Invocation of Expire procedure with Increase in RTT

Natsuki Katsumata [†a)], Kohei Ogawa [†], *non-member*, Saneyasu Yamaguchi [†], *Member*

1. Introduction

Some works [1][2] reported that the obtained throughputs are not fair in case TCP BBR and CUBIC TCP communicate simultaneously sharing the bottleneck link. A method for improving fairness by invoking the *expire* procedure for BBR was proposed [3]. In this work, we evaluated the method in a single-connection environment and show that the *expire* method's performance is less than that of the original method. We then propose to call the *expire* procedure when the most recently measured RTT has increased 10 consecutive times.

2. Related work

In the work of [3], a method to improve the fairness of throughput between CUBIC TCP and TCP BBR was proposed. This method invokes the *expire* procedure of the TCP BBR when the ratio of *RTT* divided by *RTprop* is greater than *rate_th*, which is a threshold value. We call this method the *expire* method. The throughput obtained by the *expire* method without other connections is significantly lower than that of the original one.

3. Expire with Increase in RTT

We propose a method for improving fairness without a large performance decline when it does not share the bottleneck link. We call the method the *expire*+ method in this paper. The method conducts the *expire* procedure when the following two conditions are satisfied.

1. RTT/RTprop is greater than the *rate_th*
2. RTT is increasing in the 10 times recently measure RTT.

The *expire* method conducts the procedures only with the first condition.

4. Evaluation

We evaluated the throughputs with *rate_th* 2.5. Fig. 1 shows the fairness index when they communicate simultaneously. The results indicate that both the *expire* and *expire*+ methods improve the throughput fairness remarkably compared with the original TCP BBR. Fig. 2 shows the throughputs of the method without sharing the bottleneck link. These show that the throughput of the *expire* method is less than that of the original method. On the contrary, the performance of the *expire*+ is similar to that of the original TCP BBR.

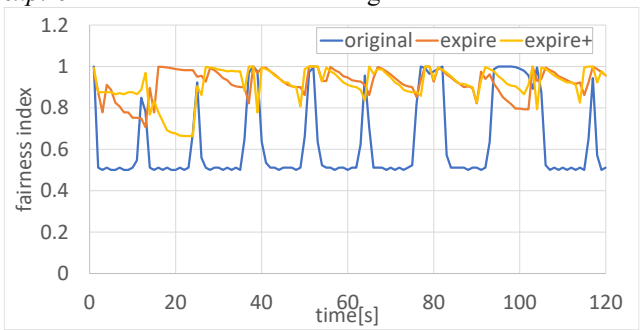

Fig. 1 throughputs fairness with sharing

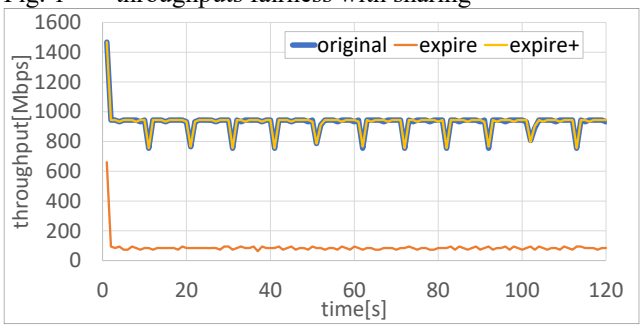

Fig. 2 throughputs without sharing

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 21K11854 and 21K11874.

References

[1] M. Hock, R. Bless and M. Zitterbart, "Experimental evaluation of BBR congestion control," *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, Toronto, ON, 2017, pp. 1-10. doi: 10.1109/ICNP.2017.8117540

[2] K. Miyazawa, K. Sasaki, N. Oda and S. Yamaguchi, "Cycle and Divergence of Performance on TCP BBR," 2018 IEEE 7th International Conference on Cloud Networking (CloudNet), 2018, pp. 1-6, doi: 10.1109/CloudNet.2018.8549411.

[3] K. Sasaki, K. Miyazawa, K. Ogawa, S. Yamaguchi, "TCP BBR Performance Improvement on a Network with Increasing RTT," The Eighth International Symposium on Computing and Networking (CANDAR 2020), Nov. 2020.

[†]The authors are affiliated with the Kogakuin University, Shinjuku-ku, Tokyo, 163-8677, Japan.

[a)] E-mail: cm21010@ns.kogakuin.ac.jp