

Copyright © 2005 IEEE. Reprinted from  
International Conference on Advanced Information Networking and  
Applications (19th : 2005 : Taipei, Taiwan)

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Adelaide's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to [pubs-permissions@ieee.org](mailto:pubs-permissions@ieee.org).

By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

# Hamming Distance and Hop Count Based Classification for Multicast Network Topology Inference

Hui Tian and Hong Shen  
Graduate School of Information Science  
Japan Advanced Institute of Science and Technology  
1-1 Asahidai, Tatsunokuchi, Ishikawa, 923-1292, Japan  
hui-t,shen@jaist.ac.jp

## Abstract

*Topology information of a multicast network benefits significantly to many applications such as resource management, loss and congestion recovery. In this paper we propose a new algorithm, namely binary hamming distance and hop count based classification algorithm (BHC), to infer multicast network topology from end-to-end measurements. The BHC algorithm identifies multicast network topology using hamming distance of the sequences on receipt/loss of probe packets maintained at each pair of nodes and incorporating the hop count available at each node. We analyze the inference accuracy of the algorithm and prove that the algorithm can obtain accurate inference at higher probability than previous algorithms for a finite number of probe packets. We implement the algorithm in a simulated network and validate the algorithm's performance in accuracy and efficiency.*

*Key words:* Multicast network, topology inference, sequence, hamming distance.

## 1 Introduction

For the efficient use of network resources, multicast has become one of the most popular forms of communication. Knowledge of multicast network topology can significantly facilitate resource management, and can be applied to build schemes for loss recovery and congestion control in the context of multicast sessions supporting heterogeneous receivers [1, 4]. End-to-end measurements-based multicast network inference becomes one of the most efficient and effective approaches to obtain the information of the network internal characteristics such as topology and loss performance.

Much research on multicast topology inference from

end-to-end measurements can be found in [2, 3, 6, 7, 5, 8]. The key idea underlying the approach is that receivers sharing common paths on the multicast tree associated with a given source will see correlations in their packet losses or delays. The multicast tree can thus be inferred based on the shared loss or delay statistics on transmitted probe packets. The main advantage of the approach lies in its applicability to inferring multicast trees requiring no support from internal nodes. It permits however, identification of a logical multicast topology rather than the actual physical topology in previous work. A long path with no branches would be identified as a single logical link. Thus, all single-child nodes are deleted in the inferred logical topology. It is obvious that in practice this may not be an appropriate inference of the actual topology because there may exist many nodes with only one child. Moreover, as we have presented in [11], the prevalent method to estimate correlation used in [2, 5, 6, 7, 9, 12] for siblings identification may produce fault results. Therefore, we propose the usage of hamming distance. Incorporating the hop count into the hamming distance based classification proposed in [11], a new algorithm, which is called the binary hamming distance and hop count based classification algorithm(BHC) is proposed in this paper. It can not only solve fault correlation existing in current algorithms, but also identify the nodes and links failed to be inferred in [7]. In this paper, the BHC algorithm will be described in detail. We will also prove the superiority of the BHC algorithm by both theoretical proof and simulation. Moreover, we extend the same idea on combining hamming distance with hop count to general tree topology inference so that our topology inference algorithm is generalized.

The paper is organized as follows. In Section 2 a mathematical model of multicast network is introduced. The BHC algorithm for multicast topology inference is presented in Section 3. Section 4 gives a theoretical analysis on inference accuracy of the algorithm. Section 5 presents the sim-

ulation results of the algorithm. Extension of the algorithm to the general trees is discussed in Section 6. Section 7 concludes the paper.

## 2 Mathematical Model of Multicast Network

We begin with description of the mathematical model for the real multicast network as presented in [7, 5]. In this model, the physical multicast tree is represented by a tree comprising actual network elements (the nodes) and communication links connecting them.

Let  $T = (V, L)$  denote a multicast tree with node set  $V$  and link set  $L$ . The root node 0 is the source of probe packets, and  $R \subset V$  denotes the set of leaf nodes representing the receivers. A link is said to be internal if neither of its endpoints is the root or a leaf node. Each non-leaf node  $k$  has a set of children node  $d(k) = \{d_i(k) \mid 1 \leq i \leq n_k\}$ , and each non-root node  $k$  has a parent  $p(k)$ . The link  $(p(k), k) \in L$  is denoted by link  $k$ . Write  $j \prec k$  if  $j$  is descended from  $k$ ,  $k = p^r(j)$  if  $j$  is  $r$ -level descended from  $k$ , where  $r$  is a positive integer. Let  $a(U)$  denote the nearest common ancestor of a node set  $U \subset V$ . Nodes in  $U$  are said to be siblings if they have the same parent, i.e., if  $f(k) = a(U), \forall k \in U$ . The subtree of  $T$  rooted at  $k$  is denoted by  $T(k) = (V(k), L(k))$ , and the receiver set  $R(k) = R \cap V(k)$ . Figure 1 shows an example of multicast tree model.

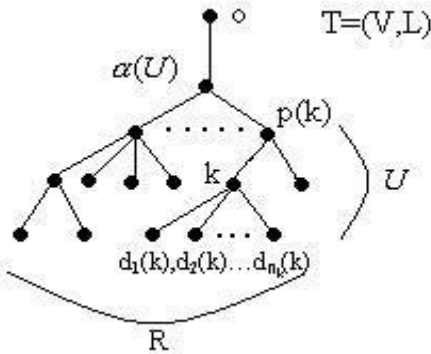


Figure 1. A multicast tree model

For each link an independent Bernoulli loss model is assumed with each probing packet being successfully transmitted across link  $k$  with probability  $p_k$ . Thus the progress of each probing packet down the tree is described by an independent copy of a stochastic process  $X = (X_k)_{k \in V}$  as follows.  $X_0 = 1$ ,  $X_k = 1$  if the probing packet reaches node  $k \in V$  and 0 otherwise. If  $X_k = 0$ ,  $X_j = 0, \forall j \in d(k)$ . Otherwise,  $P[X_j = 1 | X_k = 1] = p_j$  and  $P[X_j = 0 | X_k = 1] = 1 - p_j = \alpha_j$ . Define  $p_0 = 1$ . The pair  $(T, p)$  is called a loss tree.  $P_{T,p}$  denotes the distribution of  $X$

the loss tree  $(T, p)$ .  $X_k^{(i)}$  denotes the loss measurement of node  $k$  for  $i$ th probe packet. If  $0 < p_k < 1, \forall k \in V \setminus \{0\}$ , the loss tree is said to be a canonical tree. Any tree  $(T, p)$  in non-canonical form can be reduced to a canonical tree [7]. Henceforth only canonical loss trees are considered in this paper.

## 3 Binary Hamming Distance Classification Algorithm

Topology inference for multicast network in binary tree form is firstly considered for simplicity. Extension to general trees will be described later. We begin the section with a simple introduction on the previous algorithms. We then give the detailed description of the BHC algorithm.

As the description in Section 2,  $X_k^{(i)}$  is 1 if the  $i$ th probe packet reaches node  $k$  and 0 otherwise. Each node in the network maintained a "0-1" sequence  $\{X_k^{(i)}\}, 0 < i < n, k \in V$  after  $n$  probe packets are sent. The sequence maintained by internal routers is obtained by  $\bigvee_{l \in R(k)} X_l^{(i)}$  because the internal node is said to receive a probe packet surely if any receiver descended from it receives the probe packets. Based on all these "0-1" sequences, we can compare the correlation among different nodes so as to reconstruct the multicast network topology and infer the internal link characteristics.

In the previous work such as [7, 6, 5, 9, 12], the product of the probabilities of successful transmission on all the links between the root 0 to node  $k$ , denoted by  $A(k) = \prod_{j \succeq k} p_j$ , is used to infer the network topology.  $A(k) = \prod_{j \succeq k} p_j$  is the product of the probabilities of successful transmission on each link between  $k$  and the root 0. If the probability of one probe packet is successfully transmitted from the root to the nearest ancestor node of a receivers subset is minimized, the receivers are considered as siblings. That is, this approach identify the siblings by minimizing the following equation.

$$A(i, j) = \frac{P[\bigvee_{l \in R(i)} X_l = 1] P[\bigvee_{l \in R(j)} X_l = 1]}{P[\bigvee_{l \in R(i)} X_l = \bigvee_{l \in R(j)} X_l = 1]} \quad (1)$$

In practice,  $A(i, j)$  is estimated by  $A^{(n)}(i, j)$  as follows:

$$A^{(n)}(i, j) = \frac{\sum_{m=1}^n X_i^{(m)} \cdot \sum_{m=1}^n X_j^{(m)}}{n \cdot \sum_{m=1}^n X_i^{(m)} \cdot X_j^{(m)}}, \quad (2)$$

where  $X_k^{(m)} = \bigvee_{l \in R(k)} X_l^{(m)}$ . Because we cannot get actual probability in Equation (1), it is estimated by the observation from  $n$  probe packets, eg.,  $P[\bigvee_{l \in R(i)} X_l = 1]$  is estimated by  $\frac{\sum_{m=1}^n X_i^{(m)}}{n}$ . As  $n$  goes to infinity,  $A^{(n)}(i, j)$  is consistent with  $A(i, j), i, j \in V$ . However, it is obvious that the estimation causes bias with a finite number of probe

packets. Even worse,  $A^{(n)}(i, j)$  is used for determine every siblings pair, the bias brought by estimation may cause severe mistakes in topology inference. We addressed this problem in [11] and then proposed a solution by applying hamming distance.

As defined in [11], the hamming distance between nodes  $p$  and  $q$  can be obtained by the following equation, where “ $\oplus$ ” is the exclusive-OR operator.

$$H_d(u, v) = \sum_{m=1}^n X_u^{(m)} \oplus X_v^{(m)}. \quad (3)$$

The main reason why we apply hamming distance is that hamming distance is the simplest and most efficient method to identify the similarity and dissimilarity among different bit sequences. We also apply hop count information to the BHC algorithm because hop count provides level information on topology structure which avoids failure in inferring the single-child nodes in the network. Thus, we propose the BHC algorithm as follows, where each node  $k$  is associate with a hop count  $k.hop$ .

1. *Input*: The set of receivers  $R$ , number of probe packets  $n$ , observed sequences at receivers  $(X_k^{(i)})_{k \in R}^{i=1, \dots, n}$ ;
2.  $R' := R$ ,  $V' := \phi$ ,  $L' := \phi$ ,  $h = \max_{k \in R}(k.hop)$ ,  $W_m = \phi$ , ( $m = 1, \dots, h$ );  $V'$  is the set of nodes that was discovered;  $L'$  is the set of discovered links,  $W_m$  is a set of nodes with hop count value  $m$ ,  $m$  is initialized as the maximum value of hop count  $h$  for all nodes in  $R$ .
3. for  $k \in R$ , do
4.  $W_{k.hop} := W_{k.hop} \cup \{k\}$
5. while  $m > 1$  do
6. while  $W_m \neq \emptyset$  do
7. Let  $u$  be the first element in  $W_m$ ; search for  $v \in W_m$  to minimize  $H_d(u, v)$ , ( $u \neq v$ );
8. if  $H_d(u, v) > \delta_m$  then  $S = \{u\}$ , Set  $r$  to be  $u$ 's virtual parent node; //Initially, sibling nodes set  $S := \emptyset$ ,  $\delta_m$  is a given threshold.
9. else  $S = \{u, v\}$ , Set  $r$  to be  $u$  and  $v$ 's virtual parent node;
10. for  $i = 1, \dots, n$  do  $X_r^{(i)} := \bigvee_{l \in S} X_l^{(i)}$ ;
11.  $r.hop := m - 1$ ;
12.  $V' := V' \cup S$ ;  $W_m := W_m \setminus S$ ;  $W_{m-1} := W_{m-1} \cup \{r\}$ ;
13. for each  $l \in S$ ,  $L' := L' \cup \{(r, l)\}$ ;
14.  $S = \emptyset$ ;
15.  $m := m - 1$ ;
16.  $V' := V' \cup \{0\}$ ;  $L' := L' \cup \{0, r\}$ ;
17. *Output*: Inferred topology  $(V', L')$ .

Firstly all the receivers are classified into different node sets  $W_m$  ( $1 \leq m \leq h$ ) according to their values of hop count. The hamming distances of each node pair in  $W_m$

are calculated. Inference begins with identifying siblings in the node set with maximum value of hop count. The node pair is identified to be siblings if its hamming distance is minimal and less than the threshold. Remove the siblings from the node set with the hop count being  $m$  and add the parent node into the node set with hop count reduced by 1. The “0-1” sequence of the parent node is obtained by “OR” operation of those of the siblings. When all nodes in  $W_m$  are grouped decrease hop count value by 1. Repeat the same procedure among the nodes in the node set  $W_{m-1}$ . The algorithm ends when the hop count becomes 1.

## 4 Comparison of Inference Accuracy

Our algorithm and the previous algorithms can get the consistent result with the real multicast network as the number of probe packets increases to infinity. With a finite number of probe packets, we will prove in this section that our BHC algorithm can obtain accurate results with a higher probability, because the hamming distance is found to be superior to  $A^{(n)}(i, j)$  used in [7, 6, 5, 12] in many cases.

**Definition 1** Let  $s_1$  and  $s_2$  be two receivers that have a common ancestor node  $i$ ,  $s_3$  be a receiver for which node  $i$  is not an ancestor,  $W_k$  is the node set with the hop count  $k$ . Define  $s(i)$  as follows:

$$s(i) = \{(s_1, s_2, s_3) : \forall s_1, s_2, s_3 \in W_k, 1 \leq k \leq h\}$$

**Definition 2** For  $(s_1, s_2, s_3) \in s(i)$ , let  $D_H(s_1, s_2, s_3)$  be the difference of hamming distance between non-siblings and siblings, and  $D_A(s_1, s_2, s_3)$  be the difference of  $A^{(n)}(\cdot, \cdot)$  between non-siblings and siblings. That is,

$$D_H(s_1, s_2, s_3) = H_d(s_1, s_3) - H_d(s_1, s_2),$$

$$D_A(s_1, s_2, s_3) = A^{(n)}(s_1, s_3) - A^{(n)}(s_1, s_2).$$

**Lemma 1** If  $\min_{(s_1, s_2, s_3) \in s(i)} D_H(s_1, s_2, s_3) > 0$ , node  $s_1$  and node  $s_2$  can be identified as siblings correctly. The same holds for  $D_A(s_1, s_2, s_3)$ .

Lemma 1 holds because the hamming distance or  $A^{(n)}(\cdot, \cdot)$  of a node and its non-sibling nodes should be greater than that of it and its siblings.

We denote by  $n_{s_i}^1$  the number of probe packets transmitted from the root to node  $s_i$  successfully, and by  $n_{s_i s_j}^1$  the number of probe packets transmitted successfully from the root to both nodes  $s_i$  and  $s_j$  at the same time,  $i = 1, 2, 3$ .

**Lemma 2** For  $(s_1, s_2, s_3) \in s(i)$ , if inequality (4) holds, the hamming distance approach can identify the siblings while  $A^{(n)}(\cdot, \cdot)$  cannot; if inequality (5) holds,  $A^{(n)}(\cdot, \cdot)$

can identify the siblings while the hamming distance approach cannot; in all other cases, both approaches can identify siblings correctly.

$$\frac{1}{2}(n_{s_2}^1 - n_{s_3}^1) < n_{s_1 s_2}^1 - n_{s_1 s_3}^1 \leq \frac{n_{s_1 s_3}^1}{n_{s_3}^1}(n_{s_2}^1 - n_{s_3}^1), \quad (4)$$

$$\frac{n_{s_1 s_3}^1}{n_{s_3}^1}(n_{s_2}^1 - n_{s_3}^1) < n_{s_1 s_2}^1 - n_{s_1 s_3}^1 \leq \frac{1}{2}(n_{s_2}^1 - n_{s_3}^1), \quad (5)$$

proof

Since  $H_d(s_1, s_2) = n_{s_1}^1 + n_{s_2}^1 - 2n_{s_1 s_2}^1$  and  $A^{(n)}(s_1, s_2) = \frac{n_{s_1}^1 \cdot n_{s_2}^1}{n \cdot n_{s_1 s_2}^1}$ , we have,

$$\begin{aligned} D_H(s_1, s_2, s_3) &= n_{s_1}^1 + n_{s_3}^1 - 2n_{s_1 s_3}^1 - (n_{s_1}^1 + n_{s_2}^1 \\ &\quad - 2n_{s_1 s_2}^1) \\ &= (n_{s_3}^1 + 2(n_{s_1 s_2}^1 - n_{s_1 s_3}^1)) - n_{s_2}^1, \end{aligned} \quad (6)$$

and

$$\begin{aligned} D_A(s_1, s_2, s_3) &= \frac{n_{s_1}^1 \cdot n_{s_3}^1}{n \cdot n_{s_1 s_3}^1} - \frac{n_{s_1}^1 \cdot n_{s_2}^1}{n \cdot n_{s_1 s_2}^1} \\ &= \frac{n_{s_1}^1}{n \cdot n_{s_1 s_2}^1} \cdot \left( (n_{s_3}^1 + \frac{n_{s_3}^1}{n_{s_1 s_3}^1} \cdot (n_{s_1 s_2}^1 - \right. \\ &\quad \left. n_{s_1 s_3}^1)) - n_{s_2}^1 \right). \end{aligned} \quad (7)$$

From Lemma 1, we know that nodes  $s_1$  and  $s_2$  will be identified as siblings if  $D_H(s_1, s_2, s_3) > 0$  using the hamming distance approach for any  $(s_1, s_2, s_3) \in s(i)$ . If  $(s_1, s_2, s_3)$  results in  $D_H(s_1, s_2, s_3) < 0$ , the hamming distance approach will fail to identify  $s_1$  and  $s_2$  as siblings correctly. Similar conditions holds for  $A^{(n)}(\cdot, \cdot)$  approach. Therefore, we can conclude that if any  $(s_1, s_2, s_3) \in s(i)$  results in  $D_A(s_1, s_2, s_3) < 0$  while  $D_H(s_1, s_2, s_3) > 0$ , the hamming distance approach is superior to the  $A^{(n)}(\cdot, \cdot)$  approach in siblings identification, and vice versa. Lemma 2 describes the complete conditions based on equations (6) and (7). #

Based on Lemma 1 and Lemma 2, we find that in most cases, the performance of the hamming distance approach and  $A(\cdot, \cdot)$  approach are almost same according to our detailed discussion on the receiving of nodes  $s_1, s_2$  and  $s_3$ . However the occurrence of the hamming distance approach outperforming  $A(\cdot, \cdot)$  approach is more frequent than that of  $A(\cdot, \cdot)$  outperforming the hamming distance approach. As we have found, if only links  $s_1$  and  $s_2$  are in the similar condition, the hamming distance approach can surely identify siblings correctly when  $A(\cdot, \cdot)$  approach can do so. What's more, it can also identify siblings correctly while  $A(\cdot, \cdot)$  approach fails in some cases. For instance, when both nodes  $s_1$  and  $s_2$  receive almost all probe packets while node  $s_3$  loses many probe packets, and  $n_{s_1 s_3}^1$  happens to be

equal to  $n_{s_3}^1$ , the hamming distance approach can identify siblings correctly while  $A(\cdot, \cdot)$  cannot. When both nodes  $s_1$  and  $s_2$  lose many probe packets while  $s_3$  receives almost all probe packets, and  $\frac{n_{s_1 s_3}^1}{n_{s_3}^1} < \frac{1}{2}$ , the hamming distance approach can also identify the siblings correctly which  $A(\cdot, \cdot)$  approach fails.

In very few cases, the hamming distance approach cannot identify siblings correctly while  $A(\cdot, \cdot)$  approach can. This might happen only when links  $s_1$  and  $s_2$  are in different conditions which result in dissimilar sequences on nodes  $s_1$  and  $s_2$ . In a multicast network, only a few siblings links may exhibit completely different performances. Even if under this condition, it should be noted that in most cases the hamming distance approach can also identify siblings correctly as  $A(\cdot, \cdot)$  approach does. Thus from all cases discussed, we can see that the probability that the hamming distance approach succeeds but  $A(\cdot, \cdot)$  approach fails is greater than the probability in the opposite situation. Validation on this by simulation is given in Section 3.4.

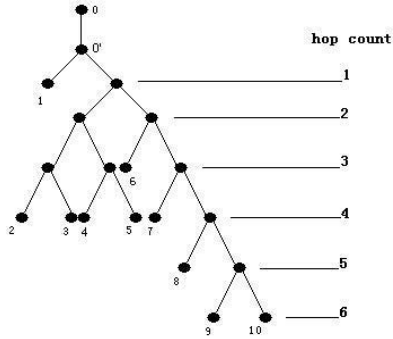
Though obtaining an exact probability of hamming distance approach outperforming  $A(\cdot, \cdot)$  approach is very difficult because it varies with different network connections and conditions, we have seen from our analysis that with a finite number probe packets, the hamming distance approach is more likely to work out the accurate topology than  $A(\cdot, \cdot)$ . In another word, due to the greater probability of the hamming distance approach outperforming  $A(\cdot, \cdot)$  approach in siblings identification, we may conclude that the use of the hamming distance approach in the BHC algorithm can result in better performance in inference accuracy.

## 5 Simulation Results

In this section, we validate the BHC algorithm by comparing it with HBLT which provides the best combination of inference accuracy and computational efficiency among all previous topology inference algorithms as [7, 6, 5, 12, 9]. The proof on the superiority of HBLT can be referred to [9]. Therefore, the comparison between BHC and HBLT is sufficient to justify the superiority of the BHC algorithm.

The simulation uses a network topology shown in Figure 2. Node 0 is the sender, nodes 1 – 10 are receivers. All the links are configured at different capacities. Routers are set low memories. The root node 0 generates probe packets in a 20Kbit/s-1Mbit/s stream. Every probe packet comprises one UDP request packet with 1000bytes.

Due to the low capacity and high IP background utilization of links, multicast probe packets can be delayed or even lost. Then for each receiver, the collected data is set to 1 if the probe packet is received, otherwise 0. Both algorithms work on the collected "0-1" sequences. From TTL field of the probe packets obtained from receivers, hop count re-



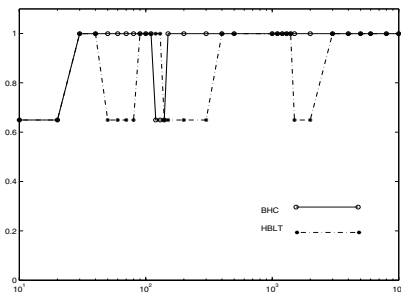
**Figure 2.** An Multicast Network Topology

quired by BHC and HBLT can easily be obtained.

To describe how close the inferred topology is to the original physical network, we define similarity degree function as follows:

**Definition 3** Define similarity degree to be  $\alpha \cdot s + \beta \cdot p$ , where  $s$  denotes the ratio of the number of nodes whose siblings are identified correctly to the total number of nodes,  $p$  denotes the ratio of the number of nodes whose parent nodes are inferred accurately to the total number of nodes.  $\alpha, \beta$  are the weights of these two factors. When all the inferred descendants of a node are similar to the real descendant of the node, we say that the node is inferred correctly.

Let  $\alpha = 0.5$  and  $\beta = 0.5$ , then the compared result is shown in Figure 3.



**Figure 3.** Similarity degree comparison of HBLT and BHC when  $\alpha = 0.5, \beta = 0.5$

Figure 3 shows that 150 probe packets are enough for BHC to get the accurate topology constantly. With the same number of probe packets, however, HBLT cannot achieve this in our simulated network. Our experiments show that HBLT tends to infer accurate topology constantly when the number of probe packets exceeds 3000. Therefore, we can conclude that BHC can work out the accurate topology with fewer probe packets than that needed by HBLT in our sim-

ulated network. That is, BHC is more efficient in topology inference than the HBLT algorithm.

Figure 3 also validates our analysis on siblings identification by the hamming distance approach and  $A(\cdot, \cdot)$  approach. It shows that the probability that hamming distance approach outperforming  $A(\cdot, \cdot)$  approach is greater than the probability of the opposite situation.

## 6 Topology Inference for General Trees

It is more complicated to infer the topology for general trees than the binary case. We introduce a threshold  $\epsilon$  into grouping the siblings set  $S$ . The set  $S$  is grouped if the hamming distance between any pair of nodes in  $S$  is sufficiently close to being minimal.

The grouping step starts by finding a pair of nodes  $\{u, v\}$  that has the minimum hamming distance in  $S$ , then adjoining further elements to it provided the following inequality is satisfied:

$$H_d(u, v')(1 - \epsilon) < H_d(u, v). \quad (8)$$

Thus we replace line 9 of the BHC algorithm by the following steps so that topology inference for general trees can be performed.

- 9a. else  $\{S = \{u, v\};$
- 9b. while there exists  $v' \in W_m \setminus S$  such that  $H_d(u, v')(1 - \epsilon) < H_d(u, v)$  do
- 9c.  $S := S \cup \{v'\};$
- 9d. Set  $r$  to be the virtual parent node of all identified siblings in  $S$ .

We set  $\delta_m$  to  $\lg m \cdot \frac{n}{k}$  in order to avoid mistakenly identifying siblings due to the existence of two single nodes, where  $k$  is the estimated expected number of branches of the multicast network.

As presented in [7], the violation of the condition described in inequality (15) has the interpretation that the ancestor  $a(U)$  is separated from  $a(\{u, v\})$  by a link with loss rate at least  $\epsilon$ . The convergence of the inferred topology to the true topology is mainly influenced by  $\epsilon$ . If only  $\epsilon$  is less than the internal link loss rates, the inferred topology will be convergent to the true topology. However, the internal link loss rates are unknown in advance. A small value of  $\epsilon$  is more likely to satisfy the above condition but at the cost of slow convergence. A large value of  $\epsilon$ , on the other hand, is more likely to result in systematically removing links with small loss rates which causes wrongly convergence. However, those links with high loss rates are accurately inferred in both cases which is enough for many applications in practice. Thus, the choice of  $\epsilon$  doesn't pose a serious problem. In order to obtain more accurate topology practically, we envision the scheme of incorporating the link loss inference into topology inference as we recently proposed in [10].

## 7 Conclusion

An improved algorithm for multicast network topology inference, the binary hamming distance and hop count based classification algorithm (BHC), has been proposed in this paper. With a finite number of probe packets, the topology inferred by BHC has been proved to be more accurate than those inferred by the previous algorithms. Through simulation, it has also been shown that BHC significantly outperforms the previous algorithms in efficiency. At the end of this paper, the proposed algorithm has been extended to topology inference for general trees.

## Acknowledgement

This work was supported by the 21st Century COE Program of Ministry of Education, Culture, Sports, Science and Technology, and by Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research under its General Research Scheme (B) Grant No. 14380139.

## References

- [1] B.N.Levine, S.Paul, and J.J.GARCIA-Luna-Aceves. Organizing multicast receivers deterministically according to packet-loss correlation. In *Proceedings of ACM Multimedia'98*, 1998.
- [2] R. Caceres, N. G. Duffield, J. Horowitz, F. L. Presti, and D. Towsley. Loss based inference of multicast network topology. In *IEEE Conference on Decision and Control, Phoenix, AZ*, 1999.
- [3] R. Castro, M. Coates, and R. Nowak. Maximum likelihood identification from end-to-end measurements. Technical report, DIMACS Workshop on Internet Measurement, Mapping and Modeling, DIMACS Center, Rutgers University, Piscataway, New Jersey, February 2002.
- [4] D.Li and D.R.Cherton. OtersOn-Tree Efficient Recovery using Subcasting:a reliable multicas protocol. In *Proceedings of IEEE ICNP'98*, 1998.
- [5] N. Duffield, J. Horowitz, F. Presti, and D. Towsley. Multicast topology inference from end-to-end measurements. Technical report, ITC Seminar on IP Traffic Measurement, Modeling and Management, Monterey, CA, Sept. 2000.
- [6] N. Duffield, J. Horowitz, and F. L. Presti. Adaptive multicast topology inference. In *IEEE Infocom 2001, Anchorage, Alaska*, April 22-26, 2001.
- [7] N. G. Duffield, J. Horowitz, F. L. Presti, and D. Towsley. Multicast topology inference from measured end-to-end loss. *IEEE Trans. Information Theory*, 2002.
- [8] S. Ratanasamy and S. McCanne. Inference of multicast routing tree topologies and bottleneck bandwidths using end-to-end measurements. In *Proc. IEEE Inforcom'99, New York, NY*, 1999.
- [9] H. Tian and H. Shen. Analysis on binary loss tree classification with hop count for multicast topology discovery. In *Proceedings of IEEE CCNC'04*, Jan. 2004.
- [10] H. Tian and H. Shen. Multicast-based inference of network internal loss performance. In *Proceedings of ISPAN'04*, May. 2004.
- [11] H. Tian and H. Shen. Discover multicast network internal characteristics based on hamming distance. In *Pro. IEEE ICC'2005(to appear)*, May, 2005.
- [12] H. Tian and H. Shen. An improved algorithm of multicast topology inference from end-to-end measurements. In *Proceedings of the 5th Int'l Symp. on High Performance Computing ISHPC-V*, Oct. 2003.