

# DHMT: A Scalable and Efficient Hybrid Application-layer Multicast Protocol

(Work-in-Progress)

Jing Li<sup>1,2,3</sup>, Naijie Gu<sup>1,2</sup>, Weijia Jia<sup>2,3</sup>

<sup>1</sup>Department of Computer Science and Technology, University of Science & Technology of China, Hefei, China

<sup>2</sup>Joint Research Lab of Excellence, CityU-USTC Advanced Research Institute, Suzhou, China

<sup>3</sup>Department of Computer Science, City University of Hong Kong, Hong Kong, China

[jennylee@mail.ustc.edu.cn](mailto:jennylee@mail.ustc.edu.cn), [gunj@ustc.edu.cn](mailto:gunj@ustc.edu.cn), [itjia@cityu.edu.hk](mailto:itjia@cityu.edu.hk)

## ABSTRACT

A scalable and efficient hybrid application-layer multicast protocol-DHMT (Dynamic Hybrid Multicast Tree) is proposed in this paper. It adopts the hybrid architecture of hierarchy and Fibonacci tree to multicast packets efficiently. In DHMT, the underlying topology characteristic is considered by introducing an idea of local area to reduce packet delivering on costly links. In each local area, members are constructed into a hierarchical architecture in term of layer and cluster. The roots of all local hierarchical architecture are organized into a Fibonacci multicast tree. In addition, the cores in DHMT are selected dynamically to reflect the instantaneous network situation. The architecture characters improve the delay performance of DHMT. The simulation shows that DHMT is scalable and efficient and it works well especially for large multicast group.

## Categories and Subject Descriptors

C.2.2 [Network Protocols]: routing protocols.

## General Terms

Algorithms, Design, Experimentation

## Keywords

Application-layer multicast, hybrid architecture, Fibonacci tree

## 1. INTRODUCTION

Application-layer multicast (ALM) was proposed solve the lack of ubiquitous multicast support of IP multicast among all internet service providers. It builds an overlay architecture by having the end hosts self-organize into logical overlay networks. The multicast packets are replicated and forwarded by end hosts. It offers multicast function by unicasting on underlying links.

Current proposed ALM protocols can be classified into three flavors according to overlay topology design: the mesh-first, tree-first and implicit ALM protocols. NARADA [1] is a mesh-first ALM protocol. It firstly organizes the multicast group members into a mesh topology and then constructs a spanning tree whose root is the multicast source. It is a robust protocol, but the scalability is bad. YOID [2] is a tree-first ALM protocol. It firstly builds a shared data delivery tree among members. The tree structure is easy to construct and has logarithmic scaling behavior.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

INFOSCALE 2007, June 6-8, Suzhou, China  
Copyright © 2007 ICST 978-1-59593-757-5  
DOI 10.4108/infoscale.2007.218

But YOID is not efficient. NICE [3] and CAN-based multicast [4] is two examples of implicit ALM protocol. CAN-based multicast maps the multicast group members into Content-Addressable Network (CAN) and multicasts the packets with “flooding scheme” to all the neighbors. NICE is hierarchical infrastructure. It involves several layers and each layer has a set of clusters. NICE is a scalable protocol. However, it does not consider underlying topology and it compromises the delay performance. Therefore, a novel ALM protocol-DHMT is proposed in this paper. It is a hybrid architecture which involves hierarchical tree and Fibonacci tree.

## 2. DHMT ARCHITECTURE

In DHMT, the idea of hierarchy and cluster is adopted. Two innovations are the considering of underlying topology characteristic when clustering and the use of Fibonacci multicast tree to achieve the short delay performance. A new introduced conception “local area” means that end hosts in it all attach to the same router. In each local area, group members are constructed into a hierarchical tree. The cores of each local area are organized into a Fibonacci multicast tree.

In each local area, group members are partitioned into different clusters with the size between  $k$  and  $3k-1$  where  $k$  is a constant. Every cluster has a cluster core achieved by using the dynamic core selection algorithm based on the instantaneous network situations. DHMT employs the on-demand probing scheme to search the member who has the minimum sum of unicast delays to other members in a cluster. The cluster cores in layer  $L_i$  compose higher layer  $L_{i+1}$ . The hierarchy will continue until the number of members is not larger than  $3k-1$ . The cluster core in the highest layer is the local core.

The Fibonacci series based multicast algorithm is adopted to organize the local cores into a Fibonacci multicast tree. The input of this algorithm is a local core sequence constructed under considering delay distance and the size of local area. The algorithm constructs a multicast tree by partitioning the group members sequence into different sizes based on the idea of Fibonacci series. The Fibonacci series  $\{f_i\}$

satisfies  $f_0=0, f_1=1; f_n=f_{n-1}+f_{n-2}, \text{ if } n>1$ .

### Algorithm 1 (Fibonacci series based multicast):

Input: member sequence  $\Phi=(d_1, d_2, \dots, d_k)$ ,  $d_s$  is source node,  $d_s \in \Phi$ . The number of members in  $\Phi$  is  $K$ ,  $f_n \leq K < f_{n+1}$

Output: a multicast tree constructed for all members in  $\Phi$

- 1 If ( $K=2$ )  $d_s$  send packets to the only destination;
- 2 If ( $K>2$ ) partition  $\Phi$  into  $\Phi_1$  and  $\Phi_2$  where  $d_s$  is in the larger subsequence and the smaller one includes  $f_{n-2}$  members;
  - 2.1 If ( $s > f_{n-2}$ )  $\{ \Phi_1 = \langle d_1, d_2, d_3, \dots, d_{f_{n-2}} \rangle; \Phi_2 = \langle d_{f_{n-2}+1}, d_{f_{n-2}+2}, \dots, d_K \rangle; \}$   
 Else  $\{ \Phi_1 = \langle d_1, d_2, \dots, d_{K-f_{n-2}} \rangle; \Phi_2 = \langle d_{K-f_{n-2}+1}, d_{K-f_{n-2}+2}, \dots, d_K \rangle; \}$
  - 2.2 If ( $s < f_{n-2}$ )  $\{ d_s$  firstly sends packets to  $d_1$ ,  $d_1$  and  $d_s$  are in charge of multicasting in  $\Phi_1$  and  $\Phi_2$  respectively;  $\}$   
 Else  $\{ d_s$  firstly sends packets to  $d_{K-f_{n-2}+1}$ ,  $d_s$  and  $d_{K-f_{n-2}+1}$  are in charge of multicasting in  $\Phi_1$  and  $\Phi_2$  respectively;  $\}$
- 3 Multicast packets from  $d_1$  to all members in  $\Phi_1$  and from  $d_s$  to all members in  $\Phi_2$  (or from  $d_s$  to all members in  $\Phi_1$  and from  $d_{K-f_{n-2}+1}$  to all members in  $\Phi_2$ ) by recursive calls Algorithm 1

DHMT employs the refresh messages to follow the membership alterations in the group. When there is an end host to join or leave, it sends corresponding message to related members and some adjustments are made for the whole architecture.

### 3. SIMULATION EVALUATION

We use the NS-2 to compare the performance of NICE, CAN-based multicast and DHMT. Performance comparisons are given along the metrics: 1) Average End-to-end Delay (AED): it is the ratio of the sum of end-to-end delay from a source to each group member to the number of members; 2) Average Link Stress (ALS): it is the ratio of the sum of times that identical packet copies traverse over the underlying links to the number of links in the group.

The first simulation is to observe AED and ALS when the number of group members varies from 50 to 600 and 1 sending sources. Figure 4(a) illustrates the comparison of AED performances. The flooding routing in CAN-based multicast incurs much longer AED than the other two protocols. DHMT achieves better AED performances than NICE for the existence of the local area and the Fibonacci multicast tree. Figure 4(b) illustrates the ALS performances. DHMT achieves better ALS performances than NICE. The flooding routing scheme enables CAN-based multicast to achieve the smallest ALS. In the second simulation, the AED and ALS are observed under the situation that the number of the sending source varies and the number of group members is always 100. Figure 5(a) illustrates the AED performance when the number of sending source varies from 1 to 10. The flooding scheme leads the CAN-based multicast to create the worst AED. The characters of DHMT make it achieve better AED than in NICE. Figure 5(b) illustrates the AED of NICE and DHMT when the number of sending source varies from 30 to 50. NICE has worse AED than DHMT because it neglects the underlying network properties. The simulations illustrates that DHMT has better AED and ALS performance.

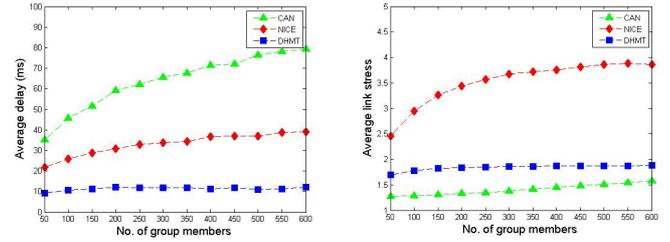


Fig. 4. The AED and ALS performance of the three protocols with the increasing group members

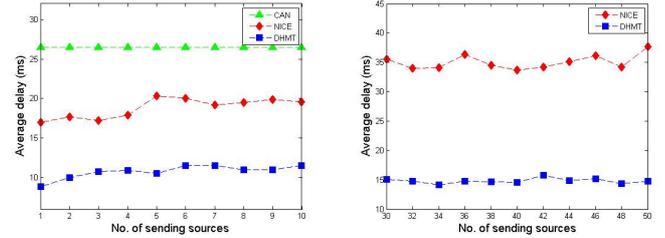


Fig. 5. The AED performance with 100 members and the sending sources varies from 1 to 10 and from 30 to 50

### 4. CONCLUSION

Focusing on the scalability and efficiency of ALM, DHMT is proposed based on the hierarchy and the Fibonacci series. A hierarchical architecture is built in each local area by partitioning the group members into clusters and layers. All local cores are organized into Fibonacci multicast tree which can results in short delay performance. The considering of underlying network properties alleviates the problem of redundant packet transferring on costly links. The simulation results show that DHMT has its benefits compared with traditional solutions and it is efficient and scalable.

### 5. ACKNOWLEDGEMENTS

This research is supported by the National Basic Research Fund of China ("973" Program) under Grant No.2003CB317003, and has been benefited from various discussions among the group members of the Joint Research Lab between CityU and USTC in their advanced research institute in Suzhou (China).

### 6. REFERENCES

- [1] Y.H. Chu, S.G. Rao and H. Zhang, A case for end system multicast. *Proc. of ACM SIGMETRICS*, June 2000.
- [2] P. Francis, Yoid: Extending the multicast internet architecture. *White paper* <http://www.aciri.org/yoid/>, April 2000.
- [3] S. Banerjee, B. Bhattacharjee and C. Kommareddy, Scalable application layer multicast. *Proc. of ACM SIGCOMM*, August 2002.
- [4] S. Ratnasamy, M. Handley, R. Karp, Application-level multicast using content-addressable networks. *Proc. of NGC*, 2001.