Why Are You Still Using Shortest Path?

– Path Selection Strategy Utilizing High-functional Nodes –

Taro Hashimoto†, Katsunori Yamaoka††, and Yoshinori Sakai††
†,††Tokyo Institute of Technology
2–12–1–S3–68 Ookayama, Meguro, Tokyo, 152–8552, Japan
†taro@net.ss.titech.ac.jp, ††{yamaoka, ys}@ss.titech.ac.jp,

I. Introduction

Live streaming is delay sensitive and their end-to-end *allowable delay* is an important factor. Allowable delay means the elapsed time in which data should arrive at an end receivers before being played back. Data arriving at end receivers later than the allowable delay are useless and cannot be played back. Therefore, not only must data losses be reduced but also a specific end-to-end data delay must be maintained to ensure the QoS of the application.

For live streaming media, routing problems have been an important issue for ensuring the QoS of applications. Generally in unicast routing, the shortest path is often used to minimize the delay from the sender to the receiver. Also, multicast tree configuration depends on the shortest path tree to minimize the delay from the sender to receivers, or the minimum spanning tree to minimize the network resources of the whole tree. As an example shown in Fig. 1, when the traffic conditions of a certain link on the multicast tree become worse and a bottleneck develops during data transmission, a multicast reconfiguration method can avoid the bottleneck link (link B-A) and modify the part of a given multicast tree adaptively to recover and improve the QoS against traffic variations. In multicast tree reconfiguration [1], [2], the shortest delay path is frequently used as an alternate path to ensure a small end-to-end delay.

However, the Shortest Path Tree (SPT) reconfiguration is not always appropriate when *high-functional nodes* are placed on part of nodes on the network. High-functional node is a node with a special capability, such as loss detection, recovery, etc., to improve the QoS of an application, in addition to the functions of normal node. In this situation, the QoS of applications varies depending on the number of high-functional nodes and their locations. However, the shortest delay path selection method does not take into account the existence of these nodes, and thus, it might not be able to select a path with a sufficient number of high-functional nodes. As an example shown in Fig. 1, alternate paths, (C-A) and (D-A) were found. The path (S-C-A-R) is the shortest delay path from the sender to the receiver and has only four high-functional nodes. The other path (S-F-E-D-A-R) has a slightly larger delay than the shortest delay path, but it includes six high-functional nodes. Generally, previous path reconfiguration methods, such as PS-SPT [2], would select the shortest delay path from the sender to the receiver, i.e. (S-C-A-R). However, assuming variations in traffic, the shortest delay path might not always be able to maintain the QoS, due to the lack of high-functional nodes. Instead, the path (S-F-E-D-A-R) might be able to maintain the QoS because it has a sufficient number of high-functional nodes. It does not matter which path from sender to receiver is taken as long as the QoS of the end receiver is sufficient. A path that can deal with variations in traffic effectively should be selected.

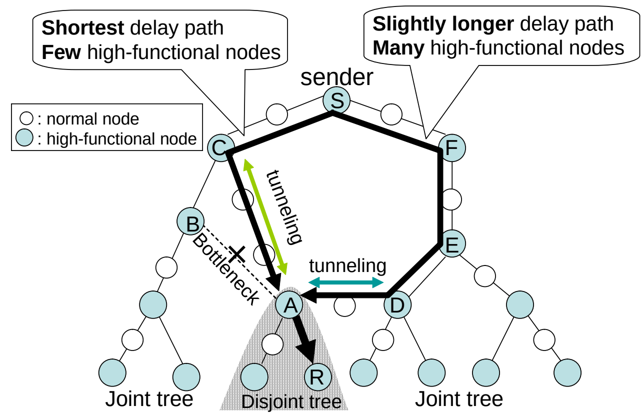


Fig. 1. Two paths for reconfiguration

Therefore, we propose a new path selection method utilizing high-functional nodes for multicast tree reconfiguration. Specifically, we focus on important factors for path selection that take into account not only the delay on the path but also the number of relay nodes and their locations, which we call the *path selection strategy*. Then, we develop three path selection methods that follow the strategy. Our methods provide a multicast tree with tolerance for variations in traffic after reconfiguration.

Here, our path selection method focuses on multicast tree reconfiguration. However, its idea is not limited to this. In the environment that high-functional nodes are partly placed on the network, our path selection strategy and method can be applied to the other models, such as unicast, P2P, ALM, and overlay networks.

II. Path Selection Utilizing High-functional Nodes

A. QMLS router as High-functional Node

First, we targeted at QoS Multicast for Live Streaming (QMLS) protocol [3] as an example protocol that uses high-functional relay nodes on a multicast tree. This protocol defined two types of losses, *network loss* and *application loss*. Some packets are lost while being sent to receivers; these are called network losses. Packets with delays exceeding the allowable delay cannot be played back; these are called application losses.

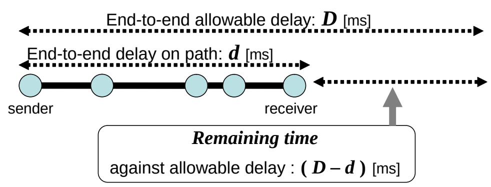


Fig. 2. Remaining time against allowable delay

The purpose of the QMLS protocol is to reduce the total losses, which are the sum of network and application losses, in live streaming multicast as much as possible within allowable delay constraints. Specifically, this protocol focuses on allowable delay. A QMLS sequence number is assigned to each packet and used by QMLS routers. When lost packets are detected by a gap in the QMLS sequence numbers, the QMLS router requests retransmission from an upstream QMLS router immediately using NACKs. The router also stores forwarded packets for future retransmission for a limited time in its cache. It can therefore retransmit the packets that are in its cache when downstream QMLS routers request them.

Therefore, QMLS protocol can improve the QoS of the applications.

B. Path Selection Strategy

We determined the following path selection strategy for utilizing high-functional relay nodes.

1) Delay: The delay on the path should be still as short as possible because live streaming media require small end-to-end delays from the sender to receivers. Therefore, a path with near the shortest delay should be selected.
2) Allowable delay: A specific amount of extra time is required to recover lost packets due to retransmission delays. Therefore, the *remaining time* against the allowable delay should be as long as possible for future retransmissions (Fig. 2). Remaining time means $(D-d)$, where d [ms] is the delay on the path and D [ms] is the end-to-end allowable delay required by applications.
3) Number of relay nodes: The more relay nodes are placed on a path, the better they can detect and recover lost packets. Such a path has the potential to maintain the QoS of applications in case of packet losses.
4) Distance between two adjacent relay nodes: One retransmission requires two times the amount of delay between two adjacent relay nodes. Therefore, each distance between adjacent pair of relay nodes (each link in the QMLS overlay network) should be as short as possible.

These strategies should guarantee an efficient relay process that can deal with variations in traffic after reconfiguration.

C. Path selection Algorithm

From here onward, we use a simplified model in which all the nodes on the multicast tree are QMLS routers, which act as a high-functional relay node at any time. We can consider an overlay network of QMLS routers for path reconfiguration, which is possible using tunneling between nodes.

As shown in Fig. 1, when a link in the multicast tree becomes a bottleneck, the multicast tree is divided into two sub-trees (*disjoint tree* and *joint tree*) and an alternative path is selected to recover the tree.

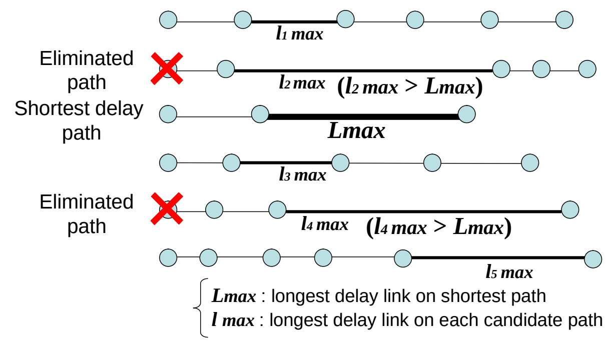


Fig. 3. Path elimination from multiple candidates

First, each alternate shortest path is searched for from the root node of disjoint tree to each on-tree relay node (in the joint tree). A path from its on-tree relay node to the sender can exist on the on-tree path in the joint tree so that the service of the joint tree is not disrupted, and most links of the multicast tree therefore remain the same. Consequently, we can obtain multiple candidate paths for reconfiguration using an alternate path (from the root node of the disjoint tree to an on-tree relay node) and the on-tree path (from the on-tree relay node to the sender).

Then, we should select an appropriate path from these candidates. Here, we propose the following three path selection methods, taking the path selection strategy into consideration.

1) PSDR: The Path Selection method taking into account Delay and Relay nodes (PSDR) selects a path out of multiple candidates, based on path selection strategies 1, 2, and 3, as discussed in Sect. II-B. Each candidate path is evaluated using the following function:

$$EV(r, d) = r \times (D - d) \tag{1}$$

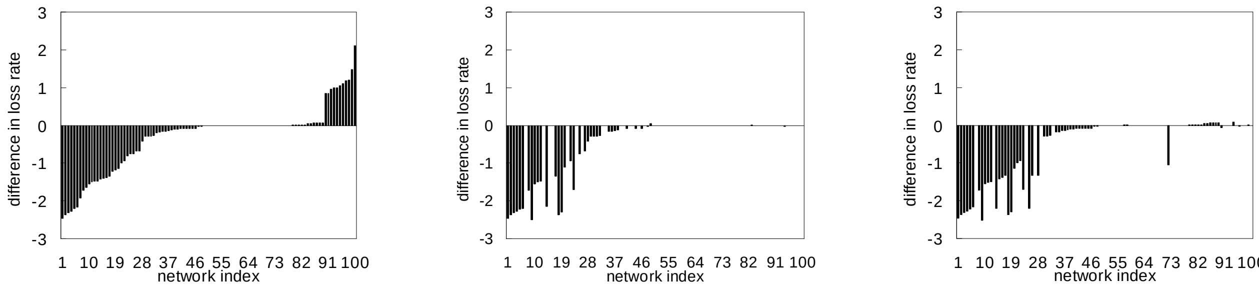
where EV is the evaluated value, r is the number of relay nodes on the path, D [ms] is the end-to-end allowable delay and d [ms] is the delay on the path.

Note that, according to strategy 1, larger values of r are preferable. In addition, according to strategies 2 and 3, the *remaining time* against the allowable delay $(D - d)$ should be as long as possible. For this reason, r and $(D - d)$ are multiplied together to select the path which has the maximum value for EV.

2) PSDR-DP: PSDR with a limited Distance between relay nodes with Parameters (PSDR-DP) focuses on strategy 4, in addition to strategies 1, 2, and 3. PSDR-DP eliminates the unsuitable paths that include extremely long delay links from candidates first. For this purpose, we use the following path elimination inequality to decide whether a focused path should be eliminated from candidates:

$$l_{max} \leq L_{max} \times \alpha \tag{2}$$

where L_{max} [ms] is the longest distance between two adjacent relay nodes on the shortest delay path of candidates, and l_{max} [ms] is the longest distance on a focused candidate path. $\alpha\,(0 \leq \alpha \leq 1.0)$ is a parameter that regulates the amount of eliminated paths from candidates. As the value of α gets smaller, the number of eliminated paths increases. The candidate path that does not meet Eq. (2) is eliminated first,



(a): PSDR (b): PSDR-DP ($\alpha = 0.7$) (c): PSDR-RP ($\beta = 0.45$)

Fig. 4. Difference in loss rate compared to SPT

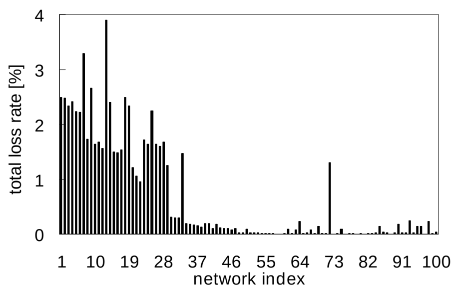


Fig. 5. Total loss rate of SPT

and then Eq. (1) is applied to the rest. An example is shown in Fig. 3.

3) PSDR-RP: PSDR with limitations on the Retransmission delay using Parameters (PSDR-RP) also focuses on strategy 4, in addition to strategies 1, 2, and 3. One retransmission requires two times the amount of delay between two adjacent relay nodes. This makes remaining time become shorter. Therefore, PSDR-RP focuses on the relationship between the retransmission delay on the l_{max} and the *remaining time* $(D-d)$. We use following inequality:

$$l_{max} \leq (D-d) \times \beta \tag{3}$$

where $\beta\,(\geq 0)$ is a parameter that regulates the amount of eliminated paths from candidates. As the value of β gets smaller, the number of eliminated paths increases. Any candidate path that does not meet Eq. (3) is eliminated first, and then Eq. (1) is applied to the rest.

III. Evaluation

A. Simulation

Network topologies were randomly generated by topology generator. First, we assumed one link with a maximum delay on the multicast tree was a bottleneck and divided the multicast tree into two regions. Then, the multicast tree was reconfigured by SPT and our proposed method using PSDR, PSDR-DP, and PSDR-RP. Then, CBR traffic as stream flow was transmitted from the sender to all receivers. Packet drop rate on each node were varied randomly from 0 to 10%, which represents buffer overflow on each node due to other incoming traffic. For evaluation, we used the path with longest delay from the sender to the receiver, which had been located in the disjoint tree region. We evaluated the total losses of the receiver.

B. Simulation Results

The total loss rate on the receiver using SPT is shown in Fig. 5, where the x-axis shows indices of 100 different network topologies. Each topology has 60 nodes and 10 degree in average, respectively. *Differences in loss rates* as compared to SPT are shown in Fig. 4, to evaluate the improvement by using our method. The difference in the loss rate is described as the *(loss rate for each of our path selection methods) – (loss rate of SPT)* on each network topology. A negative value in the graph means that the proposed method recovered losses better than SPT and decreased the total loss rate, while a positive value means the opposite. Note that the network indices on the x-axis in Fig. 4-(a) were sorted in ascending order of difference in loss rate between PSDR and SPT. In addition, the orders of the network indices on each x-axis in Figs. 4 and 5 are the same as Fig. 4-(a).

PSDR is shown to generally improve the loss rate compared to SPT in Fig. 4-(a), however, some degraded cases, such that inappropriate path with long distance link is selected, can be seen. PSDR-DP ($\alpha = 0.7$) selected a path with a loss rate that was better than or comparable to SPT, and it did not select inappropriate paths on any networks (Fig. 4-(b)). Furthermore, PSDR-RP ($\beta = 0.45$) did not select inappropriate paths on any networks and had a loss rate that was better than or comparable to SPT (Fig. 4-(c)). This shows that it is effective to refrain from choosing paths with too long distance links. The average improvement in the loss rate of our proposed methods compared to SPT was 45.7 % (PSDR), 58.0 % (PSDR-DP), and 70.7 % (PSDR-RP), respectively. This shows that multicast tree reconfigured by our methods tolerates variations in traffic.

IV. Conclusion

We proposed a new path selection method utilizing high-functional nodes for multicast tree reconfiguration for live streaming media. We focused on a path selection strategy that takes into account not only the delay on the path but also the number of relay nodes and their locations on the path. For future work, we intend to apply our method to models where both high-functional nodes and normal nodes exist.

References

[1] D. Zappala, "Alternate Path Routing for Multicast," *IEEE/ACM Trans. Netw.*, vol.12, no.1, Feb. 2004, pp.30–43.

[2] W.Y. Tseng and S.Y. Kuo, "Real-time Multicast Routing with Efficient Path Selection for Multimedia Application," *IEICE Trans. Inf. & Syst.*, vol.E84-D, no.7, July 2001, pp.889–898.

[3] Y. Somchit, A. Kobayashi, K. Yamaoka, and Y. Sakai, "A Proposal for a Multicast Protocol for Live Media," *IEICE Trans. Commun.*, vol.E88-B, no.3, March. 2005, pp.1128–1138.