Minimizing Cost and Delay in Shared Multicast Trees

Seok-Joo Koh[a], Jong-Hwa Yi, Jin-Ho Hahm, Byoung-Moon Chin, and Chee-Hang Park

Existing tree construction mechanisms are classified into source-based trees and center-based trees. The source-based trees produce a source-rooted tree with a low delay. However, for the applications with multiple senders, the management overheads for routing tables and resource reservations are too high. The center-based trees are easy to implement and manage, but a priori configuration of candidate center nodes is required, and the optimization nature such as tree cost and delay is not considered. In this paper, we propose a new multicast tree building algorithm. The proposed algorithm basically builds a non-center based shared tree. In particular, any center node is not pre-configured. In the proposed algorithm, a multicast node among current tree nodes is suitably assigned to each incoming user. Such a node is selected in a fashion that tree cost and the maximum end-to-end delay on the tree are jointly minimized. The existing and proposed algorithms are compared by experiments. In the simulation results, it is shown that the proposed algorithm approximately provides the cost saving of 30 % and the delay saving of 10 %, compared to the existing approaches. In conclusion, we see that the cost and delay aspects for multicast trees can be improved at the cost of additional computations.

Manuscript received December 31, 1998; revised January 24, 2000.
[a] Electronic mail: sjkoh@pec.etri.re.kr

I. INTRODUCTION

The multicasting can be defined as the distribution of the same information stream from one to many nodes concurrently. In the last few years, multicast routing has attracted a lot of attention from the network community, since many emerging applications are of multicast nature such as teleconferencing, remote education and collaborative applications [17].

Recent studies related to Internet protocol (IP) multicast routing can be categorized into the following two approaches; source based tree (SBT) and shared tree. SBT is composed of the shortest paths from a source to all receivers. Thus a different tree is built and managed for each source. In contrast, the shared tree is a single delivery tree that is shared by all users of a group. In this tree, a single router acts as 'center' of a multicast tree for a group.

Clearly, shared trees have some advantages over the SBT in terms of scalability. The scalability concerns represent that the required network state information such as routing table entries is exponentially increased, as the size of network or multicast group gets larger. A good scaled protocol is referred to the protocol that can be easily applied even to a large-scale network. In SBT, a distinct tree is built for each active source. Thus SBT scales $O(S*G)$, where S is the number of active sources and G is the number of group members. This represents that each router in the network must maintain the routing table entries per source and per group. On the other hand, the shared tree eliminates the source-scaling factor S since all sources share the same tree. Accordingly, CBT scales $O(G)$.

However, most existing studies related to the shared trees have not considered the cost and delay characteristics of the multicast tree. The tree cost minimization is directly related to the maximization of network resource utilization, which is one

Table 1. Acronyms for multicast routing protocols.

Acronym	Multicast Routing Protocols or Mechanisms
SBT	Source Based Tree
CBT	Center Based Tree
DVMRP	Distance Vector Multicast Routing Protocol
MOSPF	Multicast extensions to Open Shortest Path First
PIM-DM	Protocol Independent Multicast – Dense Mode
PIM-SM	Protocol Independent Multicast – Sparse Mode
QoSMIC	Quality of Service Multicast Internet protoCol
SMP	Simple Multicast protocol
BGMP	Border Gateway Multicast protocol

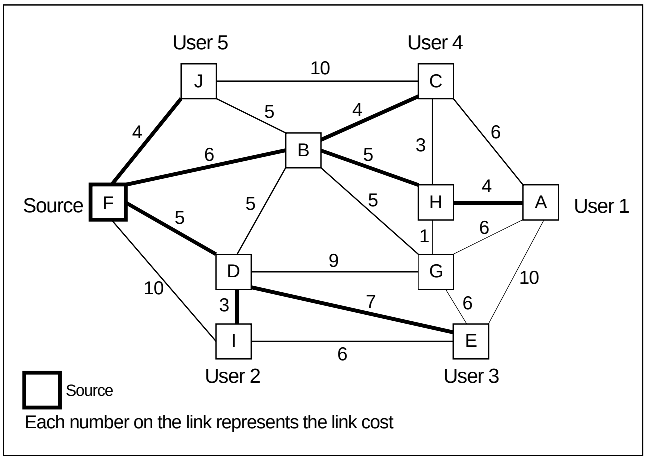


Fig. 1. Source based tree.

of the main interests to current Internet service providers. The delay aspect is also very important to support high quality of multimedia applications services. Thus the existing multicast trees are not optimal in terms of network resources utilization and quality of services. In addition, the shared trees in PIM-SM, CBT, SMP and BGMP protocols require a priori configuration of candidate center nodes. Thus a set of routers should be pre-configured as candidate centers. In fact, the selection of suitable center nodes is very difficult since information of user locations is not known a priori.

In this paper, we propose a new algorithm to construct a shared multicast tree. Differently from the existing approaches, our algorithm has the following primary goals.

• *Limited impact of candidate center selection problem.* We want to limit the impact of the center node selection. To do this, no center node is pre-configured for multicast routing. Instead, a new multicast node is dynamically assigned to each incoming user.

• *Cost-delay minimization for shared multicast tree.* To do this, a multicast node among current tree nodes is selected for a new incoming user such that the tree cost and the maximum end-to-end delay are jointly minimized.

This paper is organized as follows. Section II reviews the existing multicast routing algorithms. In Section III, we propose a new algorithm to build a shared multicast tree and to describe the relationship between the proposed and existing algorithms. In Section IV, The performance of each candidate algorithm is compared by simulations. Section V concludes this paper.

II. PREVIOUS WORKS

Many multicast routing protocols have been proposed in the last few years. Table 1 provides the acronyms for those existing protocols. Roughly, these multicast routing algorithms can be classified into source-based trees and shared trees.

A source-based tree (SBT) is a source-rooted shortest path tree, which consists of the shortest paths from a source to all receivers. The SBT is employed in DVMRP [16], PIM-DM [6], and MOSPF [13]. These protocols are slightly different each other in the viewpoint of underlying unicast routing protocols. DVMRP constructs a reverse path tree by using the distance vector protocol, while MOSPF employs the OSPF protocol. In contrast, PIM-DM is independent of any specific unicast routing protocol. All these protocols are based on SBT.

Figure 1 illustrates an example of SBT with 10 nodes. In the figure each number on the link represents the link cost. The link cost usually depends on available link bandwidth and transmission delay between two end nodes. In the example network, SBT consists of the shortest paths from the source node F to all users. Thus the tree cost is 38 and the maximum end-to-end delay is 15. In the figure, tree cost represents the sum of link costs of links in the tree, and the maximum end-to-end delay is defined as the maximum path length among all sender-to-receivers paths in the tree. Along the resulting tree, multicast data will flow from source to each user.

A shared tree is a single delivery tree that is shared by all users of a group. All active senders in a group share a common tree for multicasting. Typically a shared tree is constructed by using a center node. Such a tree is called a center-based tree. This approach is employed in CBT [2], [3], PIM-SM [7], SMP [14] and BGMP [11], [15]. On the other hand, QoSMIC protocol [4], [8] uses a non-center based shared tree. Each incoming user is connected to one of the current in-tree routers for multicasting. A more detailed explanation for each approach is as follows.

The CBT was developed to address the scalability issues of SBT. In CBT, a multicast tree is built by dynamically choosing a suitable center node. All members join a group by sending pecial join messages toward the center. The routers along the

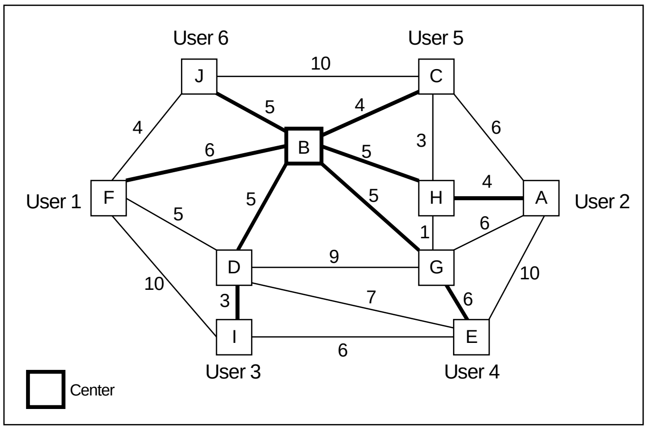


Fig. 2. Center based shared tree.

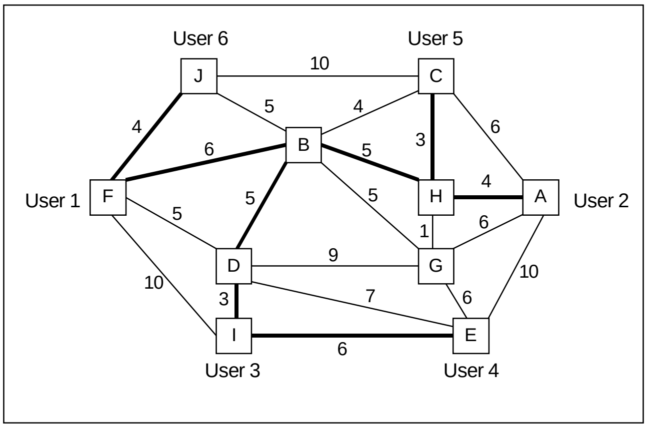


Fig. 3. Greedy search based shared tree.

path keep state about which ports are in the tree. The result is a tree of the shortest paths from the center to all members.

PIM-SM is basically similar to CBT. In PIM-SM, however, there are two possible trees for traffic from a particular source to the group members; shared tree and SBT. Note that CBT uses only the shared tree. PIM-SM uses a shared tree by default. However, it can selectively switch to SBT on a per-source basis if the routing policy permits. The forwarding state created by PIM-SM is unidirectional in that it only allows traffic to flow away from the center, not toward it. The CBT however builds a bi-directional tree.

SMP was proposed to reduce the complexity of center node selection. In conventional IP multicast, only a multicast address is assigned to a group. Thus the selection of center nodes is additionally required in CBT. The basic idea in SMP is that a multicast group is identified through a pair of a distinguished center node and a multicast address. Given a pair of center node and multicast address, SMP constructs a center-based tree.

BGMP was developed for inter-domain multicast routing. BGMP can also build two kinds of trees, similar to PIM-SM. Trees in BGMP are those of domains, not routers. BGMP presumes that groups of class D addresses are assigned to each domain. The trees of BGMP are also bi-directional as those of CBT.

Figure 2 illustrates an example center-based tree. In the figure, it is assumed that node B is selected as the center, and such information is broadcast to all of routers in the network. User 1 first arrives at the network through the access node F and finds the shortest path between two nodes, F and B. The other users also send a join message toward the center node, and the shortest paths are calculated from the access nodes and the center. The result is the center-rooted shortest path tree, as shown in Fig. 2. In the figure, the tree cost is 43 and the maximum end-to-end delay is 20. Based on the tree, each user can send multicast data to the other users.

The QoSMIC protocol is another shared tree approach. In QoSMIC, any center is not used. Thus it is free from the controversial issue of center node selection. QoSMIC constructs a tree based on the greedy search [4]. Each incoming user is connected to the closest branch of the existing tree. Thus QoSMIC leads to a lower tree cost than center-based tree approach. QoSMIC can also build SBT when needed. In fact, the main contribution of QoSMIC is to provide QoS guaranteed routes. To do this, QoSMIC first identifies multiple routes from the incoming user to the existing tree. These routes typically consist of the links that have available resources enough to provide the required bandwidth. Then QoSMIC selects the shortest path among those multiple routes.

Figure 3 illustrates a greedy search based tree in QoSMIC. It is assumed that each user arrives at the network in the order of User 1, 2, 3, 4, 5 and 6. Let us consider a group that consists of users 1, 2 and 3. In this case, QoSMIC protocol produces a tree consisting of tree nodes; F, B, H, A, D and I. When user 4 joins the group by way of node E, the user is connected to the in-tree node I by the greedy search. After users 5 and 6 join the group, the result tree is shown in Fig. 3. The tree cost is 36 and the maximum end-to-end delay is 24, which corresponds to the length of path between users 4 and 6. Based on the tree, each user can send multicast data to the other users.

In summary, SBT has better end-to-end performance (e.g. lower delay) than shared tree. However, SBT is not scalable since a routing entry is required per source per group. Thus the overhead is too high for the group applications with multiple senders. Center-based shared trees are beneficial in terms of scalability. However, the optimization of tree cost or delay is not considered, and the decision such as center node selection is required. In QoSMIC, such a decision is not required, and the tree cost is minimized. However, the end-to-end delay is high in the result tree, and thus such a tree is not suitable to delay-sensitive applications. The SBT and shared trees have complementary behavior, and both are useful in different situations.

III. SHARED MULTICAST TREE WITH COST-DELAY MINIMIZATION

In this paper, we propose an algorithm to construct a shared tree with cost-delay minimization. We denoted the proposed algorithm by the Shared Tree with Cost-Delay (STCD) minimization. The STCD tree is similar to the QoSMIC tree in that no center node is used. STCD however considers a joint minimization of tree cost and maximum end-to-end delay, while the QoSMIC considers only the tree cost.

1. STCD Algorithm

Let us denote T by be a current tree of the concerning group. When a new user wants to join the group, STCD algorithm selects a distinct node among tree nodes in T for multicast routing. Such a node is called a multicast node for the new arrival user. A multicast node is selected in a fashion that the sum of two cost components is minimized. Two cost components are the length of path from the arrival user to the selected multicast node and the maximum end-to-end delay on the result tree.

If the access node of the incoming user is already in T, it is not necessary to select any multicast node. Otherwise, a multicast node should be selected for the user. The following model provides a rule to select a multicast node.

Multicast Node Selection (MNS) Model

Minimize $$\sum_{i \in T} \{ k\, f_i(x_i) + (1-k) g_i(x_i) \} \quad (1)$$

Subject to $$\sum_{i \in T} x_i = 1 \quad (2)$$

$$x_i = \{0, 1\} \quad \forall i \in T \quad (3)$$

In the objective function, $f_i(x_i)$ represents the length of the shortest path between the user node and the selected multicast node i, and $g_i(x_i)$ represents the maximum end-to-end delay on the result tree, after the shortest path is added to the current tree T. In the model, k represents a relative weight for tree cost and maximum delay, which is ranged from 0 and 1. For example, we can employ $k = 0$ for delay-sensitive applications, and $k = 1$ for the applications sensitive to tree-resources. The constraints (2) and (3) simply designate that just one node must be selected among in-tree nodes in T. Then in the MNS model the multicast node is finally selected that minimizes the objective function value (1).

Figure 4 illustrates an example of the proposed STCD tree. In the network, user 1 first arrives at the network through node F. When the user 2 arrives by way of node A, the shortest path between node F and A is obtained. Then the current multicast

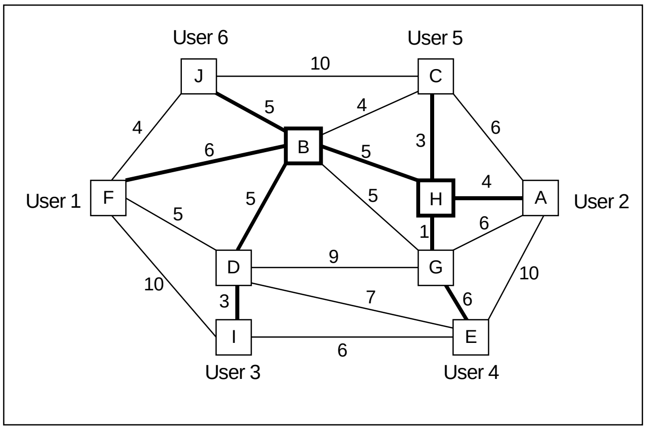


Fig. 4. Cost-delay based shared tree.

tree consists of the nodes F, B, H and A as shown in the figure. We now consider user 3 who joins through the node I. Based on the MNS model, the node B is selected as the multicast node for the user. In this case, the path length is 5 + 3 = 8. The maximum delay is 17, which represents the length of the path between nodes I and A. Now let us consider user 4 who joins by way of the node E. As a similar way, the node H is selected as the multicast node. In this case, the path length is 6 + 1 = 7 and the delay is 20. In the figure, we know that total tree cost is 38, and the maximum delay is 20.

We can easily show that the worst-case computational complexity of the proposed algorithm is $O(n^3)$, where n represents the number of nodes in T, since the calculation of the shortest path is required for each candidate multicast node [10]. Such an additional complexity may give some overhead to current multicast routers. However, this overhead will become negligible, as the computing powers are drastically improved in near future. In addition, we believe that QoS aspects for multicasting are much improved at the cost of additional computations.

2. Comparison with Existing Approaches

SBT provides the lowest delay among existing approaches. However, it is not scalable for applications with multiple senders, since a different tree is built for each distinct source. Thus tree costs tend to become high. Center based trees were proposed to solve the scalability issues of SBT. However, the optimal center placements must be made for efficient multicast routing. This is a difficult problem because user locations are not known a priori. QoSMIC tried to address such a center selection problem by using a non-center based shared tree.

The proposed STCD is similar to the QoSMIC in that a non-center based shared tree is used. However, STCD jointly minimizes the tree cost and maximum delay, while QoSMIC considers only tree cost. In fact, we can see that QoSMIC algorithm corresponds to a special case of $k = 1$ in the MNS model.

The main contribution of QoSMIC is to provide QoS guaranteed routes, not to simply optimize tree costs, as described in Section II.

In terms of the worst-case computational time-complexity, the SBT requires $O(n^3)$, where n represents the number of nodes, because the shortest paths must be calculated for each source node in the network. The CBT requires $O(n^2)$ because the shortest path between each access node and the core node is obtained. However, the CBT requires the core node selection problem to be solved additionally. On the other hand, in the QoSMIC protocol, the complexity of $O(n^4)$ is required. This is because the shortest paths are computed for each target in-tree node $O(n^3)$ and additional procedure $O(n)$ must be performed to check whether each link on the path provides enough link capacity for the QoS requirement.

Table 2. Test multicast algorithms.

Tree Type	Related Protocols	Symbol	Descriptions
Source Based Tree	DVMRP, MOSPF, PIM-DM	SBT	Each source node has a different multicast tree
Center Based Shared Tree	CBT, PIM-SM, SMP, BGMP	CBT	The shortest paths from a center to all receivers are found.
Non-Center Based Shared Tree	QoSMIC, STCD (Proposed)	STCD ($k=1$)	Tree cost (the same as QoSMIC)
		STCD ($k=0.5$)	Both tree cost and maximum delay
		STCD ($k=0$)	Maximum delay

IV. EXPERIMENTAL RESULTS

In this section, we compare the performance of existing and proposed algorithms. To do this, we classify multicast tree algorithms into three types; source-based tree, center-based shared tree and non-center based shared tree. Table 2 summarizes those test algorithms. In the table, SBT assumes that all of group users are active sender as well as receiver. In the non-center based tree, STCD ($k = 1$) represents the QoSMIC algorithm. Both STCD ($k = 0.5$) and STCD ($k = 0$) include the delay characteristics in the tree-building process.

To evaluate the performance of those algorithms, we employ two kinds of test networks. The first is a real network topology [5], which is the map of the major MBONE routers. We eliminate routers with only one incident link, since such routers do not affect routing. The final graph has 32 routers, 80 links, and average degree of 2.5. The second is randomly generated networks with 100 nodes. To generate such networks, we employ the Georgia Technology Internetwork Topology Models (GT-IMT) software [18]. For each test network, we assume that group users randomly arrive at the network. The link cost is randomly assigned as an integer number ranged from 1 to 10. In particular, in the implementations of CBT algorithm, a center node is randomly selected in the network. To compare the performance of test algorithms, we measured the following two metrics; tree costs and maximum end-to-end delay. We run every experiment 100 times, and the results are averaged. In our experiments, it is not easy to reasonably define the worst-case results, because two different measures are evaluated together for the same networks. Thus we present only average results.

Figures 5 and 6 show the performance of shared trees in MBONE networks. In Fig. 5, STCD (1) and STCD (0.5) give better performance than CBT and STCD (0) in terms of tree cost. Both STCD (1) and STCD (0.5) provide nearly the same performance. STCD (0) shows the worst performance. This is because any tree cost characteristic is not considered in the algorithm. In the figure, we also know that the gap between CBT and STCD (0.5) or STCD (1) is larger, as the number of users increases. Figure 6 shows the maximum delay of shared trees in MBONE networks. It is clear that STCD (0) and STCD (1) give the best and the worst performance, respectively. Both CBT and STCD (0.5) provide nearly the same performance. Note that the gap of maximum delay between STCD (0.5) and STCD (0) is smaller than that of between STCD (0.5) and STCD (1).

Table 3 summarizes simulation results for SBT and shared tree algorithms. In the table, the CBT performance is represented as 100. From the table, it is clear that SBT algorithms provide the lowest delay. However, SBT is not cost-effective in terms of tree costs. Compared to CBT, both STCD (0.5) and STCD (1) give cost-savings of approximately 30 %. STCD (0) shows the worst performance in terms of tree costs. On the other hand, STCD (0.5) provides better performance than STCD (1) in terms of maximum delay. The STCD (0) is slightly better than STCD (0.5), but the amount of delay-saving is too little, compared to its large tree cost. Thus we can conclude that STCD (0.5) produces more reasonable trees than STCD (0) and STCD (1).

Table 4 shows the relative performance of test algorithms in randomly generated networks with 100 nodes. The results of test algorithms are nearly the same as those in Table 3. However, we note that the delay performance of STCD (0.5) is better than that of CBT. This implies that the performance of STCD (0.5) gets better, as the network size becomes larger. From the table, we also know that the proposed STCD (0.5) provides the cost-saving of 30 % and the delay-saving of 10 % approximately, compared to CBT.

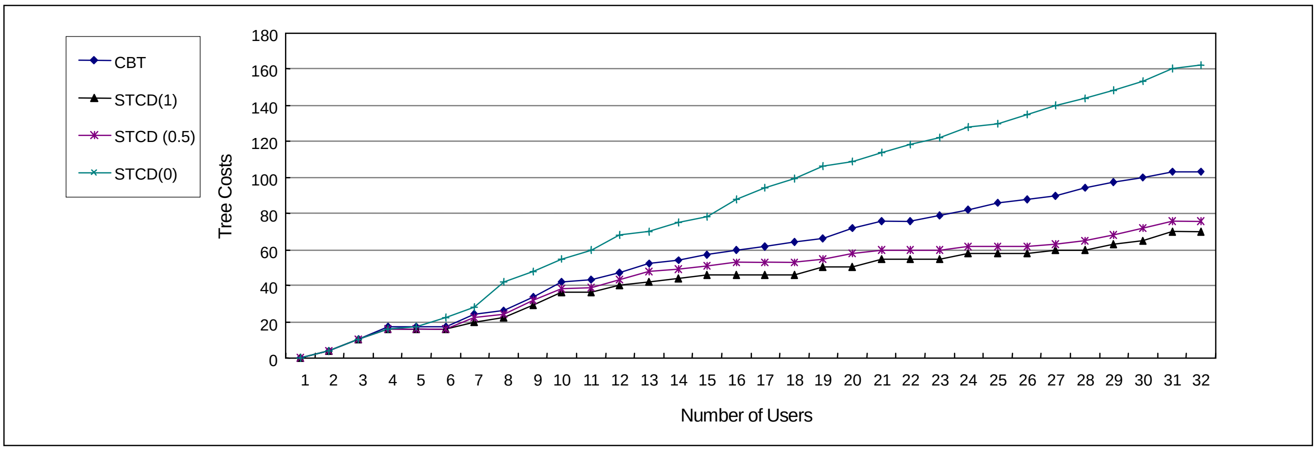


Fig. 5. Tree costs of shared trees in MBONE networks.

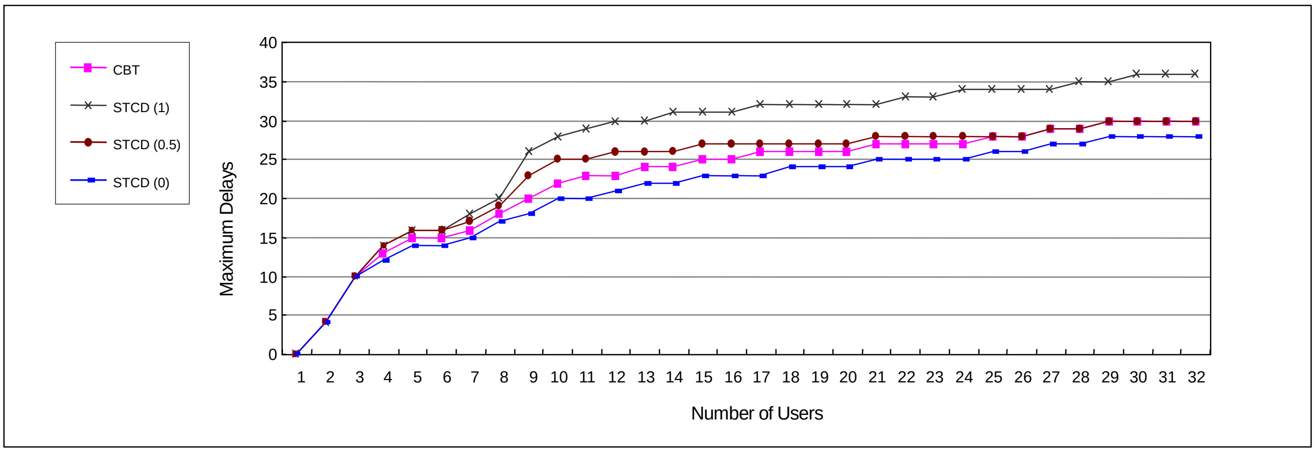


Fig. 6. Maximum delays of shared trees in MBONE networks.

Table 3. Relative performance of test algorithms in MBONE networks.

	SBT	CBT	STCD		
			STCD (1)	STCD (0.5)	STCD (0)
Tree Cost	286	100	66	73	160
Maximum Delay	69	100	122	100	95

Each number in the table represents a relative value to the CBT performance of 100.

Table 4. Relative performance in random networks with 100 nodes.

	SBT	CBT	STCD		
			STCD (1)	STCD (0.5)	STCD (0)
Tree Cost	224	100	70	74	156
Maximum Delay	70	100	132	90	87

Each number in the table represents a relative value to the CBT performance of 100.

V. CONCLUSION

Existing tree building mechanisms are classified into source based tree and center based tree. The source-based tree is a source-rooted tree with a lower delay. However, the management of routing tables and network resources becomes more complex for the applications with multiple senders. Center based trees are easy to implement and manage, but a priori configuration of candidate center nodes is required. In addition, the optimization nature such as tree cost or delay is not considered.

In this paper, we proposed a new multicast tree building algorithm. The proposed algorithm basically builds a non-center based shared tree for multicast. In the proposed algorithm, a multicast node among the existing tree nodes is assigned to each incoming user. Such a node is selected in a fashion that tree cost and the maximum end-to-end delay are jointly minimized. In the simulation results, it is shown that the proposed algorithm approximately provides cost-savings of 30 % and delay-savings of 10 %, compared to the center based tree approaches.

It seems that the proposed algorithm is one of suitable approaches for multicasting. In particular, it is expected that the proposed approach can be deployed to real-world IP based networks, together with consideration of the traffic engineering issues, which are being examined in Multi-Protocol Label Switching technology [1].

REFERENCES

[1] D. Awduche *et al*., *Requirements for Traffic Engineering over MPLS*, *RFC2702*, September 1999.

[2] A. J. Ballardie, *Core Based Trees Multicast Routing Architecture*, *RFC2201*, September 1997.

[3] A. J. Ballardie, B. Cain, and Z. Zhang, "Core Based Trees (CBT version 3) Multicast Routing: Protocol Specification," *Internet-Draft: draft-ietf-idmr-cbt-spec-v3-01.txt*, August 1998.

[4] K. Carlberg and J. Crowcroft, "Building Shared Trees Using a One-to-Many Joining Mechanism," *ACM Computer Communication Review*, January 1997, pp. 5–11,

[5] S. Casner, "Major MBONE routers and links," Available from ftp://ftp.isi.edu:mbone/mbone.topology.ps, 1994.

[6] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, A. Helmy, D. Meyer, and L. Wei, "Protocol Independent Multicast Version 2 Dense Mode Specification," *Internet-Draft: draft-ietf-pim-v2-dm-01.txt*, November 1998.

[7] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, *Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*, *RFC2362*, June 1998.

[8] M. Faloutsis, A. Banerjea, and R. Pankaj, "QoSMIC: Quality of Service sensitive Multicast Internet protocol," *ACM SIGCOMM '98*, September 1998.

[9] S. Koh *et al*., "Non-core Based Shared Tree for IP Multicasting," *IEE Electronics Letters*, Vol. 35, No. 11, May, 1999, pp. 872-873.

[10] D. Kosiur, *IP Multicasting*, Carol Long, Ed., John Wiley & Sons, Inc. (publisher) , 1998.

[11] S. Kumar, P. Radoslavov, D. Thaler, C. Alaettinoglu, D. Estrin and Mark Handly, "The MASC/BGMP Architecture for Inter-domain Multicast Routing," *ACM SIGCOMM '98*, September 1998.

[12] T. Maufer, *Deploying IP Multicast in the Enterprise*, Prentice-Hall Inc., ISBN 0-13-897687-2, 1998.

[13] J. Moy, *Multicast Extensions to OSPF*, *RFC1584*, March, 1994.

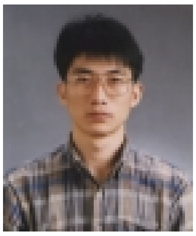
[14] R Perlman, C. Lee, A. Ballardie, J. Crowcroft, Z. Wang, and T. Maufer, "Simple Multicast: A Design for Simple, Low-Overhead Multicast," *Internet-Draft: draft-perlman-simple-multicast-01.txt*, November 1998.

[15] D. Thaler, D. Estrin, and D. Meyer, "Border Gateway Multicast Protocol (BGMP): Protocol Specification," *Internet-Draft: draft-ietf-bgmp-spec-00.txt*, Januaty 2000.

[16] D. Waitzman, S. Deering and C. Partridge, *Distance Vector Multicast Routing Protocol*, *RFC1075*, November 1988.

[17] J. Yi and E. Pastor, "Specification of Communication Services for Multimedia Cooperative Applications Based on the Reference Model of ODP," *ETRI Journal*, Vol. 20 No. 2, June 1998.

[18] E. W. Zegura, "GT-ITM: Georgia Tech Internetwork Topology Models," Available from http://www.cc.gatech.edu/fac/Ellen.Zegura/graphs.html, 1999.



Seok-Joo Koh received B.S. and M.S. degrees in management science from KAIST in 1992 and 1994 respectively. He also received Ph.D. degree in industrial engineering from KAIST in 1998. Since 1998, he has been a senior researcher at Protocol Engineering Center in ETRI. His research interests include Internet multicast routing, reliable multicast transport, telecommunication network design and planning.

Jong-Hwa Yi received the M.S. degree in electronics engineering from Hanyang University, Seoul, Korea in 1990 and the Ph.D degree in telematics engineering from Technical University of Madrid, Spain in 1996. She joined ETRI in 1990 and she is a senior member of research staff in Protocol Engineering Center (PEC). Current interest research fields include distributed application architectures, open distributed processing, cyber education and object-oriented modeling.

Jin-Ho Hahm received his B.A. degree in electrical engineering in 1982 from Hanyang University. He also received M.S. and Ph.D. degrees in computer engineering in 1984 and 1998 from Hanyang University. He joined ETRI in 1984 and has been working for Protocol Engineering Center (PEC) since 1991. He has also been the team manager of Advanced Standardization Team at PEC. He is currently involved in Joint Project with NIST.

Byoung-Moon Chin was born in Seoul, South Korea in 1953. He received his B.S. degree in electrical engineering in 1976 and M.S. degree in computer engineering in 1983, both from Seoul National University. He received his Ph.D. Degree in computer science in 1996 from Korea Advanced Institute of Science and Technology (KAIST). He joined ETRI in 1980, where he is currently working as Director at the Protocol Engineering Center, and a Special Rapporteur of ITU-T SG7 Q.23 on data communication protocol testing. His research interests include protocol testing, test methodology, protocol engineering and computer networks.

Chee-Hang Park received the B.S. degree in applied physics from Seoul National University, Korea in 1974, the M.S. degree from Korea Advanced Institute of Science and Technology, Korea in 1980, and the Ph.D. degree in computer science from University of Paris 6, France in 1987. During the last 10 years he has been involved as project leader in several large projects such as Multimedia Computer System Development and High Speed Parallel Computer System Development. His research interests include multimedia systems, distributed system, middleware, groupware, network virtual computing, and mobile agent architecture. He is currently Executive Director of Information Support Division, ETRI.