

## A SDN-like Loss Recovery Solution in Application Layer Multicast

Wenqing Lei<sup>1</sup>, Cheng Ma<sup>1</sup>, Xinchang Zhang<sup>2, a</sup>, Lu Wang<sup>2</sup>

<sup>1</sup>Key Laboratory of Information Security and Intelligent Control in Universities of Shandong, Shandong Youth University of Political Science, Jinan 250103, China

<sup>2</sup>Shandong Provincial Key Laboratory of Computer Networks, Shandong Computer Science Center(National Supercomputer Center in Jinan) , Jinan, 250101, China

<sup>a</sup>email: zhangxc@sdas.org

**Keywords:** Application Layer Multicast; Loss Recovery; Delivery Tree

**Abstract.** Most existing Application layer multicast (ALM) solutions use a tree structure to deliver the multicast data. However, the data distribution of tree-based ALM is unreliable because of group members' dynamics and the error correlation of the ALM tree. In this paper, we propose a SDN-like loss recovery solution, called DALMLR, for application layer multicast. This solution uses some additional proxy servers to provide reliable loss recovery service, and uses a SDN-like method to compute the loss recovery path. Owing to the centralized recovery path computing way and the reliable loss recovery proxies, DALMLR can provide rapid loss recovery service.

### Introduction

Application layer multicast (ALM) is an important communication way for group applications (e.g. video conference and file sharing). In ALM, network infrastructures need no additional modification, which addresses the problem of non-ubiquitous deployment of IP multicast. Most existing ALM solutions use a tree structure, consisting of the data source and receivers, to deliver the multicast data. The members of a multicast group are dynamic in nature. That is, the participants may join and leave the group at any time. A member's departure will cause all its descendants, in the ALM tree, to become disconnected from the ALM tree for some time, which brings some data loss. Group members' dynamics and the above error correlation problem make the data distribution of tree-based ALM unreliable [1].

There have been some schemes for improving the reliability of tree-based ALM. The proxy-based solutions (e.g., OMNI [2]) use some additional proxy servers to assist in disseminating the multicast data. PRM [3] employs a proactive component in which each group member forwards received data to some other members with a low probability, and uses a reactive mechanism to handle data losses due to link errors and network congestion. Some ALM solutions use TCP to connect adjacent nodes, to attempt to provide reliable data delivery service. Jin et al. pointed out that the ALM solution based on TCP cannot provide reliable ALM service [4]. [5] presents a UDP-based reliable ALM solution. Most existing solutions heavily depend on group members. Therefore it is difficult for them to overcome the limitation caused by the dynamics and resource constraints of group members.

The emerging software defined networking (SDN) technique separates data plane and control plane. In SDN, The switches in the data plane simply forward the data packets, and the controller manipulates network behaviors in a centralized way. In this paper, we propose a SDN-like loss recovery solution, called DALMLR, for application layer multicast. This solution uses some additional proxy servers to provide reliable loss recovery service, and uses SDN-like method to arrange the recovery packet routing. The proposed solution can provide rapid loss recovery service for tree-based ALM.

## Related Work

So far there have been many ALM protocols which address how to use tree structure to deliver the multicast data from the source to all the receivers [6] [7] [8] [9]. As mentioned previously, the data delivery is unreliable in the tree-based ALM.

Some ALM solutions attempt to provide reliable multicast service based on the TCP connections between adjacent nodes. Simon Wong et al. pointed out that (1) TCP-based reliable approach may not achieve high throughput due to TCP backoff mechanism, and (2) the hosts at the leaves of the delivery tree may suffer from high delay, as a data segment has to be completely received before being forwarded downstream, packet-by-packet manner for the reliable service [5].

In [5], a UDP-based reliable ALM solution LER is proposed. In this solution, group members are randomly divided into several planes, and each member selects some members in other planes as its recovery neighbors. LER randomly divides group members into several planes and independently builds an overlay tree in each plane. In each plane, a member acts as the multicast tree root (i.e., the plane source). LER takes high measurement and computation overheads to select proper recovery neighbors [4].

[1] proposes a hierarchical loss recovery solution, called HR, which can be used as a reliable extension to existing tree-based ALM protocols. In HR, the group members at the top of the ALM tree are in a recovery plane (i.e., Plane 1), and the members at the bottom of the ALM tree are in another recovery plane (i.e., Plane 2). HR employs a robust and quick approach to recover the losses at the group members in Plane 1, which can avoid the potential losses at the downstream nodes and provide many relatively reliable recovery sources. HR uses a loss recovery approach, which can effectively reduce the link load, to recover the loss found by the group member in Plane 2. In HR, the recovery packet is retransmitted by an area-constrained multicast means, which can recover the loss at one or multiple group members with little recovery diffusion.

Recently the technology of software defined networking (SDN) emerges [10] [11] [12] [13], which separates the network control plane from the data forwarding plane with the promise to dramatically improve network resource utilization, simplify network management, reduce operating cost, and promote innovation and evolution. The SDN controller is capable to collect information from network devices and change settings of under-layer infrastructure for different traffic flows. With the full knowledge of network condition, SDN controller can adaptively set up different routes for different flows so as to maximize service utility.

## DALMLR Design

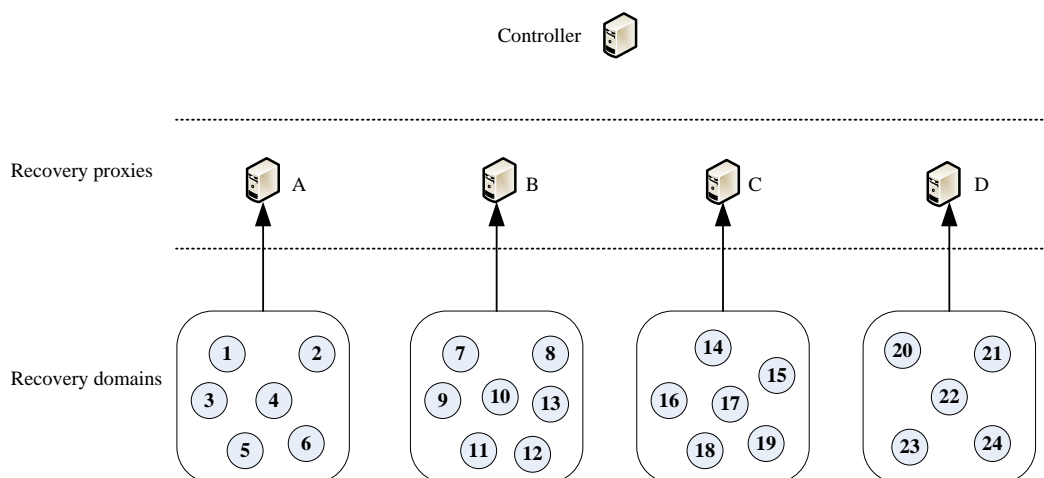


Fig.1. Infrastructure of DALMLR

Figure 1 shows DALMLR's infrastructure. DALMLR employs some proxies to provide loss recovery service for application layer multicast. Additionally, it also uses a controller to manage the

loss recovery proxies and arrange loss recovery path. The multicast data is disseminated to group member along the ALM tree and directly disseminated to loss recovery proxies. These proxies buffer all the multicast data until the corresponding multicast session ends. Each loss recovery proxy is responsible for a recovery domain, and each group member belongs to a specific recovery domain. Each loss recovery proxy periodically reports the bandwidth using ratio and the number of group members in the domain that it is responsible for to the controller.

We say a proxy is a candidate proxy of a group member if the distance from the proxy to the group member does not exceed an expected bound. When a group member joins the multicast session, it sends a message to the controller to inquire the existing loss recovery proxies. After receiving the loss recovery proxy list from the controller, it measures the distances between the loss recovery proxies and itself. Then it sends a registration message, including candidate proxies, to the controller to apply for joining a loss recovery domain. The controller returns the candidate proxy whose corresponding domain has the least members. Subsequently, the group member joins the above proxy. Figure 2 explains the above procedure.

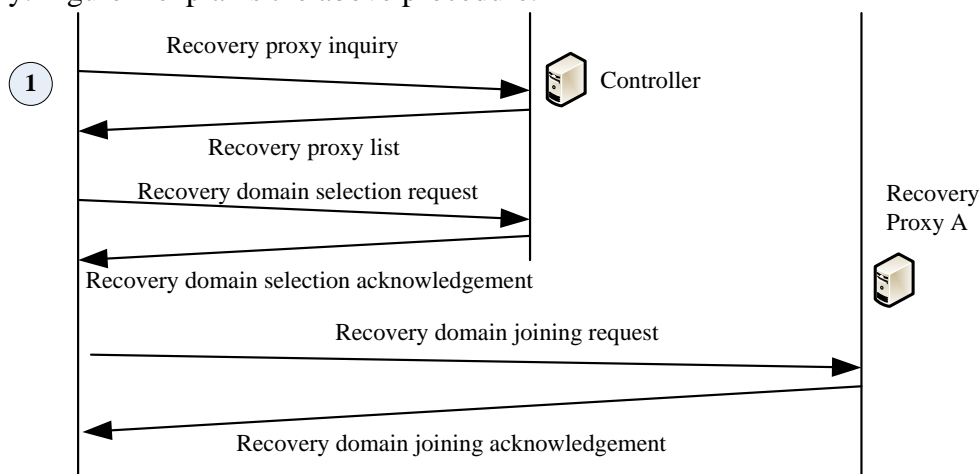


Fig.2. Recovery domain registration

When a group member  $m$  detects a data loss, it first sends the loss recovery request to the proxy  $A$  that is responsible for the domain which it belongs to. If the proxy has available bandwidth, it provides the loss recovery service to this group member. Otherwise, it informs this group member to select other proxy. If  $m$  cannot obtain the loss recovery service from  $A$ , it sends a message to the controller. The controller checks the bandwidth using situation of candidate proxies (except  $A$ ) and returns the candidate proxy whose available bandwidth is maximal. Then  $m$  obtains the loss recovery service from this candidate proxy. Figure 3 explains the above procedure.

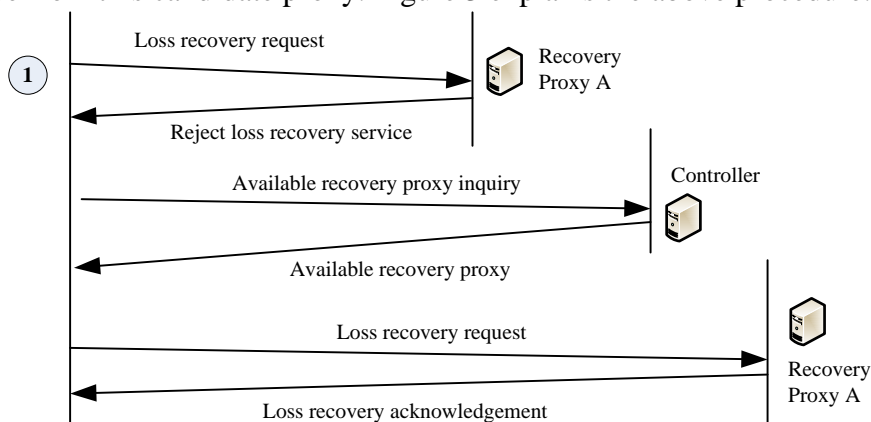


Fig.3. Loss recovery procedure

From above introduction, we can see that our idea is very like that of SDN. Specially, the controller has an overall view of the loss recovery, and thereby decides how to implement the loss recovery. Since the loss recovery proxies are reliable and the recovery path selection needs at least two rounds, our solution has rapid and reliable loss recovery capability.

## Experiment results

In this section, we use NS2 [13] to evaluate the performance of DALMLR. We used GT-ITM generator to create a 4,200-node transit-stub graph as underlying network topology. The proxies and group members were connected to random stub-domain nodes. We use HCCast as the application example of DALMLR [14]. To evaluate the loss recovery performance, we produce data loss by the following way: After each group member successfully received  $\alpha$  packets, it lost  $\delta$  packets with the probability of  $\sigma$ . In our experiment, 5 loss recovery proxies were used.

We use loss recovery failure ratio to denote the ratio of the number of the recovery requests, which cannot be provide corresponding recovery service, to that of total recovery requests. Figure 4 and 5 describe the average loss recovery rounds and loss recovery failure ratios in 10 groups with different scales of group members. Note that data points in the two figures represent the averages over 100 runs. From the two figures, we can see that the DALMLR can provide rapid loss recovery service and has high reliability.

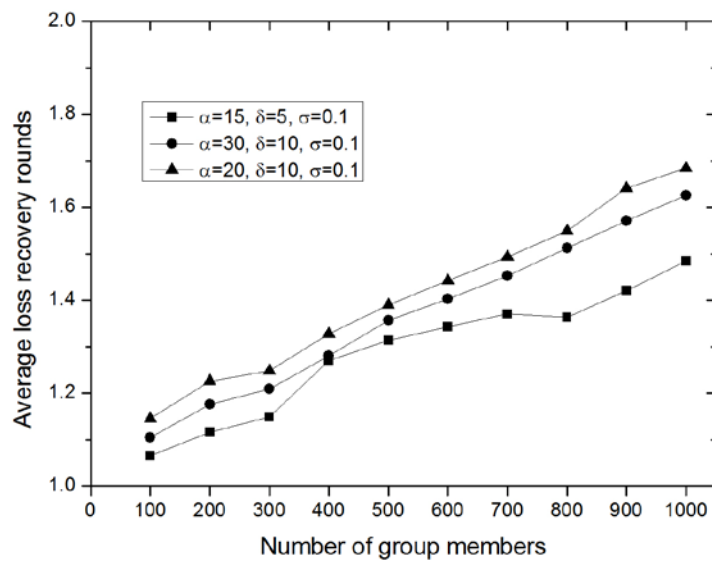


Fig.4. Average loss recovery rounds

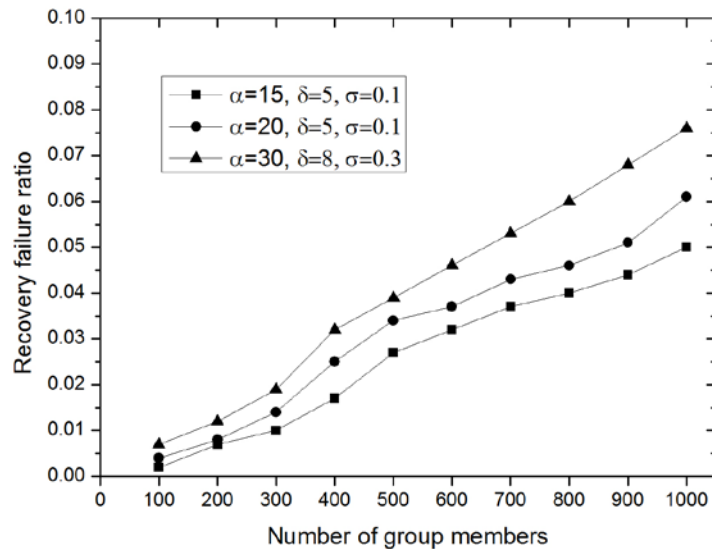


Fig.5. Recovery failure ratio

## Conclusion

To solve the unreliability problem of tree-based ALM, this paper proposes a SDN-like loss recovery solution DALMLR. This solution uses some additional proxy servers, instead of the dynamic group members, to provide loss recovery service. Therefore the recovery sources are reliable. Like the SDN technique, DALMLR uses a centralized way to compute the loss recovery path. DALMLR can provide rapid loss recovery service for tree-based ALM.

## Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant No. 61272433 and 61472230, and the Shandong provincial Natural Science Foundation of China under Grant No. ZR2013FM010 and ZR2014YL042.

## References

- [1] Xin-Chang Zhang, Mei-Hong Yang, Xiao-Jing Zhu, Wan-Ming Luo. A loss recovery approach for reliable application layer multicast [J]. *The Journal of Systems and Software*, 2012 85 1198-1204.
- [2] S. Banerjee, C. Kommareddy, K. Kar, B. Bhattacharjee, S. Khuller. OMNI: An efficient overlay multicast infrastructure for real-time applications [J]. *Computer Networks*, 2006 50(6) 826-841.
- [3] S. Banerjee, S. Lee, B. Bhattacharjee, A. Srinivasan. Resilient Multicast using Overlays [J]. *IEEE/ACM Transaction on Networking*, 2006 14(2) 237-248.
- [4] X. Jin, W.P.K. Yiu, S.H.G. Chan. Loss Recovery in Application-Layer Multicast [J]. *IEEE Multimedia*, 2008 15(1) 18-27.
- [5] K.-F. Simon Wong, S.-H. Gary Chan, W.-C. Wong, Q. Zhang, W.-W. Zhu, Y.-Q. Zhang. Lateral error recovery for application-level multicast. *Proceedings of IEEE Infocom* [C]. 2004.
- [6] Li Lao, Jun-Hong Cui, Mario Gerla, Shigang Chen. A Scalable Overlay Multicast Architecture for Large-Scale Applications [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2007 18(4) 449-459.
- [7] Xiaolong Li, A. D. Striegel. A Case for Passive Application Layer Multicast [J]. *Computer Networks*, 2007 51(11) 3157-3171.
- [8] M. Kobayashi, H. Nakayama, N. Ansari, N. Kato. Reliable Application Layer Multicast Over Combined Wired and Wireless Networks [J]. *IEEE Transactions on Multimedia*, 2009 11(8) 1466-1477.
- [9] Zhou Su, M. Oguro, Y. Okada, J. Katto, S. Okubo. Overlay Tree Construction to Distribute Layered Streaming by Application Layer Multicast [J]. *IEEE Transactions on Consumer Electronics*, 2010 56(3) 1957-1962.
- [10] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: enabling innovation in campus networks [J]. *ACM SIGCOMM Computer Communication Review* 2008 38(2) 69–74.
- [11] R. Sherwood, M. Chan, A. Covington, G. Gibb, M. Flajslik, N. Handigol, T.Y. Huang, P. Kazemian, M. Kobayashi, J. Naous, et al. Carving research slices out of your production networks with openflow [J]. *ACM SIGCOMM Computer Communication Review* 2010 40(1) 129–130.
- [12] Guohan Lu, Rui Miao, Yongqiang Xiong, and Chuanxiong Guo. Using cpu as a traffic co-processing unit in commodity switches. *Proceedings of the first workshop on Hot topics in*

software defined networks [C]. 2012.

[13] The Network Simulator - ns-2. <http://www.isi.edu/nsnam/ns>.

[14] X. Zhang, X. Li, W. Luo, B. Yan. An Application Layer Multicast Approach Based on Topology-Aware Clustering. *Computer Communications*, 2009 32(6) 1095-1103.