A bandwidth allocation scheme to improve fairness in data center networks*

Yusuke Ito^{1a)}, Hiroyuki Koga¹, and Katsuyoshi Iida²

¹ Graduate School of Environmental Engineering, University of Kitakyushu,
1-1 Hibikino, Wakamatsu, Kitakyushu, Fukuoka 808-0135, Japan

² Global Scientific Information and Computing Center, Tokyo Institute of Technology,
2-12-1 Ookayama, Meguro, Tokyo 152-8550, Japan

a) wash000i@net.is.env.kitakyu-u.ac.jp

Abstract: In cloud computing, users can now enjoy various Internet services provided by data centers (DCs) through a wide variety of access networks and communication terminals anytime, anywhere. However, the problem arises that a user's quality of service is unfair among users who are different distances from DCs. Therefore, we propose a fair bandwidth allocation scheme based on collectable information in DC networks. This scheme collects flow information from routers and servers in DC networks, and then fairly allocates transmission rates among flows based on the collected information. We show the effectiveness of this approach through simulation evaluations.

Keywords: bandwidth allocation, fairness, TCP, data center, data center networks

Classification: Network

References

- [1] J. Zhang, F. Ren, and C. Lin, "Survey on transport control in data center networks," *IEEE Netw.*, vol. 27, no. 4, pp. 22–26, Jul./Aug. 2013. DOI: [10.1109/MNET.2013.6574661](https://doi.org/10.1109/MNET.2013.6574661)
- [2] R. Meza, "A survey on congestion control," *Eur. J. Comput. Sci. Eng.*, vol. 10, 5 pages, 2013.
- [3] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Netw.*, vol. 1, no. 4, pp. 397–413, Aug. 1993. DOI: [10.1109/90.251892](https://doi.org/10.1109/90.251892)
- [4] R. Pan, B. Prabhakar, and K. Psounis, "CHOKe: A stateless active queue management scheme for approximating fair bandwidth allocation," Proc. IEEE INFOCOM2000, pp. 942–951, Mar. 2000. DOI: [10.1109/INFCOM.2000.832269](https://doi.org/10.1109/INFCOM.2000.832269)
- [5] I. Psaras, W. K. Chai, and G. Pavlou, "Differentiated bandwidth allocation with TCP protection in core routers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 1, pp. 34–47, Jan. 2009. DOI: [10.1109/TPDS.2008.71](https://doi.org/10.1109/TPDS.2008.71)
- [6] K. Kusuha, I. Kaneko, K. Iida, H. Koga, and M. Shimamura, "A unified congestion control architecture design to improve heterogeneous wireless

*Earlier version of this letter was presented at ACM CoNEXT2014 Student Workshop, Dec. 2014.

- network efficiency and accommodate traffic by various rich applications,” Proc. IFIP HET-NETs2013, 5 pages, Nov. 2013.
- [7] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling innovation in campus networks,” *ACM SIGCOM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Apr. 2008. DOI:10.1145/1355734.1355746
- [8] The ns-3 network simulator, <http://www.nsnam.org/>.
- [9] R. Jain, D. M. Chiu, and W. Hawe, “A quantitative measure of fairness and discrimination for resource allocation in shared systems,” DEC Research Report TR-301, Sep. 1984.

1 Introduction

Users can now connect to the Internet through a wide variety of access networks and communication terminals such as smart phones or PCs. This has created a great demand for cloud computing which allows users to enjoy various Internet services provided by data centers (DCs) [1] anytime, anywhere. Major services available through cloud computing include file sharing or transaction services which commonly use TCP [2] as a reliable data transmission protocol.

In cloud computing, the problem arises that a user’s quality of service is unfair among users who are different distances from DCs. This problem is caused by how TCP controls congestion. TCP cannot exactly identify the conditions of other flows, so it estimates an available bandwidth based on packet losses. Since congestion control is applied according to round-trip time (RTT), the transmission rate for a user with a short RTT is effectively increased compared with that of a user having a long RTT. Fairness in terms of throughput among flows with different RTTs needs to be improved.

To improve fairness among flows, many kinds of the active queue management (AQM) technologies applied on routers have been proposed. These technologies realize fair communication among flows, but the processing cost on routers increases enormously as the number of flows increases.

In this study, we propose a bandwidth allocation scheme based on collectable information in DC networks to provide fair communication service to each user. This scheme collects flow information including the bandwidth of each link, the number of competing flows, and the RTT of each flow from routers and servers in DC networks, and then fairly allocates transmission rates among flows based on the collected information. We show the effectiveness of this approach through simulation evaluations.

2 Related work

Many kinds of AQM algorithms have been proposed to improve fairness among flows. A typical AQM algorithm is RED [3]. This approach sets maximum and minimum queue lengths and drops packets when the current queue length is larger than the set maximum queue length. This approach realizes fair communication by preferentially discarding packets which belong to a high rate flow. However, it is difficult to determine an appropriate maximum and minimum queue length. This is

Fig. 1. Proposed scheme

because the appropriate values depend on network conditions such as the number of flows and delay time. Unlike RED, another AQM algorithm, called CHOKe, does not require parameter settings [4]. This approach randomly selects one packet from the output queue on routers when the router receives a packet. If the source addresses of the received and selected packets are the same, both packets should be dropped. In addition, CHOKeW, an improved version of the CHOKe algorithm, has been proposed [5]. This approach adjusts the packet dropping rate of each flow based on service classes and network conditions. These AQM algorithms improve fairness among flows, but the processing cost on routers increases enormously as the number of flows increases.

3 Proposed scheme

The proposed scheme fairly allocates transmission rates among flows based on collectable information in DC networks as shown in Fig. 1. Flow information such as the bandwidth of each link, the number of competing flows, and the RTT of each flow is collected by UCCA [6], which uniformly manages the congestion information in the networks. The RTT of each flow can be measured using TCP's timestamp option. If the server does not send any data to the client, it periodically sends probe packets to measure RTT. Routers in DC networks periodically inform servers of the number of existing flows, which can be measured by extending the OpenFlow architecture [7]. The servers identify bottleneck links based on the information about the number of existing flows received from the routers and the bandwidth of each link, and then calculate the transmission rate which should be allocated to each flow by dividing the bandwidth of the bottleneck link by the number of competing flows. Note that we focus on a case where bottleneck links exist in DC networks to allocate resources in DC networks to each flow fairly and effectively, so we will attack a case where bottleneck links exist outside DC networks in future work.

To allocate transmission rates to TCP flows with window-based congestion control, the bandwidth delay product of each flow as the allocated window size should be calculated based on the bandwidth which should be allocated and the RTT of each flow. Our scheme employs an allocation factor α and the allocated

window size is multiplied by α . For downwards communication from a server to a client, the server can simply allocate the calculated window size to each flow. On the other hand, for upwards communication from a client to a server, the server needs to inform the client of the allocated window size. Our scheme uses TCP's advertised window size to notify the client of the window size. This bandwidth allocation enables fair communication among flows which have different RTTs.

4 Simulation model

To investigate the efficiency of our scheme, we evaluated it through simulation using Network Simulator ns-3 [8] after its implementation. Fig. 2 shows the simulation topology. In this simulation, there are two client groups A and B, and these groups have access links with different delay times. The clients of each group communicate with the server using TCP NewReno. The propagation delay time of access links is set to a uniform random number which ranges from 2 to 6 ms for GroupA or from 10 to 16 ms for GroupB. The propagation delay time of other links is set to 1 ms. The bandwidth of all links is set to 200 Mb/s. The allocation factor α varies from 1.0 to 1.8. The interval time to notify the flow information is set to 0.05 ms.

We focus on two evaluation scenarios: join and leave scenarios. 25 clients of each group communicate with the server after the simulation starts. At 12 s after the simulation starts, one client of each group starts communication in the join scenario, while stops communication in the leave scenario.

We evaluate Jain's fairness index [9] and total throughput for the proposed scheme and conventional TCP. The fairness index is defined as Eq. (1), where x is the throughput of each flow and n is the number of existing flows.

$$f(x_1, x_2, x_3, \dots, x_n) = \frac{\left(\sum_{i=1}^n x_i \right)^2}{n \sum_{i=1}^n x_i^2} \quad (1)$$

Namely, fairness is higher as the index gets closer to 1. We evaluate the fairness index and total throughput averaged over 10 trials, which are calculated at intervals of 0.1 s.

5 Simulation results

We now show simulation results and discuss the effectiveness of the proposed scheme as compared with the conventional scheme. Figs. 3(a) and 3(b) show the fairness index and total throughput, respectively, of the proposed and conventional schemes in the join scenario. The proposed scheme achieves a higher fairness index than that of the conventional scheme. This is because the proposed scheme allocates an adequate window size to each flow according to its RTT. In particular, the proposed scheme with a small α achieves good fairness, but it degrades the total throughput. Otherwise, the proposed scheme with a large α achieves poor fairness and high throughput. Namely, the proposed scheme has a trade-off relationship

Fig. 2. Simulation topology

Fig. 3. Performance comparisons

between the fairness and total throughput for α . In this simulation, the proposed scheme with α of 1.2 enables good fairness and high total throughput.

In the leave scenario, the fairness index and total throughput of the proposed and conventional schemes are shown in Figs. 3(c) and 3(d), respectively. Similarly in the join scenario, the proposed scheme achieves a higher fairness index as compared with the conventional scheme. On the other hand, the proposed scheme with a small α drastically degrades the total throughput. This is because the proposed scheme only utilizes the allocated bandwidth, so the impact on the total throughput in the leave scenario is larger than that in the join scenario. However, the proposed scheme with α of 1.2 achieves good fairness and high total throughput

like the join scenario. Consequently, the proposed scheme enables fair communication among flows which have different RTTs as well as high total throughput by setting the adequate value of α (slightly larger than 1 in this simulation).

6 Conclusion

Our proposed bandwidth allocation scheme uses collectable information in DC networks to provide fair communication service to each user. Simulation evaluations have indicated that this scheme with the adequate allocation factor enables higher fairness and throughput performance than can be achieved with conventional TCP. In future work, we will consider the most effective way to dynamically adjust the allocation factor as well as to fairly allocate link bandwidth in DC networks to each user even when access networks are bottlenecks.

Acknowledgments

This work was supported in part by JSPS KAKENHI Grant Number 25280028.