Gateway implementation for evaluating jitter reduction packet forwarding control method

Hikaru YAMAGUCHI†, ***Nonmember***, **Masaki MURAKAMI**†, ***Student member***, **Takashi KURIMOTO**†, ***Member***, **Satoru OKAMOTO**†, **and Naoaki YAMANAKA**†, ***Fellows***

1. Introduction

Mission-critical applications such as Virtual Reality and advanced driving systems require low-latency and low-jitter packet transmission. To achieve low jitter, a method that assigns accumulated jitter to packets and performs priority control has been proposed [1]. This paper confirms that this method can be applied to real network environments.

2. Packet control methods for jitter reduction [1]

In a network node, packet contention occurs when packets input from multiple links are output to a single link. In this case, packet ordering control is necessary to realize low jitter forwarding.

The packet control method for jitter reduction adds a timestamp to each packet when it is input to and output from a network node. The accumulated jitter is calculated using the timestamps at each node and added to the packet. Each node updates the current accumulated jitter value using the accumulated jitter value up to the previous node and the input/output time at the previous node. The difference between the accumulated jitter value and the target jitter value for that packet determines the priority of that packet. As shown in Fig. 1, the node performs sequencing control and outputs packets based on the priority of packets in each queue. At the time, packets with higher jitter are prioritized to achieve low jitter. In addition, this method has little impact on latency because low jitter is achieved by priority control. This impact is also not a problem because jitter is generally more demanding than latency as an application QoS requirement.

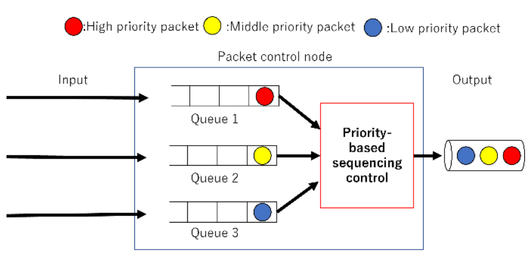

Fig. 1 Priority control method in the routing node.

†The author is with Graduate School of Science and Technology, Keio University, Kohoku-Ku, Kanagawa, 223-8522, Japan.

3. Gateway and forwarding node implementation

We implemented a proof of concept (PoC) system using the Click Modular Router [2] for packet forwarding. The forwarding node requires a timestamp and accumulated jitter value to control the packet forwarding order. A 32 bytes tag is added to each packet. Therefore, we have implemented a function at the gateway node that adds the tags to packets entering the network and removes the tags from packets leaving the network. This function is implemented in Python with Scapy. We examined the gateway function in the PoC.

In the experiment, we applied a gateway to send and receive UDP packets as they passed through the network. As a result, packet communication was confirmed at both ends of the gateway. We also confirmed that the forwarding nodes in the network recorded timestamps on the tags assigned by the gateway, and that reduction of end-to-end jitter was worked in the network by controlling the order of the packets.

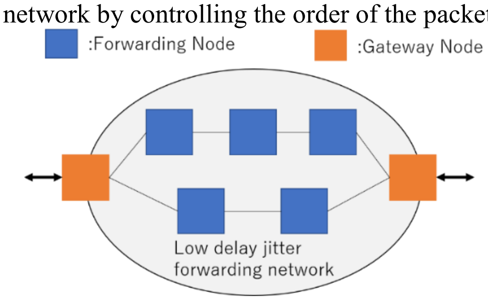

Fig. 2 Network overview with gateway installed.

4. Conclusions

We have confirmed that the proposed packet forwarding method can be applied to end-to-end packet transmission. In the future, we plan to apply them to actual applications.

Acknowledgments

This work is partly supported by the commissioned research (02501) of the National Institute of Information and Communications Technology(NICT), Japan.

References

[1] Y.Nasu, T.Kurimoto, S.Okamoto, and N.Yamanaka, "Packet-by-packet priority control for few microseconds order jitter application," International Conference on Emerging Technologies for Communications(ICETC2021), C2-3, Dec. 2021.

[2] E.Kohler, et al. "The Click modular router." ACM Transactions on Computer Systems (TOCS) 18.3 (2000): 263-297.