L to denote the effective transmission radius of each mobile node and the side length of square regions respectively. In our proposed protocol, it sets the value of L equal to $R/\sqrt{2}$ that guarantees each pair of nodes in the same cluster always within the effective transmission range. By the assistance of position information of each node that obtained from positioning device such as global positioning system (GPS), each node can do self-clustering. The entire network is divided into a number of $L \times L$ square regions, called zones or clusters.

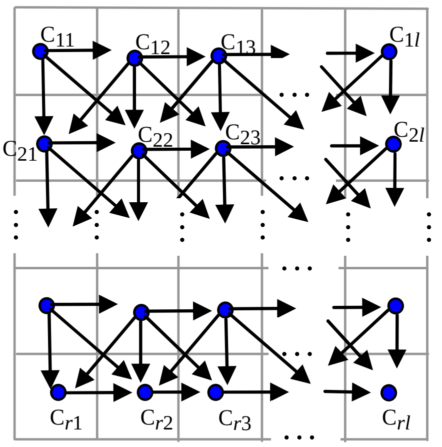
Cluster head and gateway selection policy: After the clusters have been constructed, a cluster head selection algorithm is used to determinate a cluster head of each cluster. Next, a gateway node selection algorithm is exploited to select the gateway node between adjacent clusters. Gateway nodes are responsible for packet relay while the adjacent cluster heads are out of the effective transmission radius. The cluster head selection algorithm always chooses a node nearest to the center of cluster as the cluster head by contention. A node of this kind has longer distance away from the side of cluster; it will take more time to roam out of this region so that it will keep a longer route life. If the distance of two adjacent cluster heads is longer than the effective transmission radius, the gateway selection algorithm is running to choose an intermediate node that is nearest to those two cluster heads as a gateway node. Otherwise, the gateway selection algorithm will not be triggered.

Best predecessor replacement strategy: The procedure of route discovery is based on a best predecessor replacement policy [35]. It progresses as follows. When a node receives a probe packet, it checks the QoS constraints and compares the accumulated metric (e.g. delay and cost) of the current probe packet with the previous probe packets'. If the QoS constraints are satisfied and the accumulated metric of the new probe is better than the previous probes' accumulated metric, the node changes its predecessor to the node that the new probe packet comes from and forwards this probe packet immediately. Owing to every node select the best predecessor, the path constructed by this algorithm is optimal.

Route discovery procedure: When a source node needs to transmit packets and a valid route is not available, it initiates a route discovery procedure to setup a path. It sends a route probe packet, called PROBE, to its cluster head. If the destination is in the same cluster, the cluster head will forward this probe packet to the destination node directly. On the contrary, the cluster head forwards the PROBE packet to its gateway nodes. After receiving the PROBE, the gateway nodes forward the PROBE to the proper neighboring cluster head immediately, the process is repeated until either the destination or an intermediate node with a valid route to the destination is reached. Then, the destination or intermediate node will select an optimal route based on the best predecessor replacement policy and reply an acknowledgement packet, denoted ACK, to its predecessor. The ACK packet will be continually forwarded along the reverse direction of PROBE until the source node is reached. Once the source node received the ACK packet, the route is established and the source node starting to transmit data packets. While a node received a PROBE packet with better accumulated metric and there is a route between the source node and this node, the node must to send a TEARDOWN packet to the source node to delete the old path between them. The algorithm running at each cluster head node c and gateway node g is illustrated in appendix A.2 and A.3 respectively. Appendix A.1 and A.4 depicted the algorithm running in the source and destination node.

Route maintenance: When a packet encounters a broken link in the data transmission procedure at a node. The node will inform the source node immediately by sending an ERR packet backward to it. While an immediate node receives an ERR packet, it deletes the related routing table entry to the broken link and relays the ERR packet to source node. While the source node received an ERR packet, it deletes the related entry of routing table and initiates route discovery to reconstruct new path immediately.

Theorem 1: If the number of source and destination nodes is n_s and n_d respectively and the whole network is partitioned into r rows and l columns, the maximum number of nodes participate in the route discovery process n_r is less than $5rl+ n_s + n_d$.



Proof:

Inspect the above figure, we find a rule to calculate the total number of gateway nodes without reduplicate. In column 1's first (r-1) rows, each row has at most 3 gateway nodes. In column 2 to column (l -1)'s first (r-1) rows, each row has at most 4 gateway nodes. In column l's first (r-1) rows, each row has at most 2 gateway nodes. In row r's first (l-1) columns, each column has at most 1 gateway node. The total number of gateway nodes can be calculate as $[3 + 4(l - 2) + 2] (r-1) + (l-1)$.

The total number of cluster head nodes will no more than the total number of regions that is equal to rl.

n_r = number of gateway nodes+ number of cluster head nodes + $n_s + n_d$

$< = \{[3 + 4(l - 2) + 2] (r-1) + (l-1)\} + rl + n_s + n_d$

$< 4rl + rl + ns + nd = 5rl + n_s + n_d$. Q.E.D.

In ODMRP routing protocol, all network nodes must participate in the route discovery process. A node received a probing packet will replicate and forward it to all neighbor nodes within propagation range. The probing traffic is proportion to the number of network nodes n that will cause tremendous probing packets and is not suitable for large scale network. In our proposed protocol, the route discovery process is responsible by the source node, destination nodes, cluster heads and gateway nodes not by all network nodes. The probing traffic is proportion to the number of clusters that decreases the probing traffic significantly. Thus, LACMQR is scalable. If the number of source and destination nodes is n_s and n_d respectively and the whole network is partitioned into r row and l column, the maximum number of nodes participate in the route discovery process n_r is less than $5rl+ n_s$

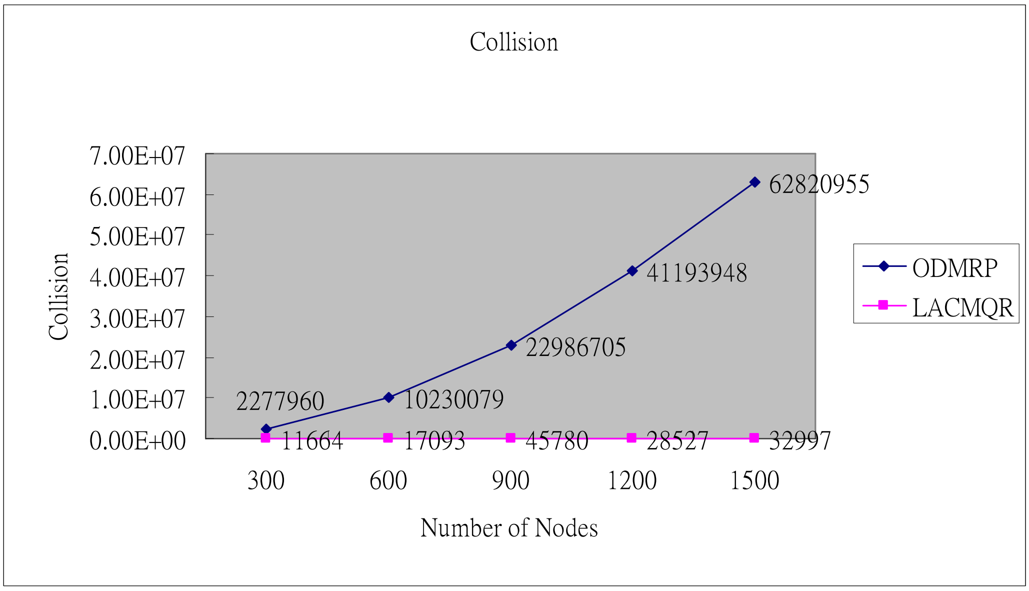


Fig. 1 Collision occurred in ODMRP and LACMQR.

+ n_d. The proof is given in theorem 1. The larger number of network nodes n, the more efficiency our protocol will be.

3 Simulation Model and Results

We have developed a simulator for our distributed cluster based routing protocol LACMQR. The simulator was implemented within Global Mobile Simulation (GloMoSim) library by C++ language [36]. The GloMoSim library is a scalable simulation environment for mobile wireless network using parallel discrete-event simulation capability provided by PARSEC [37]. We tried to evaluate the performance of LACMQR and ODMRP. The implementation of ODMRP followed the specification in the Internet Draft draft-ietf-manet-odmrp-02.txt [14]. The real execution time, average collision, average probe overhead, data loss rate and throughput are studied by simulation. The network nodes were generated according to a uniform distribution. All nodes were placed in a 1000 M × 1000 M range to simulate actual network. In LACMQR, we let the side length of the square region L to be 200 meters. In every run, there are two multicast groups. One of them has two source nodes and the other has one source node. The traffic generators used by the three source nodes in the simulator are CBR. The CBR simulates a constant bit rate traffic generator. The generators initiated the first packet in different time and send a 512 bytes packet every 500ms time interval. The join time and leave time of all group members were set to 0 and 400 seconds respectively. The QoS constraint we concerned in the simulation is end-to-end delay. The bandwidth is 2Mbps.

Fig. 1 shows the total times of collision happened in ODMRP and LACMQR for different network size. The number of collision occurred in LACMQR is much less than ODMRP and collision is increased in proportion to the network size. This result meets our expectation. The probability of collision is proportioned to the number of packets want to be transmitted. The lager amount of nodes needs to transmit packet and transmitting packet by broadcasting will produce a mass scale of traffic and induce more collision. In ODMRP, a lot of nodes take part in the route probing and data relay process and it transfer data and control packet by broadcasting, which causes the times of collision to increase near exponential. On the contrary, in LACMQR only cluster head, gateway, source and destination nodes are responsible for routing and data transmission. LACMQR send packet to target node by unicasting. These two characteristics result in the lower occurrence of collision in our protocol.

The control packets of ODMRP include join query packets, join reply packets and acknowledgement packets. In LACMQR, the control packets consist of join query packets, join reply packets and tear down packets. Fig. 2 depicts the curve of the total number of control packets transmitted in ODMRP and LACMQR. The figure shows that the mesh-based protocol ODMRP produced higher control packets

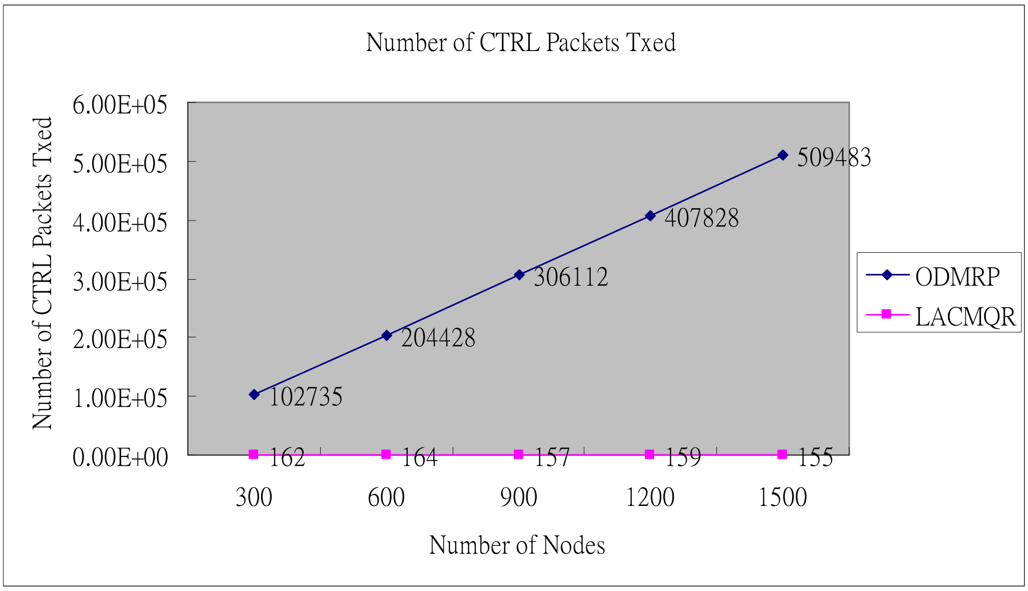


Fig. 2 The number of Control packets transmitted by ODMRP and LACMQR.

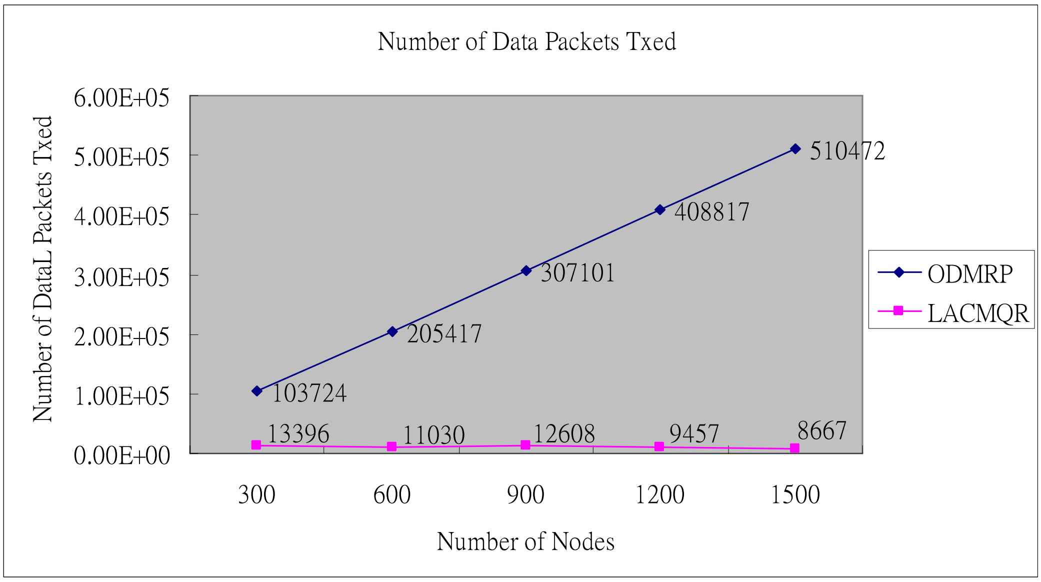


Fig. 3 The number of Data packets transmitted by ODMRP and LACMQR.

than the tree based methodology LACMQR. The considerable quantities of control packet that generated by ODMRP is resulted from the large amount of routing nodes and their flooding behavior. In LACMQR, the routing nodes are proportion to the number of clusters and limited within an upper bound that we described in theorem 1. When an intermediate node receives a PROBE packet, it relays the packet to the accurate neighbor nodes by unicasting. Because the numbers of partitions are identical, the amounts of PROBE packets are similar for different network size.

The total number of DATA packets transmitted in ODMRP and LACMQR for different network size is illustrated in Fig. 3. In ODMRP, DATA packets are transmitted by broadcast. While the member of forwarding group receives a non-duplicated DATA packet, it rebroadcasts this packet to its neighbor nodes until all neighbor nodes have received this packet the flooding stop. DATA packet may be duplicated and forwarded along different

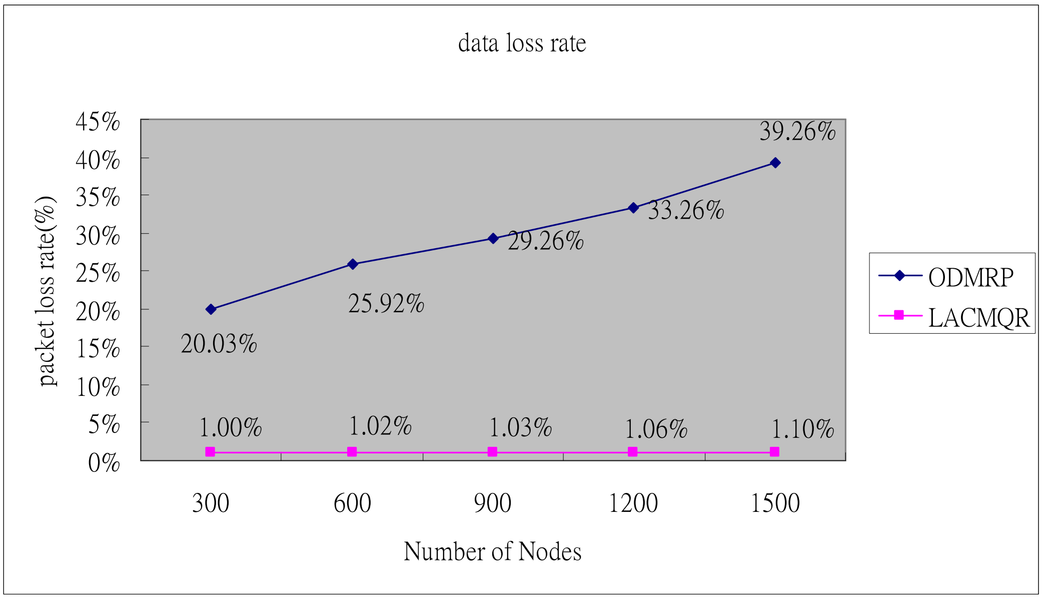


Fig. 4 Data loss rate of ODMRP and LACMQR for different network sizes.

paths. DATA packet flooding in forwarding group will generate large quantities of duplicated packets propagate in the network. It causes a high probability of collision and reduces the performance of packet transmission. In LACMQR, the DATA packets are transmitted along the constructed multicast tree by unicast. While the intermediate node receives a DATA packet, it replicates and relays the received DATA packet to the right next hop to the destinations. Each node will receive a same DATA packet only once. Hence, ODMRP produces a huge amount of duplicated DATA and exhausts a lot of resources.

Fig. 4 presents the data loss rate of ODMRP and LACMQR. In this figure, we find that the data loss rate of ODMRP is much higher than LACMQR. The high data loss rate is also resulted from the poor characteristics of ODMRP that we mentioned above. The flooding policy used in ODMRP produces a lot of duplicated packets to propagate around the network. The limited resources (e.g., bandwidth and power etc.) are mostly exhausted by those unnecessary packets.

4 Conclusion

In this paper, we proposed a distributed cluster-based multicast routing algorithm, called LACMQR. Our algorithm requires maintaining every node's local state that saves the storage and communication overhead significantly. We divided the entire network into a number of square regions called *zones* or *clusters* by the assistance of physical location information of every mobile node that get from global positioning device. The route discovery is running in source node, destination nodes, cluster heads and gateway nodes, which reduces the probing traffic significantly. The comparison of LACMQR and ODMRP was studied through extensive simulation. The simulation results reveal that LACMQR has much better performance than ODMRP. The mesh-based routes that constructed by ODMRP result in a flooding basis packet transmission. The flooding behavior of ODMRP produces considerable redundant packets that induce collision and large resources consumption. This is the main reason that ODMRP's performance is inferior to LACMQR's. The cluster head selection algorithm and gateway node selection algorithm have a great effect on the performance and the route lifetime of LACMQR. We are now trying to develop a new cluster head and gateway selection algorithm based on genetic algorithm.

References:

[1] Toomas Kirt, Leo Vohandu, “Self-Organizing Scheduling in Ad Hoc Networks,” *WSEAS Transactions on Computers*, Vol. 5, No. 9, Sep. 2006, pp. 1972-1977.

[2] E. M. Royer, C. K. Toh, “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless

Networks," *IEEE Personal Communications*, pp. 46-55, Apr. 1999.

[3] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Comp. Commum. Rev.*, Oct. 1994, pp. 234-244.

[4] C. C. Chiang, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," *Proc. IEEE SICON '97*, Apr. 1997, pp. 197-211.

[5] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *proc. 2nd IEEE Wksp. Mobile Comp. Sys. And Apps.*, Feb. 1999, pp. 90-100.

[6] J. Broch, D. B. Johnson and D. A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," *IETF Internet draft, draft-ietf-manet-dsr-01.txt*, Dec. 1998.

[7] 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, pp. 480-86.

[8] R. Dube et al., "Signal Stability based Adaptive Routing (SSA) for Ad-Hoc Mobile Networks," *IEEE Personal Communication*, Feb. 1997, pp. 36-45.

[9] Manki Min, Panos M. Pardalos, "Energy Efficient Multicasting Problem in Wireless Ad-Hoc Networks," *WSEAS Transactions on Computers*, Vol. 5, No. 11, Nov. 2006, pp. 2827-2839.

[10] M.S. Corson and S.G. Batsell, "A Reservation-Based Multicast (RBM) Routing Protocol for Mobile Networks: Initial Route Construction Phase," *ACM/Baltzer Wireless Networks*, Vol. 1, No. 4, Dec. 1995, pp. 427-450.

[11] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade, "AMRoute: Adhoc Multicast Routing Protocol," *Internet-Draft, draft-talpade-manet-amroute-00.txt*, Aug. 1998.

[12] L. Ji and M.S. Corson, "A Lightweight Adaptive Multicast Algorithm," *In Proceedings of IEEE GLOBECOM'98*, Sydney, Australia, Nov. 1998, pp. 1036-1042.

[13] J.J. Garcia-Luna-Aceves and E.L. Madruga, "A Multicast Routing Protocol for Ad-Hoc Networks," *In Proceedings of IEEE INFOCOM '99*, New York, NY, Mar. 1999, pp. 784-792.

[14] S.-J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," *Internet Draft, draft-ietf-manet-odmrp-02.txt*, Jan. 2000.

[15] C.W. Wu and Y.C. Tay, "AMRIS: A Multicast Protocol for Ad hoc Wireless Networks," *In Proceedings of IEEE MILCOM'99*, Atlantic City, NJ, Nov. 1999.

[16] E.M. Royer and C.E. Perkins, "Multicast Ad-hoc On-Demand Distance Vector (MAODV) Routing," *IETF Internet Draft, draft-ietf-manet-maodv-00.txt*, July 2000.

[17] S.K.S. Gupta and P.K. Srimani, "An Adaptive Protocol for Reliable Multicast in Mobile Multi-hop Radio Networks," *In Proceedings of IEEE WMCSA'99*, New Orleans, LA, Feb. 1998, pp. 111-122.

[18] C.-C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks," *Baltzer Cluster Computing*, Vol. 1, No. 2, 1998, pp. 187-196.

[19] C.-C. Chiang, M. Gerla, and L. Zhang, "Adaptive Shared Tree Multicast in Mobile Wireless Networks," *In Proceedings of IEEE GLOBECOM'98*, Sydney, Australia, Nov. 1998, pp. 1817-1822.

[20] Y.-B. Ko and N.H. Vaidya, "Geocasting in Mobile Ad Hoc Networks: Location-Based Multicast Algorithms," *In Proceedings of IEEE WMCSA'99*, New Orleans, LA, Feb. 1999, pp. 101-110.

[21] P. Sinha, R. Sivakumar, and V. Bharghavan, "MCEDAR: Multicast Core-Extraction Distributed Ad hoc Routing," *In Proceedings of IEEE WCNC'99*, New Orleans, LA, Sep. 1999, pp. 1313-1317.

[22] T. Ozaki, J.B. Kim, and T. Suda, "Bandwidth-Efficient Multicast Routing Protocol for Ad-Hoc Networks," *Proc. IEEE ICCCN'99*, Boston, MA, Oct. 1999, pp. 10-17.

[23] S.-J. Lee, W. Su, and M. Gerla, "Ad Hoc Wireless Multicast with Mobility Prediction," *Proc. IEEE ICCCN'99*, Boston, MA, Oct. 1999, pp. 4-9.

[24] E. Crawley et al., "A Framework for QoS-based Routing in the Internet," *RFC 2386, http://www.ietf.org/rfc/rfc2386.txt*, Aug 1998.

[25] "QoS Issues in Ad Hoc Wireless Networks," *IEEE Network Magazine*, Feb. 2001, pp. 142-148.

[26] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: A Core Extraction Distributed Ad Hoc Routing Algorithm," *IEEE JSAC*, Vol. 17, No. 8, Aug. 1999, pp. 1454-65.

[27] Shigang Chen and Klara Nahrstedt, "Distributed Quality-of-Service Routing in Ad Hoc Networks," *IEEE JSAC*, Vol. 17, No. 8, Aug. 1999, pp. 1488-1505.

[28] R. Jain, A. Puri and R. Sengupta, “Geographical Routing Using Partial Information for Wireless Ad Hoc Networks,” *IEEE Personal Communications*, Feb. 2001, pp. 48-57.

[29] J.C. Navas and T. Imelinski, “Geocast-geographic Addressing and Routing,” Proc. *ACM/IEEE MOBICOM ’97*, Vol. 3, 1997, pp. 66–76.

[30] Y. Ko and N. H. Vaidya, “Location-aided Routing (LAR) in Mobile Ad Hoc Networks,” *Proc. MOBICOM ’98*, Aug. 1998, pp. 66–75.

[31] B. Karp and H. T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” *Proc. MOBICOM’00*, Aug. 2000.

[32] L. Blazevic, S. Giordano and J. Boudec, “Self-Organizing Wide-Area Routing,” *Proc. SCI 2000/ISAS 2000*, July 2000.

[33] “Educational Observatory Institute GPS page,” *http://www.edu-observatory.org/gps/gps.html*

[34] K. Pahlavan, X. Li and J. P. Makela, “Indoor Geolocation Science and Technology,” *IEEE Communication Magazine*, Feb. 2002, pp. 112-118.

[35] C. C. Shih and T. F. Shih, “Cluster-Based Multicast Routing Protocol for MANET”, *WSEAS Transactions on Computers*, Vol. 6, No 3, March 2007, pp. 566-572.

[36] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla, “GlomoSim: A Scalable Network Simulation Environment”, *UCLA CSD Technical Report, #990027*, May 1999.

[37] R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, B. Park, H. Song, “Parsec: A Parallel Simulation Environment for Complex Systems,” *Computer*, Vol. 31, No. 10, October 1998, pp. 77-85.

Appendix A

A.1 Algorithm running at source node s:

```
While (true)
{
    if (node s needs to send a packet && route are not exist )
            send a PROBE packet to the cluster head of node s;
    Waiting to receive ACK packets || TEARDOWN packets;
    if (ACK packet received)
            connection setup success and starting to transfer data packet;
    if (TEARDOWN packet received)
            send a PROBE packet to the cluster head of node s;
}
```

A.2 Algorithm running at cluster head node c:

```
Suspend until receives a control packet CP from node p;
switch (packet type) {
case PROBE:
if (the QoS constraint is satisfied && the accumulated metric of CP is better than previous PROBE packets'
     accumulated metric){
if ( there is a route between source node and c)
    send a TEARDOWN packet to c's old predecessor ;
     let node p to be the predecessor of node c;
     update accumulated metric;
if ( c is the cluster head of the destination node || c exist routes to destinations)
    send a ACK packet to c's predecessor p;
     for (every gateway node j of c except node p)
            send a PROBE packet to c's gateway node j;
        }
else
     discard PROBE packet;
case ACK:
            if (node c has enough resource for this connection) {
                        reserve the demanded resources;
                        let node p to be the successor of node c;
                        send a ACK packet to c's predecessor;
            }
            else
                        send a FAILURE packet to c's successor;
case FAILURE:
            release reserved resources for this connection;
            send a FAILURE packet to c's successor;
case TEARDOWN:
            delete the route specifies in the TEARDOWN packet;
            send a TEARDOWN packet to c's predecessor;
case ERR:
       deletes the related routing table entry to the broken link;
            send a ERR packet to c's predecessor ;
}
```

A.3 Algorithm running at gateway node g:

```
Suspend until receives a control packet CP from node p;
switch (packet type) {
case PROBE:
if ( the QoS constraint is satisfied && the accumulated metric of CP is better than previous PROBE packets' accumulated metric){
if ( there is a route between source node and g)
      send a TEARDOWN packet to g's old predecessor ;
    let node p to be the predecessor of node g;
    update accumulated metric;
if ( g is the destination node || g exist routes to destinations)
      send a ACK packet to g's predecessor p;
    for (each adjacent cluster head node j of g except node p)
            send a PROBE packet to node j;
}
else
    discard PROBE packet;
case ACK:
    if (node g has enough resource for this connection) {
        reserve the demanded resources;
        let node p to be the successor of node g;
        send a ACK packet to g's predecessor;
    }
    else
        send a FAILURE packet to g's successor;
case FAILURE:
    release reserved resources for this connection;
    send a FAILURE packet to g's successor;
case TEARDOWN:
    delete the route specifies in the TEARDOWN packet;
    send a TEARDOWN packet to g's predecessor;
case ERR:
deletes the related routing table entry to the broken link;
    send a ERR packet to g's predecessor;
}
```

A.4 Algorithm running at destination node d:

```
Suspend until receives a control packet CP from node p;
switch (packet type)
{
 case PROBE:
  if (node d received a PROBE packet from node p with better metric)
   {
      if ( there is a route between source node and d)
   send a TEARDOWN packet to d's old predecessor ;
let node p to be the predecessor of node g;
update accumulated metric;
      send an ACK packet to the predecessor of node d;
  }

 case FAILURE :
  release reserved resources for this connection;
}
```