QoS-Aware Multicast Protocol Using Bounded Flooding (QMBF) Technique

Zhi Li and Prasant Mohapatra
Department of Computer Science
University of California at Davis
Davis, CA, 95616

***Abstract*— Many multicast applications, such as video-on-demand and tele-education have quality of service(QoS) requirement from the underlying network. Recently, many QoS-based multicast protocols have been proposed to meet these requirements. However, few of them can achieve high success ratios. In this paper, we propose a new QoS-based multicast protocol, QoS-aware Multicast Protocol using Bounded Flooding (QMBF) technique. Every network node has local network cell topology information as well as QoS state information. The QMBF utilizes this information to increase the chance of finding the feasible branch. It bases on two methods to find a feasible branch: Computing out partial feasible branches using local network cell information (collected from bounded flooding messages) and multiple path searching. The design of QMBF allows it to operate on top of any unicast routing protocol or cooperate with a QoS-based unicast routing protocol.**

I. Introduction

Several multicasting routing protocols have been proposed in the literature with varying performance, cost, and implementation [6][10]. However, most of the proposed multicast protocols are QoS-oblivious. They only use the available best-effort unicast routing protocol to find paths from the sender to the receivers without considering the members' QoS requirements. On the other hand, it is known that most of the multicast applications are inherently QoS-sensitive. They usually have a requirement for QoS measure, such as delay, bandwidth, loss characteristics, etc. In order to provide satisfactory QoS to the applications, different mechanisms have been proposed, such as RSVP (resource reservation protocol), differentiated services, MPLS, etc [12]. The goal of QoS-based multicast routing is to search and construct a multicast tree that not only covers all the group members but also meets their QoS requirements.

The basic requirements of QoS-based multicast routing are: 1) Scalability to large groups; 2) Support for dynamic membership; 3) Support for receiver-initiated, heterogeneous reservations [7]. It is known that QoS-based multicast routing satisfying heterogeneous QoS requirements is NP-hard [9], because of which, most QoS-based multicast routing protocols utilize heuristic methods in the *feasible branch* (QoS-satisfying branch) searching process.

QoS-aware Multicast Protocol using bounded flooding (QMBF) technique combines the merits of source-based routing and the QoS-aware routing [2] in the feasible branch searching process. Every node broadcasts its local QoS state and least-cost unicast reaching information (derived from unicast routing information) when either of them changes over some threshold. This broadcast information is valid for some bounded hops. Every node can gather its neighbors' broadcast message and have knowledge of local network cell's topology, routing information as well as QoS states. When a multicast join request message arrives at one node, it takes two steps to finish the local partial branch search. First, if some edge nodes of local network cell have least-cost paths toward the target router, it can compute feasible paths toward these edge nodes and forward the request to them along the feasible path. If this step fails, the node will compute feasible paths toward all the edge nodes and forward the request to all the edge routers along the feasible paths. Using this mechanism the routing protocol can greatly increase the robustness and accuracy of finding QoS-satisfied paths.

This paper is organized as follows. Section II presents the related work. Section III introduces the basic idea of QMBF. Section IV describes the details of the QoS-aware multicast protocol. The protocol analysis and simulation are presented in Section V. Finally, we draw the conclusion in Section VI.

II. Related Work

Several QoS-based multicast protocols have been proposed: [2]-[4]. Some of these protocols use multiple branch searching method to increase the probability of successful search, like [1][3][5]. Their branch searching processes are based on least-cost, which may not always satisfy the QoS requirements. Chen et al have integrated QoS-awareness idea into the branch searching process [2], where the feasible branch searching is based on unidirectional broadcasting if the least-cost path can not satisfy the QoS requirements. In spanning-joins [1], a new member broadcasts join request in its neighborhoods to find on-tree routers. The reply message will collect the QoS properties along its traveling path, which is one of the candidate paths. When the new member receives multiple reply messages, it selects the best candidate path as a multicast tree branch to join the group.

The QoS sensitive multicast Internet protocol (QoSMIC) uses a "manager router" to construct shared multicast trees [3]. The new router has two ways to find an on-tree router to join a multicast group: local search and multicast-tree search. The local search period is the same as the spanning-joins. In multicast-tree search period, the host router sends a join request to the manager. The manager selects some on-tree routers as candidate routers, which will unicast bid messages towards the host router. After receiving the bid messages, the host router selects the best path to connect to the multicast tree.

The QoS-aware multicast routing protocol (QMRP) [2] constructs a shared tree by unicasting a request message from the host router toward the core router (or source router). If a router in the unicast path does not satisfy the QoS requirements, the request message is replicated and sent out to all other neighbors of the router. It introduces the idea of QoS-awareness into the path selection period, which increases the ability of finding a feasible branch. However, it requires temporal state in the network routers for each join request. It is only applicable for applications with non-additive QoS requirements such as bandwidth and buffer space, and cannot be used for additive requirements such as delay or packet loss.

Our protocol shares the same merit as QMRP, using distributed QoS-aware mechanism to search for a feasible branch. We also use the M-hops bounded broadcasting mechanism to make the feasible branch searching process more focused. Using this mechanism, every router has the knowledge of local network cell information, which can increase the chance of finding feasible branches. Another advantage is that the protocol also supports additive QoS properties without involving the backtrack process and keeping temporary states.

III. QMBF:BASICS

A. Bounded Flooding and Feasible Branch

When a new member wants to join a multicast group, QoS-based multicast routing protocols should have the ability to find a feasible branch that meets the new members' QoS requirement. Source-based QoS routing can find all the possible feasible branches because every node has the knowledge of global network topology and QoS state information. However, the scalability problem prevents it from wide implementation.

In QMBF, we use *M-hop bounded flooding* to increase the chance of finding feasible paths. The protocol is based on current least-cost based routing architecture. We assume that every node has the QoS state information of itself and its outgoing links (*local QoS state information*), which includes the available bandwidth, average delay, etc. *M-hop bounded flooding* requires every node to periodically broadcasts its local *QoS state information* and *unicast reachable information* (which nodes it can reach through each of its neighbor nodes, computed from unicast routing information) for at most for M hops (we can choose different value of M based on network size and status as discussed later). From these messages, every router can have a view of partial local network topology and QoS state information as well as the reachable information. We call this small domain of the network as *local network cell (LNC)*. The node itself is the center of its LNC whose radius is M hops. Using this information, every node can accurately and quickly direct the QoS-based multicast routing request.

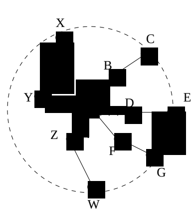

Fig.1. Illustration of bounded flooding

Let us consider the example shown in Figure 1 which shows the LNC of node A after it gets neighbors' broadcast messages (M=2 in this topology). We term X, C, E, G, W and Z as *edge nodes*, which are the last valid hop nodes of A's broadcast messages. All the edge nodes' reachable information constitute the LNC's least-cost routing information. When a join request message arrives at A, A can check which of the edge nodes have the least-cost path that leads A to the target router. Suppose the selected node is E. Then, A can locate a feasible path from itself to E based on its LNC knowledge. The kind of feasible path from a node to its LNC edge node is called a *partial feasible branch (PFB)* during the feasible branch searching process.

Compared to QMRP's method of broadcasting the request message (turning into multiple path searching), QMBF also uses the knowledge of LNC to search for feasible paths. This mechanism makes the request message always travels along a M-hop path toward the target router, which can quickly locate one feasible branch and greatly increase the success ratio.

B. Overview of QMBF

When a new member wishes to join a multicast group, it sends a JOIN message toward the target router (source router or core router). This JOIN message carries the user's QoS requirement, target address, group address and the accumulated QoS information of the path it has traversed. When one node receives the JOIN message, it first checks whether there are edge nodes with least-cost paths toward the target router. Next, it will use the LNC information to find feasible paths (PFBs) from itself to the edge nodes. Then, the node duplicates and forwards the JOIN message toward these edge nodes and reserves QoS resource along the PFBs. If there is no such feasible path or no such edge node, the current node will compute feasible paths from itself to all the edge routers. Then, it duplicates the JOIN message and sends them to all these edge nodes. The process is repeated until a JOIN message arrives at one on-tree router satisfying the QoS requirement.

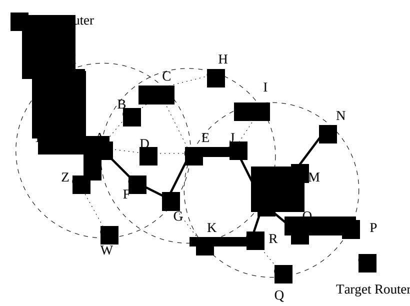

Fig.2. Example of qoS-satisfied branch searching

Figure 2 depicts an example of feasible branch searching process (the thick lines are the track of JOIN messages have passed, M=2): When a new member's join request message arrives at A along the path X-Y-A, A first begins the first step of PFB searching. It computes the feasible paths (PFBs) from itself to edge nodes which have the least-cost toward the target router. Suppose it finds E with least-cost path toward the target router. It then computes a feasible path from A to E using LNC information, and sends the JOIN message and reserves resource along the path:A-F-G-E. When the request message arrives at E, it will repeat the same process and finds another PFB: E-J-L. When L gets the JOIN message, it fails in the first step of PFB search: no edge router has least-cost path toward the target router or no feasible path from the current node to these edge nodes(PFBs). At this time L will begin the second step of PFB search: it computes feasible paths to the edge nodes (except the incoming edge node E). Then, it duplicates and forwards the JOIN messages along the PFBs.

From the above example, we can see that because QMBF utilizes the edge nodes' least-cost information and the M-hop bounded flooding method, most part of the branch is around the the least-cost path. This ensures high success ratio of branch searching.

IV. Detail Description

A. Network Functions

QMBF is based on the current network infrastructure, where the network nodes are connected by duplex, asymmetric links. We assume the network could provide the following functions: 1) each nodes can realize the available local QoS state (such as the available bandwidth, delay or other QoS parameter value); 2) The least-cost routing protocol runs at every nodes; 3) Every on-tree node knows the QoS property from the multicast source to itself (mQoS); 4) The bounded flooding functions run at every network nodes. 5). Every node can use some mechanism to get the target address (multicast source or core router 's address) of one multicast group, such as SDP(Session Directory Protocol)[11].

There are four types of messages involved in QMBF:

1) JOIN (taddr, gaddr, rQoS, aQoS, FPB): The join request message is originated by the a member and sent toward the target router requesting to find a feasible branch which can link the new member to the multicast group tree meeting the QoS requirements. It includes the multicast target address (tadrr), group address (gaddr), required QoS (rQoS), accumulated QoS property (aQoS, the QoS property from the current node to the new member), and available PFB (which is computed based on the bounded flooding messages).

2) CONFIRM (gaddr, rQoS, mQoS): It is originated by one on-tree router. It will travels along the newly-found feasible branch toward the new member. It means that the feasible branch searching process has been finished, confirming this branch and reserved resource. The message includes the group address (gaddr), the accumulated QoS from the multicast source node to the sender of the message (mQoS), etc.

3) UNACK (gaddr, rQoS): it is sent by one router along the reverse direction of the according JOIN message has passed. It means that the branch searching is fail, removing the multicast routing information and releasing the reserved resource. It includes group address, requested QoS (rQoS).

4) PRUNE (gaddr): it is sent by one end user or router, removing unnecessary branches.

The multicast tree is recorded by the multicast entries residing at the nodes of the multicast tree. Each entry of which is denoted as M{G, in, out, mQoS, fix, rQoS, num},where M.G is the address of the multicast group, M.in is the parent of the multicast tree, M.out is the set of child nodes, M.mQoS is the QoS property for this entry from the source to it (when this entry is confirmed) or the required QoS from the source to the current node to meet the new member's QoS requirement (when the entry is unconfirmed), M.fix denotes whether the entry has been confirmed. Otherwise, M.num is the number of JOIN messages have been sent out from the node for the rQoS.

B. Join-Request Process

When the JOIN message arrives at a node, the node first checks whether it already has multicast routing information for this group. If it has the information and the mQoS property of this group plus the aQoS meets the new member's QoS requirement (rQoS), it will add the incoming interface to this group's receivers' list. Then, it sends CONFIRM message toward the new member. Otherwise, it will check whether there is some available PFP information within the JOIN message. If available, it will reserve the QoS resource and add an entry to multicast routing table: gaddr, JOIN message sender id, unfixed, mQoS (the required QoS property from the source the current node meeting the new member's requirement, which equals to rQoS minus aQoS), num=1. It then updates the aQoS of the message and then forwards it along the PFB to the next node. If no such path information within the message, the node will begin the "PFB computing" process using the LNC information.

In the first step, it finds the edge nodes which has the least-cost path (not via the nodes in this LNC) information toward the target router. Then, it computes the feasible paths (PFBs) from itself to these edge nodes. If the next hop node of the feasible paths is not the incoming node (the node which had sent JOIN message to the current node), the current node will duplicate the JOIN message, update the PFB information and aQoS and send the JOIN message along these paths. At the same time, it also will reserve the QoS resource and update multicast routing table: gaddr, message sender list, unpinned mark, QoS, mQoS (the required QoS property from the source the current node meeting the new member's requirement), num.

If the current node can not find a node meeting the above requirement during the first step of "PFB computing process", it will begin the second step, where it computes the feasible paths from itself to all the edge nodes. If the next hop of the feasible paths is not the same as the incoming node, it will then duplicate the JOIN message, update the PFB information and aQoS and send the JOIN messages along these paths. At the same time, it also will reserve the QoS resource and add an entry to multicast routing table: gaddr, JOIN message sender, unfixed, mQoS (the required QoS property from the source the current node meeting the new member's requirement), the number of JOIN messages that have been sent out.

If the router can not find a PFB from its LNC information after the above two steps, it means that the feasible branch searching process is fail, it then sends an UNACK message back to the message sender.

Suppose node i receive a JOIN (taddr, gaddr, rQoS, aQoS, FPB) message from j, its functionalities are formalized in Algorithm 1.

C. Join-Response Process

When one node receives an UNACK message, it first checks how many JOIN messages of this group with the rQoS it had sent out. If there are some JOIN messages that have not been confirmed, it decreases the number of unconfirmed JOIN message. If not, it removes the entry of the multicast group of this rQoS. Then, it duplicates the UNACK messages and sends to the child nodes of the multicast group (listed in the receivers' list of according entry).

When a node receives a CONFIRM message, it first checks whether there is any entry of the same group has been confirmed with better QoS property. It sends back a PRUNE message to the incoming node. Otherwise, it will set the fix field of according routing entry as "TRUE", update the parent node and mQoS of this entry. Then, it will duplicate the CONFIRM

Algorithm 1 **Join-Request Process**

```
if i is an on-tree node of the group G
  if (i is the source of the group or core router)
    send CONFIRM(G,rQoS) to j
    exit
  retrieve the entry M of the group G
    if (M.mQoS + aQoS ≥ rQoS)
      add j into the M.out
      if (M.fix==TRUE)
        send CONFIRM(G,rQoS,mQoS) message to j
    exit
create an entry M
M.out = j
M.mQoS = rQoS - aQoS
M.fix = FAIL
if FPB includes the next hop information
  M.in = next hop
  update the aQOS field of the JOIN message
  send out the JOIN message to the next hop
else
  retrieve the edge nodes whose next hops toward taddr is not within the LNC.
  for each of these edge nodes:
   compute a feasible path from i to the edge node according to rQoS using LNC.
    if (There is such a path and the next hop of the feasible path != j)
      update aQoS, sent the JOIN message along the path.
      increase the M.num
  if (M.num == 0)
    for all edge nodes of the LNC:
      compute out a feasible path from i to the edge node which meet rQoS using LNC.
      if (There is such a path and the next hop of the path != j)
        update the aQoS field of the message and sent out the JOIN along the path.
        increase the M.num
if (M.num == 0)
  send UNACK(G,rQoS) to j
```

message, update the mQoS field and sends an CONFIRM message to the children nodes of the multicast group. It also will check whether there exists the same group's routing information with lower QoS support than the confirmed one. If so, it adds the children nodes of the routing entry to this entry. For the unfixed entries, it also duplicates the CONFIRM message and sends CONFIRM messages to these children nodes. For the confirmed items, it will sent PRUNE messages to their parent nodes.

Suppose node i receives a message from j, the above Join-Response process can be depicted in Algorithm 2.

D. Pruning Process

Whenever an end router becomes the leaf node of multicast tree, it will sends a PRUNE message up the multicast tree and remove itself from the multicast tree. When a node receives a PRUNE message, it will first update the multicast routing information. Then, it will check whether there exists other children nodes of the multicast group. If no, it will also send a PRUNE message to its parent node.

E. Optimization of QMBF

To minimize the traffic overhead of the above QMBF algorithm, we proposed the following methods to optimize it.

1) In the first step of the "PFB computing process," when one JOIN message arrives at one node, the node perhaps can locate multiple edge nodes which has least-cost path toward the target router or multiple PFBs to these edge nodes. If this node sent

Algorithm 2 **Join-Response process**

```
If the message is CONFIRM(gaddr, rQoS, aQoS):
  retrieve the multicast entry M1,M2,M3...Mn of gaddr (assuming that M1.rQoS = rQoS)
  if (exits Mk where Mk.fix = TRUE and Mk.rQoS ¿ rQoS)
    send PRUNE(gaddr) to j
    set M1.in = j
    add M1.out to Mk.out
    delete M1
    exits
  set M1.fix = TRUE
  for those Mk(k-2..n)
    if (Mk.rQoS < M1.rQoS)
      if (Mk.fix != TRUE)
        sent CONFIRM(gaddr, Mk.rQoS, aQoS) to Mk.out
        sent PRUNE(gaddr) to M.in
      add Mk.out to M1.out
      delete Mk
If the message is UNACK(gaddr,rQoS):
  retrieve the multicast entry M with gaddr and rQoS
  update M.num
  if (M.num=0)
    sent UNACK(gaddr,rQoS) to M.out
    delete M
```

out multiple copies of JOIN messages, there will be more and more branches along the JOIN paths. To control the number of branches, every time we only select the edge node that is the nearest one to the least-cost path leading the current node to target node.

2) In the second step of "PFB computing process", QMBF finds PFBs for every edge nodes, duplicates the JOIN message and send out along these partial branches. If this situation happens too frequently, there will be more and more branch nodes causing excessive overhead. We use the concept "Maximum Branch Degree" from [2] to control the traffic, which will control the number of nodes that possible become branch nodes.

3) Because QMBF is based on bounded flooding technique, if the flooding messages travels too many hops, it will also become burden for network. So, we use "bounded hops" to control the flooding traffic, which decides for how many hops the flooding message is valid.

Based on above methods, we optimize QMBF into QMBF-mn (m means the flooding hops, n means maximum branch degree). We can using different value of m and n to fit our requirement under different network situation.

V. Simulation Experiment & Results

In this section, we study and compare the performance of QMBF with other QoS-based multicast protocols. Four other algorithms were simulated: single-path joining protocol, directed spanning-joins protocol, QMRP2, and QMRP3. For QMBF, we simulated six schemes: QMBF-12, QMBF-13, QMBF-22, QMBF-23, QMBF-32, and QMBF-33.

A. Simulator Environment

The simulations were conducted on the Waxman network topologies[8]. We use the following approach to randomly generate network topology: network nodes are randomly chosen in a square ($\alpha * \alpha$) grid. A link exists between the nodes u and v with the probability p(u,v) =a*exp(-d(u,v))/b* $\sqrt{2} * \alpha$), where d(u,v) is geometric distance between u and v, a and b are constants that are less than 1. In the simulation, a and b are randomly chosen so that the average degree of nodes is between

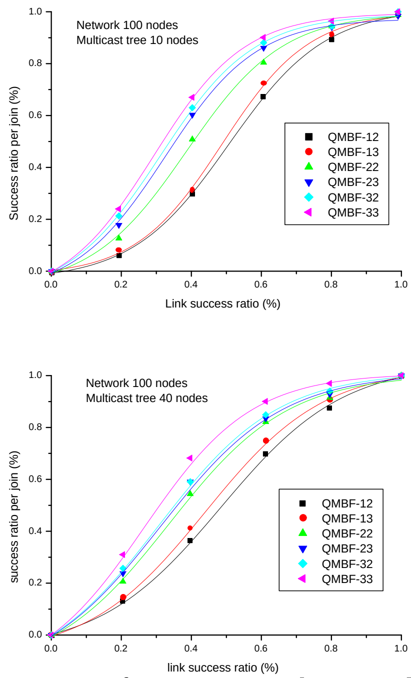

Fig.3. Success ratio of QMBF-mn with two multicast group size

4 and 5. The networks have a fixed size of 100 nodes over a 100*100 grid.

Based on above topology generation method, for each simulation, a network topology, a multicast tree, and a new member out of the tree are randomly generated. We use two types of tree size (including the internal nodes): 10 and 40 nodes. A new member has a randomly generated QoS requirement. The QoS property of each link is also randomly generated. For each situation (different protocols, multicast group sizes, link success ratios (what percent of the links meet the new user's QoS requirement.)), we run the simulation 200 times. We have mainly focused on the success ratio as the measure of the performance.

B. Results and Analysis

Figure 3 depicts the success ratio of different modes of QMBF with the two different size of multicast group. We can see that QMBF-mn's success ratio increases with increase in m and n. As m increases, the nodes can have much chance to find PFBs when some part of the least-cost paths don't meet the new member's requirement. n decides the chance the routing activity enters multiple path searching process, which also increases the chance of finding a feasible branch.

Figure 4 compares the success ratios of all the simulated protocols with the two different size of the multicast group. The figure shows that the success ratios of QMBF-22 and QMBF-23 are better than QMRP2, QMRP3, spanning-joins protocol, and single-path join protocol.

VI. Summary

In this paper, we propose a new QoS-aware multicast routing protocol called QMBF. Basing on bounded flooding technique, every network node can have a LNC topology information as well as QoS state information. The QMBF mainly utilizes this technique to increase the chance of finding the feasible branch.

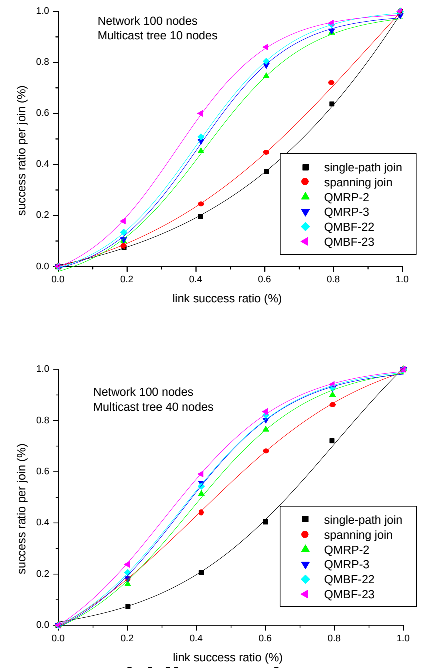

Fig.4. Success ratio of different multicast protocols with two group size

It uses two methods to find feasible branch: Computing PFB using local network cell information (collected from bounded flooding) and using multiple path searching. QMBF can either operate on top of any unicast routing protocol or cooperate with QoS-based unicast routing protocol. The protocol requires no intermediate routers to keep the temporal searching states and need not flood the whole network to find the multicast path. The simulation results shows that QMBF can achieve better success ratio than other QoS-based multicast routing protocols.

References

[1] K. Carlberg and J. Crowcroft, "Building Shared Trees Using a One-to-Many Joining Mechanism," Computer Communication Review, Jan. 1997.

[2] S. Chen, K. Nahrstedt. and Y. Shavitt, "A QoS-Aware Multicast Routing Protocol," IEEE INFOCOM, May 2000.

[3] M. Faloutsos, A. Banerrjea, and R.Pankaj, "QoSMIC: Quality of Service Sensitive Multicast Internet Protocol," SIGCOMM, Sep. 1998.

[4] A. Fei, M. Gerla, "Receiver-initiated Multicasting with Multiple QoS Constraints,", IEEE INFOCOM, May 2000.

[5] Y. Jia, I. Nikoladis, P. Gburzynski, "Multiple Path Routing," Proceedings of ICC'01, June 2001.

[6] S. Deering, "Multicast Routing in Internetworks and Extended LANs," SIGCOMM'88, Aug. 1988.

[7] S. Chen and K. Nahrstedt, "An Overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solutions", IEEE Network, Special Issue on Transmission and Distribution of Digital Video, Nov./Dec. 1998.

[8] B. M. Waxman, "Routing of Multipoint Connections," IEEE Jornal on Selected Areas in Communications, Vol. 6, No. 9, Dec. 1988.

[9] C.Diot, W.Dabbous, and J. CrowCroft, "Multipoint Communication: a survey of protocols, functions, and mechanisms," IEEE Journal on Selected Areas in Communications, Vol.15(3), April 1997.

[10] B. Wang, C. Hou, "A survey on mulitcast routing and its QoS extension: problems, algorithms, and protocols," IEEE Network Magazine, Vol.14, No.1, Jan./Feb. 2000.

[11] Handley and V.Jacobson, "SDP:Session Directory Protocol (draft2.1)," Internet Draft, Feb. 1996.

[12] X. Xiao and L. M. Ni, "Internet QoS: the Big Picture," IEEE Network, Mar. 1999.