# 모바일 애드혹 네트워크를 위한 링 기반 멀티캐스트 라우팅 구조

허 준[†] · 홍 충 선[††] · 양 육 백[†††]


## 요 약

예상치 못한 애드혹 망의 접속형태의 변경이 동반되는 멀티캐스트 라우팅 프로토콜에 대한 연구에 많은 과제를 남겨놓고 있으며, 다양한 이동 애드혹 망에 적합한 프로토콜에 대한 연구의 필요성이 제기되고 있다. 본 논문에서는 계층적 Eulerian 링 멀티캐스트 구조를 갖는 새로운 프로토콜을 제안한다. 제안한 구조는 Eulerian 링, 계층구조, 멀티캐스트 에이전트 갖으며 기존의 방법보다 효율적이며 안전한 특성을 갖는다. 제안한 구조는 트리기반 및 메시기반 멀티캐스트 프로토콜과 비교하여 제어트래픽의 양, 점대점 지연, 패킷전송률 등에 있어 우수함은 시뮬레이션을 통해 입증한다.


# A Ring-based Multicast Routing Architecture for Mobile Ad Hoc Networks

Joon Heo[†] · Choong Seon Hong[††] · Yubai Yang[†††]


## ABSTRACT

Due to the frequent and unpredictable topology changes, multicast still remains as challenge and no one-size-fits-all protocol could serve all kinds of needs in mobile ad hoc network. In this paper, we propose a novel scheme of Hierarchical Eulerian Ring-Oriented Multicast Architecture (HEROMA) over mobile ad hoc network. It has features that concentrate on efficiency and robustness simultaneously. It is also an application-driven proposal for hazard detection. Architectures including Eulerian ring, hierarchy and multicast agent are investigated in detail. Simulation results show different level of improvements on control traffic, end-to-end delay and packet delivery ratio by comparing with tree-based and mesh-based multicast protocols.


**키워드 : 계층구조(Hierarchical), Eulerian 링(Eulerian Ring), 멀티캐스트(Multicast), 애드혹 망(Ad Hoc Network)**

## 1. Introduction

A mobile ad hoc network is an autonomous system of mobile hosts (also serving as routers) connected by wireless links, the union of which forms a communication network, modeled in the form of an arbitrary communication graph. Multicasting can be defined as a one-to-many (or many-to-many) type of communication. With multicasting, data stream from a sender is transmitted only once on link that is shared along the paths to the targeted set of destinations.

Recently, there has been an increasing demand for applications like multiplayer online gaming session through portable devices. Such applications are characterized by a close degree of collaboration, typical of ad hoc network scenarios. Multicast could prove to be an efficient way of providing necessary services for these kinds of applications. Due to the frequent and unpredictable topology changes, multicasting still remains as challenging issue. Many efforts have been made to provide an efficient multicast protocol over mobile ad hoc network. For simplicity, four main categories are classified [1], tree-based, meshed-based, statelessness and hybrid protocols.

Tree-based multicast is a well-established concept developed from wired network. It could provide high data forwarding efficiency at the expense of robustness. Mesh-

※ 본 연구는 정보통신부 ITRC Project 지원에 위해 수행되었음.
† 준 회 원 : 경희대학교 대학원 컴퓨터공학과
†† 종신회원 : 경희대학교 전자정보학부 교수
††† 준 회 원 : 경희대학교 대학원 전자정보학부
논문접수 : 2004년 9월 21일, 심사완료 : 2004년 12월 9일

based multicast could provide better robustness at the expense of higher forwarding overhead. Stateless multicast explicitly mentions the list of destinations in the packet header, And hybrid multicast tries to combine the advantages of both tree and meshed-based approaches.

A mobile ad hoc network, as defined in [2], is a network in which each node plays the dual role of data originator and data routers. Detecting the environmental hazards and monitoring the remote terrain are among the significant mobile ad hoc network applications. In case of fire detection, it is significantly valuable to monitor fire-spot's shape and trend in time. Mobile ad hoc nodes, right around the fire-spot are responsible for sensing, processing and networking packets or even taking actions to launch extinguishers. Thus a ring-oriented multicast architecture is more attractive than traditional tree-based or mesh-based architecture for such applications.

Accordingly, in this paper, we propose a novel scheme of Hierarchical Eulerian Ring-Oriented Multicast Architecture (HEROMA) over mobile ad hoc network. It has features that concentrate on efficiency and robustness simultaneously. Architectures including Eulerian ring, hierarchy and multicast agent are investigated in detail. Simulation results show different level of improvements on control traffic, end-to-end delay and packet delivery ratio by comparing with tree-based and mesh-based multicast protocols.

## 2. Related Works

Recent years have witnessed the booming of protocols proposed for mobile ad hoc networks. According to how routes are created to the members of the group [1], they could be classified as tree-based approach, mesh-based approach, stateless multicast and hybrid approach.

- Tree-based multicast is a very well established concept in wired networks. Most schemes for providing multicast in wired networks are shared-tree-based. Different researchers have tried to extend the tree-based approach to provide multicast in a mobile ad hoc environment. It could provide high data forwarding efficiency at the expense of robustness.
- Mesh-based multicast may have multiple paths between a source and receiver pair. Existing studies show that tree-based protocols are not necessarily best suited for multicast in mobile ad hoc network where topology changes frequently. In such environment, mesh-based protocols seem to outperform tree-based proposals due to the availability of alternative paths, which allow multicast packets to be delivered to the receivers even if links fail.
- Stateless multicast is proposed wherein a source explicitly mentions the list of destinations in the packet header. In mobile ad hoc environment, frequent movement of mobile nodes considerably increases the overhead in maintaining the delivery tree/mesh. To minimize the effect of such a problem, stateless multicast focuses on small group multicast and assumes the underlying routing protocol to take care of forwarding the packet to respective destinations based on the addresses contained in the header. If mobile nodes move with the great velocity, the stateless multicast is a good choice because the tree-based and mesh-based approaches can not catch up with the topology change. But stateless multicast suits for only small-multicast groups operating in rather dynamic networks.
- Hybrid multicast protocols provide a way of discussing efficiency and robustness simultaneously. The tree-based approaches provide high data forwarding efficiency at the expense of low robustness, so it is not suitable for high mobility environment which is prone to the link breakage. Whereas mesh-based approaches provide better robustness (link failure may not trigger a reconfiguration) at the expense of higher forwarding overhead and increased network load. Thus, a hybrid multicasting solution is more attractive because it has a better performance by combining the advantages of both tree and meshed-based approaches.

## 3. Proposed scheme

We employ the Fisheye State Routing (FSR) protocol as our IP protocol, with which we can get knowledge within predefined-scope. Details could be found in [6]. FSR introduces the notion of multi-layer fisheye scope to reduce routing update overhead in large-scale networks. Mobile nodes exchange link state entries with their neighbors with a frequency that depends on distance to destination. With the help of FSR as IP protocol, our proposal could get the necessary information within predefined scope which is the basis of searching algorithm for hierarchical Eulerian ring.

The Hierarchical Eulerian Ring-Oriented Multicast Architecture (HEROMA) contains three main features, Eulerian ring, hierarchy and multicast agent that will be presented in the following sections.

### 3.1 A novel Eulerian ring based scheme

Let G be a planar graph, stands for a group of mobile nodes in ad hoc network. Nodes in G are denoted by V(G). Links between nodes in G are denoted by E(G) (we discuss only bi-direction link herein). The degree of V is the number of edges meeting at V, and is denoted by deg v. If there is a closed trail that includes every edge one and only then such a trail is called a Eulerian trail. If it forms a ring then it is called a Eulerian ring. Proofs of theorems are omitted and details could be found in [7]. The reason that we employ Eulerian Ring is because it is edge-traceable. In Eulerian graph we could walk along every link once and only once.

Now, first we make the following definitions for terminology and rule :

**Theorem 1** : Let G be a connected graph, then G is Eulerian if and only if every vertex of G has even degree.

Center node : which provides FSR link-state information of the region.

Multicast receiver : node which receives multicast packets.

Ei, j : a link connects node i and node j.

Leaf node : a node whose deg v = 1.

Even node : node with even degree.

Odd node : node with odd links.

**Rule 1** : Let neighbor take leaf node's task (note that leaf node has only one neighbor), then delete leaf nodes. Execute it repeatedly until there is no leaf node in G. Calculate degree then mark node type (even node and odd node).

2-odd-link : link that connects 2 odd nodes.

2-even-link : link that connects 2 even nodes.

Odd-even-link : link that connects one odd node and one even node.

**Rule 2** : Mark link type (2-odd-link, 2-even-link or odd-even-link). Maintain 2-even-link unchanged, prune 2-odd-link in descending degree (from maximum degree to minimum degree), and is denoted by freeze-link. Execute rule 2 repeatedly until no 2-odd-link exists.

**Theorem 2** : In graph G, the number of nodes of odd degree is even.

After rule 2, only odd-even-links remain. And odd nodes will appear in pairs.

**Rule 3** : Mark odd node pairs in descending degree. Find path connected them in descending way, mark it recursive-freeze-link and prune it. If new odd node generates, execute rule 3 recursively until no odd-node-pair exists.

Main ring : ring contains center node.

Assist ring : ring other than main ring but connects to main ring, it receives multicast packets from diverge nodes.

Branch : a tree or line connects to ring, it contains at least one multicast receiver and have a joint to a ring.

Master : a node that takes charge in multicast service for his ring or branch. Each ring, assist ring and branch should have its own master respectively. The master of assist ring is also referred to as sub-master.

Member : node that takes part in multicast affairs, including multicast member and forwarding node.

Diverge node : a node resides on joint of rings or joint of ring and branch. It duplicates multicast packet for the lower ring or branch.

Bridge : the only link without which graph will be disconnected.

δ : A time-out setting to preventing situations in which a node has waited too long for a token. For example, due to crash failure of node or link, a message taking time units longer than δ for transmission is considered as having been lost.

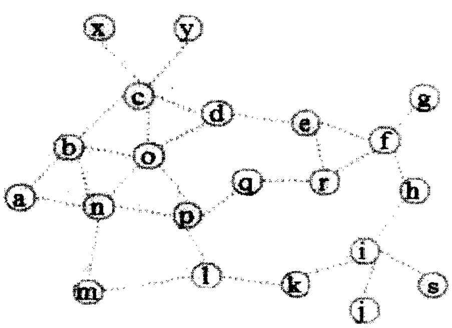
**Theorem 3** : If G is a graph in which every vertex has even degree then G can be split into disjoint cycles such that no two cycles have any edges in common.

**Rule 4** : Execute the Fleury algorithm [7] then split it into small circles according to theorem 3, construct ring-based topology by marking main ring, and assist ring and branch. Recover the freezelink and recursive-freeze-link, make them part of branch or assist ring if it contains member and so does the leaf node as generated in Rule 1.

In general, the procedure of searching Eulerian ring-based topology includes 6 steps. To clarify how the rules work, we utilize a randomly generated (Figure 3-1) as our original figure.

Step 1 : Get topology information with the help of FSR, Identify center node, multicast receivers and source. A one-to-many multicast service is discussed herein.

(Figure 3-1) shows the topology from the view of node p within certain scope, say 4 hops. Herein 4 hops is the number predefined by FSR. In (Figure 3-1), Node p is center node, {x, c, p, e, r, s} are multicast receivers.

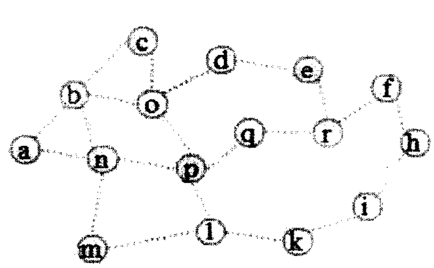



(Figure 3-1) Original topology provided by FSR

Step 2 : Execute rule 1 and rule 2.

According to rule 1, the leaf node should be deleted. Before deleting leaf nodes, let neighbor take leaf node's multicast affair. Thus {x, y, g, j, s} are pruned.

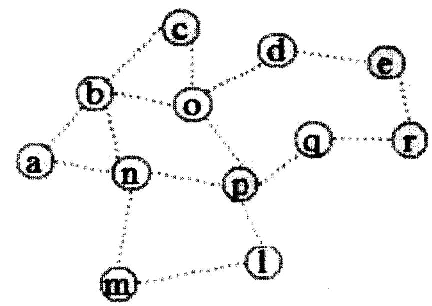
According to rule 2, {o, n, c, d, e, f, r, l} are marked as odd nodes. It should be noted that deg(c) is 3 rather than 5 because contribution of leaf nodes are counted out. We will start from node with maximum odd degree, herein {o, n} are nodes with maximum odd degree of 5, so we mark link (o, n) freeze-link. The same thing happens to Ec, d and Ee, f. Up to now, the center node based ring {p, q, r, e, d, o} has already formed. Thus after rule 2, (Figure 3-2) is generated.




(Figure 3-2) Topology after rule1 and rule2

Step 3 : Execute rule 3.

Only one odd-node-pair remains, pair {l, r}. Find a path connecting them in way of descending degree, so the path {l, k, i, h, f}is selected and is denoted by recursive-freeze-link, prune this path. Thus (Figure 3-3 is generated.




(Figure 3-3) Topology after rule3

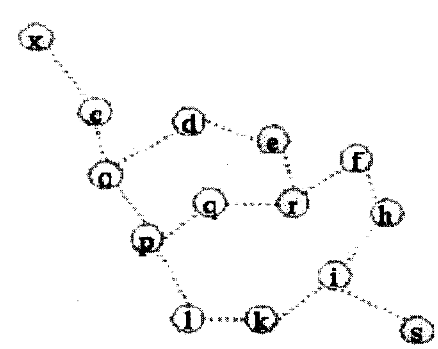
Step 4 : Choose the center node as starting node during traversing any available edge. Choosing a bridge only if there is no alternative. Stop when there are no more edges.

(Figure 3-3) is a Eulerian graph and step 4 is the method called FLEURY'S algorithm [7].

Step 5 : Split the big Eulerian ring into smaller ones according to Theorem 3.

Step 6 : Execute rule 4.

Mark main ring, assist rings, mark freeze-links and recursive-freeze-links as branches if they contain members. In order to cover all multicast receivers, we form one main ring {p, q, r, e, d, o}, recover path {o, c, x} and { l, k, i, f, h}. To enhance robustness, another ring {p, q, r, f, h, i, k, l} is generated as assist ring. Path {o, c, x} is generated as branch.




(Figure 3-4) Hierarchical Eulerian ring-oriented topology

After 6 steps aforementioned, a hierarchical Eulerian ring based architecture emerges (shown in (Figure 3-4)). It should be noted that the Eulerian ring is not unique since the procedure of rule2 and rule3 are not unique. Different selection of freeze-link or recursive-freeze-link will result in different sub-graph. Our goal is to form such a ring and branch based architecture which contains all multicast receivers.

### 3.2 Hierarchy

The second characteristic in HEROMA is hierarchy, which is achieved by the topology of different level Eulerian rings and branches. The hierarchical architecture has 2 advantages such as multicast routing update and the control traffic that are restricted locally to reduce delivery overhead. On the other hand it may shorten subscription time by registering to a nearer multicast receiver rather than a remote multicast source. As shown in (Figure 3-4) the main ring {p, q, r, e, d, o} is the first level. Branch {o, c, x} and assist ring {p, q, r, f, h, i, k, l} are the second level.

To achieve hierarchy we employ master and diverge nodes. Each ring and branch would have a master. Master of main ring is honored to manage its members including diverge nodes. The diverge node stores information of assist ring or branch in lower level. Assist ring master play the same role in their own rings as main ring master. Both master and diverge node are the basis of hierarchical architecture. Note, each node belongs to one but only master from whom it gets multicast packets. Diverge node locating at the joint point has dual duty, one is to do the work of its own ring such as forwarding multicast packets, relaying token in appropriate direction and reporting error message. The other is to establish hierarchical level by transferring multicast packets to lower level.

### 3.3 Multicast Agent (MA)

Each mobile node should maintain a Multicast Agent (MA). The architecture of MA is similar to [10], but the two important elements : status table (ST) and token daemon are added. (Figure 3-5) shows the architecture of multicast agent. For each master, ST keeps the information of its member nodes including diverge nodes. Token daemon manages token that flows along the ring. The Status Table (ST) of master includes parameters as follows : member, diverge node and multicast information for receiving and forwarding multicast packets. In case of diverge node, it should maintain information of successor, predecessor and master, as well as extra information of lower level.

#### 3.3.1 Token Daemon

Token daemon manages token that flows along the ring. In HEROMA, we share the advantage of wireless token ring protocol [11]. A modification is made on token structure shown in <Table 3-1>.

〈Table 3-1〉 Token Structure

| PT | RA | DA | SA | Seq | GenSeq | Rank |
|---|---|---|---|---|---|---|

As defined in [11], PT stands for packet type, such as Token or multicast data. RA means the ring's address defined uniquely by the master. DA and SA means destination and source address. At diverge node DA is replaced by successor node. Sequence number is initialized to a given number and increased by every node that passes the token. Its main function is to manage and recover ring. Genseq number is initialized to a given number and increased at every rotation of the token by the master. Rank is a novel option different from [11], and it is used to maintain the hierarchical level. The master utilizes token to gather members' information and inform them whom they belong to. After some interval, master should release a token to flow along the ring in a given direction. On hearing the token, node reports its own information to master as it passes the token to successor. In case of token missing or timeout that caused by link breakage, master is supposed to release a token again. In HEROMA nodes rely on implicit acknowledgement to monitor the success of their token transmissions because implicit signaling would save bandwidth. An implicit acknowledgment could be any packet with the same ring address in its header. If no token detected after a given time, other nodes could claim to be master by changing the RA of the token which is being passed around.

### 3.4 Master Election

The algorithm of electing master should be carefully considered because it is the master's duty to perform multicast task and ring recovery. The factors such as battery, computation capacity, mobility, bandwidth and bit-error-rate (BER) are important parameters to be considered. In order to simplify our model, we suppose each mobile node has same computation capability and choose only 2 parameters for master election such as average error packets rate and battery capacity. The former is a kind of BER and proposed by [8]. The latter emphasizes the lifetime of power and it is proposed by [9]. These two variables are QoS-aware routing metrics that could be easily obtained without any special extra hardware supports.

**New_P** $^{k}_{e[i,j]}$ is defined as an average error packet within sampling period K for a given direction-oriented wireless link from i to j, **P** $^{k}_{e[i,j]}$ as average error packet after sampling period K for a given direction-oriented wireless link from i to j.

$$Pe^{k}_{[i,j]} = (\mu \times Pe^{k-1}_{[i,j]}) + (1-\mu) \times New\_P^{k}_{[i,j]}) \quad i,j \in V, 1 > u > 0 \qquad (1)$$

**LT** $^{k}_{i}$ is defined as life time estimated after sampling period. **LT** $^{k}_{i}$ is calculated as :

$$LT^{k}_{i} = \rho^{-1}_{i} \times (F_i / E_{i(t)})^{-\alpha} \qquad (2)$$

Herein Ei(t) means remaining battery capacity of node i at time t ; Fi is full-charge battery capacity of node i ; ⓑ is a positive weighting factor for ratio of the remaining i stands for the transmit power at node i at time k.

We define Pi as the possibility that a node has to become a master :

$$P_i = (LT^{k}_{i}) \times (\sum Pe^{k}_{[i,j]}/N) \qquad (3)$$

$\sum P^{k}_{e[i,j]}/N$ means the average error rate among links processed by node i, and **LT** $^{k}_{i}$ indicates the live time that Ni maybe last after sampling period, the longer live time means the higher possibility to be elected as master.

### 3.5 Algorithm of Searching Eulerian Ring

When mobile nodes scale to large population, a distributed manner is more attractive. In order to generate Eulerian ring, searching algorithm will launch separately in different region at the initial stage. Then recovery algorithm may play a role if link breakage occurs. After different Eulerian ring structure formed respectively, merging algorithm becomes necessary. So in this part searching and recovery algorithm will be discussed first, then follows recovery and merging algorithm.

The searching algorithm for Eulerian ring, denoted by S1 PROCEDURE, will launch for Eulerian ring-oriented topology at the initial stage.

**PROCEDURES1 :**

```
Repeat
if (∀NiV(G), ∃ deg(Ni) = 1)
  {
   mark Ni leaf node ;
        let neighbor take Ni's task ;
   prune Ni ;
  }
  end Repeat
  If (∀Ni ∈ V(G), ∃ deg(Ni) mod 2 = 0)
  { mark Ni even-node ; }
  else {mark Ni odd-node ; }

  Repeat
  If (∀Ei ∈ E(G), Ei = 2-odd-link)
   {
  mark Ei 2-odd-link ;
  search Ei with Max deg (Ei) ;
  mark Ei freeze-link ;
  prune Ei ;
  call PROCEDURE S1-R1 ;
  }
  end Repeat
  If (∀Ei ∈ E(G), Ei = odd-even-link)
    { mark Ei odd-even-link ; }
  If (Ei ∈ E(G), Ei = 2-even-link)
    {∀mark Ei 2-even-link ; }

  Repeat
  If (∀Ni ∈ V(G), Ni = odd node )
  {
  if (deg(Ni) = max)
  {
  search Ni whose deg(Ni) is max ;
  search odd node Nj nearby ;
    find path connect Ni and Nj ;
    mark path recursive-freeze-link then prune it ;
    call PROCEDURE S1-R2 ;
  }
  else{ search Ni with max degree ; }
  }
  end Repeat

  Launch Fleury algorithm from center node ;
  T(G) : trail for FLEURY's algorithm,
  T(G) = 0 in the initial stage ;
  Repeat
  If (∃Ei, j ∈ E (G) ∧ Ei, j ∉ T(G))
  { if (Ei, j ≠ bridge)
   {
     T(G) = T(G) + Ei, j ;
     E(G) = E(G) - Ei, j ;
     Ni = Nj ;
            Prune Ei, j ;
   }
  else
   { if (∃!bridge)
    { search other Ei, j ; }
    else {
```

```
            T(G) = T(G) + Ei, j ;
            E(G) = E(G) - Ei, j ;
            Ni = Nj ;
                  Prune Ei, j ;
        }
  }
}
end Repeat
if (∀Ni ∈ T(G), ∃Ni = joint node)
  { split T(G) at Ni into Circle_i ; }
  if (center node ∈ Circle_i)
  { mark it main ring ; }
  if (∀Nim ∈ ulticast member, ∃Ni ⊂ freezelink)
  {
   Recover the freezelink ;
      mark diverge node ;
   mark branch ;
  }
```

In our proposal, adding or pruning a node will not affect the multicast service because the ring topology is fault-tolerance for single node failure or link breakage. But it does destroy integrity and might fragment Eulerian ring into pieces. Therefore a recovery algorithm is proposed.

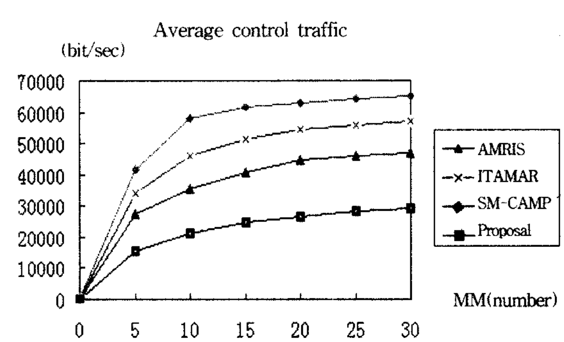
After locating the position of failure, we define two coordinators if two nodes locate nearest to failure. The key idea of recovery algorithm is to search a bridge to recover. Two coordinators should send a recovery-REQ message. Other intermediate node will relay this message until a bridge is found. If multicast group is of small size then the multicast source will be the center node and launch the searching algorithm according to its own FSR information. If multicast group scale to large population then a distributed manner is more favorable. In different regions, center node will use its topology information provided by FSR and launch searching and recovery algorithms respectively. Then merge procedure is necessary for different Eulerian ring if they belong to the same multicast group.

## 4. Simulation Results

The simulations are conducted by using OPNET Modeler v8.0. Performance is evaluated in terms of end-to-end delay by comparing with Ad hoc Multicast Routing Protocol Utilizing Increasing id-numbers (AMRIS), Core-Assisted Mesh Protocol (CAMP) and Independent-Tree Ad hoc Multicast Routing (ITAMAR). Mobile hosts are placed randomly within a 1000m*1000m area and moved randomly. Radio propagation range is 250m. The number of mobile host is 40 and member of multicast group increases from 0 to 30, ranging in the set {0, 5, 10, 15, 20, 25, 30}. This is to simulate classroom scenario ranging from lecture mode to videoconference mode. Multicast sources are randomly chosen among these mobile hosts. The member nodes join the multicast session at the beginning of the simulation and remain as members throughout the simulation. The channel capacity is 2Mbits/s and buffer size is 64000 bits. FSR is used to exchange link-state information as IP layer, thus topology information could be obtained in finite time. A simple topology of 2-level-hierarchy was taken as reference for our scheme. We use a burst source to model higher-layer traffic generation.

### 4.1 Average Control Traffic

In this scenario, HEROMA is compared with Ad hoc Multicast Routing Protocol Utilizing Increasing id-numbers (AMRIS) [3], Core-Assisted Mesh Protocol (CAMP) [5] and Independent-Tree Ad hoc Multicast Routing (ITAMAR) [4] by using average control traffic. The simulation results are shown in (Figure 4-2). The X-axis in these figures stands for number of multicast members (MM) ; the Y-axis in (Figure 4-2) stands for average control traffic and its unit is bits/sec. In (Figure 4-2), our proposal has an average control traffic around 25000 bit/s while that of ARMIS is near 40000 bit/s, that of ITAMAR is 50000 and that of CAMP reaches 60000 bit/s. Our proposal performs much better than the comparing schemes in terms of control traffic.




(Figure 4-2) Average control traffic

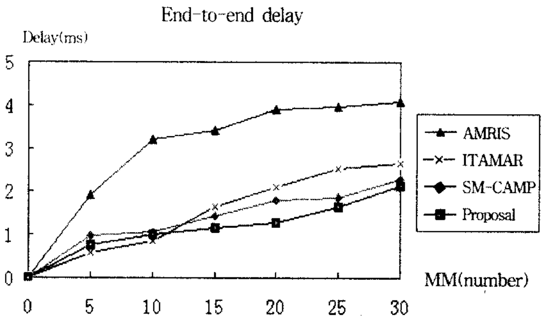
According to (Figure 4-2), we could also find CAMP shows a larger control traffic sent/received than other schemes. CAMP is a typical mesh-based multicast protocol and it is commonly accepted that mesh-based multicast scheme has low efficiency than tree-based multicast in terms of control traffic. It is dependent upon the unicast routing protocol for behaviors regarding network convergence and control traffic growth in the presence of mobility, so CAMP responses to link breakage is not immediate. It may incorrectly deduce a link breakage in the presence of high network load on control traffic. CAMP relies on the unicast routing protocol which sends triggered updates. It suffers from exponential growth in control traffic overhead under increasing mobility. Moreover, CAMP piggybacks its own update message onto WRP and those packets play a role in overhead growth.

AMRIS is a typical tree-based multicast protocol that constructs a shared multicast delivery tree. The advantage of AMRIS is its high date forwarding efficiency comparing with mesh-based schemes. ITAMAR is also a tree-based architecture while it could switch to an alternative tree, which is called Independent-tree in case of link breakage. Alternative tree is tolerant for fails because it could switch to other path quickly, while extra 25% control traffic is needed for maintaining the alternative tree.

Our proposal-HEROMA outperforms these three schemes in fields of average control traffic significantly. The most important reason is that center node gets topology information of certain range provided by FSR which deals with the topology change efficiently. So, less control traffic could be obtained at the cost of computation capacity for the searching algorithm ; the other reason lies in the hierarchical topology, with which control traffic is restricted locally in our proposal.

#### 4.2 End-to-End Delay

In this scenario, end-to-end delay for data and control traffic is tested and result is presented in (Figure 4-3). The X-axis in (Figure 4-3) stands for the number of multicast member (MM). The Y-axis stands for end-to-end delay in average and its unit is ms. End-to-end delay of our proposal increases smoothly, ranging from 1 ms to 2 ms as the number of multicast member increases. The end-to-end delay of ITAMAR is slightly lower than our proposal in small-multicast population and slightly higher while population scales large. The end-to-end delay of CAMP is almost the same as our proposal with a little higher delay. From (Figure 4-3), we could also see AMRIS performance is not good, and end-to-end delay of AMRIS is ranging from 2ms to 4ms that is almost two times of the other three schemes.



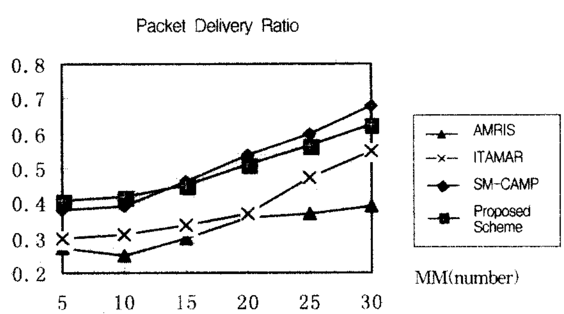

(Figure 4-3) End-to-end delay

AMRIS detects link disconnection by a beaconing mechanism. It suffers from collision when the transmission and size of beacon increase. Especially it is lack of robustness due to the tree topology having no redundant paths. In AMRIS, only the packets from the registered parent or registered child will be forwarded. If the tree link breaks then the packets are lost until the tree is reconfigured. Thus, the recovery time will result in a long end-to-end delay. The other three schemes : ITAMAR, CAMP and our proposal all have alternative path and tolerant for failures such as link breakage. Thus, a less end-to-end delay could be achieved.

In CAMP, number of redundant paths to a distant destination is fewer than that close to the center of the mesh. The mesh becomes massive with the growth of the multicast members. This means CAMP is more prone to recovery from failures as the mobility increases. CAMP performs well at end-to-end delay because meshes could generate more redundant paths when the population of multicast member is increased. ITAMAR performs best at end-to-end delay when multicast members are in small number. ITAMAR could find disjoint-link quickly for independentree in this case. When multicast members scale to large number, the end-to-end delay of ITAMAR is a little higher than that of our proposal and CAMP. The reason is that disjoint-link for independent-tree is lack and one independent-tree as backup tree is not enough for recovery.

(Figure 4-3) shows our proposal exhibits a good performance when compared with CAMP, ITAMAR and AMRIS in terms of traffic end-to-end delay. The most possible reason is that our proposal is tolerant for failures such as link breakage. On the other hand, the hierarchical architecture could deliver multicast packets to receivers simultaneously in an efficient way and shorten subscription time for new multicast member. Thus, a better performance on end-to-end delay is achieved.

The routing robustness and effectiveness as a function of multicast group size is illustrated in (Figure 4-4). The X-axis in (Figure 4-4) stands for the number of multicast member (MM). The Y-axis stands for packets delivery ratio. CAMP performs the best as the number of multicast increases. Since the mesh becomes massive with the growth of the members, more redundant routes are formed which improves the performance. If only a small number of nodes join the multicast session then the mesh actually appears closer to a tree for distant nodes and the performance is shown in (Figure 4-4). Our proposal performs slightly better than CAMP in small multicast groups and a little worse than CAMP when the number of multicast increase. Our proposal gets the benefits of FSR and establishes redundant routes optimally in small population. But, the recovery algorithm and spare route are not efficient comparison with massive meshes in CAMP. AMRIS also shows improvements with the member size growth, but it is less dramatic than CAMP and our proposal because redundant routes are not established in ARMIS. ITAMAR improves this shortcoming by establishing a redundant tree and shows a better performance than ARMIS to some extent.




(Figure 4-4) Packet delivery ratio

## 5. Conclusion

In this paper, a novel model of Hierarchical Eulerian Ring-Oriented Multicast Protocol (HEROMA) is proposed. A graph based multicast schemes including hierarchical architecture, multicast agent and relative algorithms have been discussed. Simulation results indicate that HEROMA outperforms ARMIS, CAMP and ITAMAR in terms of service latency and robustness to some extent. We hope our HEROMA could shed a little light of graph-oriented multicast scheme over mobile ad hoc network. However, this paper is limited to the scenario of small scale, our future work will focus on optimizing algorithms for large multicast group size.

## References

[1] Carlos de Morais Cordeiro, H. Gossain and D. P. Agrawal, "Multicast over Wireless Mobile Ad Hoc Networks : Present and Future Directions," IEEE Network, Vol.17, No.1, Jan., 2003.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on Sensor Networks," IEEE Communications Magazine, 0163-6804, August, 2002.

[3] C. W. Wu, Y. C. Tay, "Ad hoc Multicast Routing Protocol Utilizing Increasing id-numbers (AMRIS)," In proceedings of IEEE MILCOM'99, Atlantic City, NJ, Nov., 1999.

[4] S. Sajama and Z. J. Haas, "Independent-Tree Ad hoc MulticAst Routing (ITAMAR)," Mobile Networksand Applications, Vol.8, pp.551-566, October, 2003.

[5] E. L. Madruga and J. J. Garcia-Luna-Aceves, "Scalable Multicasting : The Core Assisted Mesh Protocol," ACM/Baltzer Mobile Networks and Applications Journal, Special Issue on Management of Mobility in Distributed Systems, 2001.

[6] Guangyu Pei, M. Gerla and T. W. Chen, "Fisheye State Routing in Mobile Ad hoc Network," Proceedings of the IEEE International Conference on Communications, New Orleans LA, pp.70-74, June, 2000.

[7] R. J. Wilson and J. J. Watkins, Graphs : an introductory approach, John Wiley & Sons, Inc, pp.122-127.

[8] Ying chun, liu yong, shi meilin, "A hybrid protocol in Ad hoc network," 2001 journal of software, China, 2001

[9] M. Maleki, K. Dantu and M. Pedram, "Power-aware Source Routing Protocol for Mobile Ad hoc Network," International Symposium on Low Power Electronics and Design, 2002 IEEE, 2002

[10] Behrouz A. Forouzam, Sophia Chung Fegan, TCP/IP Protocol Suite, McGraw-Hill Companies, Inc, chapter 10.

[11] Duke Lee, Anuj Puri, Pravin Varaiya, Roberto Attias, "A Wireless Token Ring Protocol for Ad-hoc Networks," IEEE Aerospace Conference Proceedings, Montana, USA, 2002.

**허 준**

e-mail : joon@networking.khu.ac.kr
2002년 경희대학교 컴퓨터공학과(학사)
2004년 경희대학교 컴퓨터공학과(공학석사)
2004년~현재 경희대학교 컴퓨터공학과 박사과정
관심분야 : 센서네트워크 보안, Ad hoc 네트워크, 네트워크 보안

**홍 충 선**

e-mail : cshong@khu.ac.kr
1983년 경희대학교 전자공학과(학사)
1985년 경희대학교 전자공학과(공학석사)
1997년 Keio University 정보통신전공 (공학박사)
1988년~1999년 한국통신 통신망연구소 선임연구원/ 네트워킹연구실장
1999년~현재 경희대학교 전자정보학부 부교수
관심분야 : 인터넷 서비스 및 망 관리, 네트워크보안, 모바일 네트워킹

**양 육 백**

e-mail : yang_yubai2001@yahoo.com.cn
1996년 Beijing University of Posts and Telecommunications(학사)
2004년 경희대학교 컴퓨터공학과(공학석사)
1996년~현재 Chinamobile Jilin Filiale Sr. Engineer
관심분야 : Ad hoc network, Routing Algorithm