DIAS: a Wait-Time based Distributed Information-agnostic Flow Scheduling

Zhuyun Qi[†§], Jie Xing[†], Keke Li[†], Kai Lei[†§], Bo Jin[‡], Yi Wang[‡Φ©]
[†]School of Electronic and Computer Engineering, Peking University, Shenzhen, China, 518055
[Φ] Guangdong Province Key Laboratory of Popular High Performance Computers, Shenzhen University, Shenzhen, China, 518067
[‡]SUSTech Institute of Future Networks, Southern University of Science and Technology, Shenzhen, China, 518055
[§]Pengcheng Laboratory, Shenzhen, China, 518055
[©]Corresponding Author: wy@ieee.org

***Abstract*—Existing flow scheduling schemes in Data Center Network (DCN) are designed mainly to minimize the flow complete time (FCT) of short flows and do not consider optimizing the FCT of latency-sensitive long flows. In this paper, we propose a distributed information-agnostic flow scheduling scheme (DIAS), which minimizes the FCT of both short flows and latency-sensitive long flows. In DIAS, packets are forwarded complying with their priorities, which are determined based on packets wait-time that is defined as staying time in end hosts' send buffers, and the longer a packet stays in a send buffer, the lower its priority. Meanwhile, instead of utilizing a central server to collect traffic load information, each switch feeds traffic load information which is used to adjust the thresholds of determining packets priority back to end hosts via ACK packets.**

The experimental results in ns-3 simulator show that DIAS reduces FCT by up to 54.7% and 50.1% over DCTCP and L^2DCT, respectively. Besides, DIAS ensures a smaller FCT of latency-sensitive long flows and performs better than PIAS.

I. Introduction

To achieve the minimum flow complete time (FCT) in Datacenter Networks (DCN), many flow scheduling schemes have been proposed and we can divide them into two groups: information-aware schemes (e.g. pFabric[1], PDQ[2], PASE[3], D^2TCP[4] and L^2DCT[5]), which require a priori knowledge of flow size or deadline information, and the information-agnostic scheme (i.e. PIAS[6]), which makes no assumption about the flow information. Among these two kind of schemes, information-aware flow scheduling schemes may be hard to use in reality due to the fact that it may be difficult or even unable to obtain the flow information in advance. So, PIAS performs better than others in reality.

However, PIAS has weaknesses that are listed as follows. Firstly, it needs a central server to manage the traffic load information, which is then issued to each end host to determine demoted thresholds for the priority queues. Secondly, nowadays the proportion of interactive artificial intelligence question&answer stream is increasing, which means that it is of great importance to optimize the FCT of latency-sensitive long flows as well. However, PIAS does not consider this.

Considering the flaws of existing solutions, we would like to design a Distributed Information-Agnostic flow Scheduling scheme to minimize the FCT for both short flows and latency-sensitive long flows. DIAS uses multiple priority queues in switches to implement Multilevel Feedback Queue (MFQ) and packets are forwarded according to their priorities, which are determined based on packets wait-time that is defined as the time during which packets stay in end hosts' send buffers, that is, the time difference between the instant that packets enter the buffer and the instant that packets leave the buffer. When a packet is about to be sent out by end hosts, hosts will attach a priority tag to the packet according to the wait-time. The longer a packet stays in a send buffer, the lower its priority. It is easy to see that if packets in the same flow enter the send buffer at the same time, then packets which are sent later need to wait for a longer time in the buffer, and may have lower priorities. What's more, for a latency-sensitive long flow, though the total number of packets which belong to the flow is large, these packets enter send buffers in intervals, and not all packets enter the buffer at the same time, thus the priorities of packets in latency-sensitive long flows are high, since these packets have a relatively short wait-time, just like short flows do. In addition, to dynamicly adjust the thresholds of priority queues, each switch needs to feed back the traffic load information to end hosts via ACK packets.

This paper is supported by Guangdong Province Key Laboratory of Popular High Performance Computers 2017B030314073, Shenzhen Key Fundamental Research Projects (Grant No.: JCYJ20170306091556329).

II. The DIAS Design

A. Packets Tagging in end hosts

In DIAS, end hosts are responsible for tagging packets and the process is shown in Fig.1. In Fig.1, we suppose that there are K priorities P_i, $1 \leq i \leq K$ and K wait-time thresholds α_j, $1 \leq j \leq K$ in switches, where priorities $P_1 \geq P_2 \geq ... \geq P_K$, $\alpha_1 \leq \alpha_2 \leq ... \leq \alpha_K$.

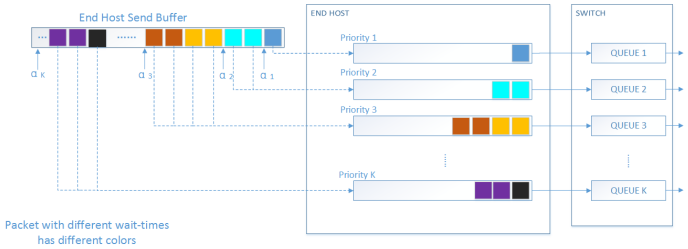

Fig. 1: DIAS overview

Each time a packet enters the send buffer, the end host tags the packet with an enter-in timestamp, and when the packet is about to be sent out, the end host will record a send-out timestamp. Obviously, the first packet that enters the send buffer has the shortest wait-time, and will be sent out with the highest priority P_1. For packets which enter the send buffer later, the wait-time becomes longer and the corresponding

priority decreases until the priority is set to P_K. The threshold to demote priority from P_{j-1} to P_j is α_{j-1}. When unexpected conditions occur (e.g. packets being dropped or time out), the wait-time is longer than α_K, in which case the end host will reset these packets with priority P_{K-1}, instead of resetting to the highest priority P_1.

B. Switch Design

When receiving packets, switches will put packet into *j*th queue based on its priority P_j. There are mainly three functions which are listed as follows in DIAS switches:

1) Priority scheduling: packets in switches are strictly dequeued based on their priorities.

2) ECN marking: in order to prevent packets in lower priority queues from starvation, DIAS applies the same approach as in DCTCP[7] to keep the queues length small.

3) Traffic load information feedback: when servers receive data requesting packets, they will retain and put the traffic load information, which is calculated by switches, to the returned ACK packet so that end hosts could adjust their thresholds.

C. Adjustment of thresholds

In DIAS, the central entity is no longer needed any more, each end host gets the corresponding traffic load information from its own packet transmitting path.

When a switch receives a packet (not an ACK packet), it calculated the traffic load information ρ based on the busy time and idle time: $\rho = T_{busy}/(T_{busy} + T_{idle})$. Then, if the traffic load information in the packet is smaller than ρ or the packet contains no traffic load information, the switch will update the traffic load information in the former case or attach the traffic load information to the packet in the latter case. Servers which receives a packet will retain the traffic load information and insert it into ACK packets. Using this way, end hosts could obtain the maximum value of traffic load information ρ_{max} which is used to adjust the demoted thresholds. Then each end host determines and updates its own thresholds based on its own traffic load information from the SYN ACK packet.

III. Evaluation

We use ns-3 simulator in which a dumbbell topology to evaluate DIAS. Besides, we disable dupACKs to avoid packet reordering and generate 400 flows in the experiment.

A. Comparison with existing Schemes

We compare DIAS with DCTCP, L^2DCT which is an information-aware scheme and PIAS that is an information-agnostic scheme, respectively. The evaluation result is shown in Fig.2 which displays the average FCT of the four schemes with 40% workloads. From the figure, we can see that DIAS performs better than all other schemes. Overall, DIAS outperforms DCTCP , L^2DCT and PIAS by 54.7%, 50.1% and 23.0%, respectively.

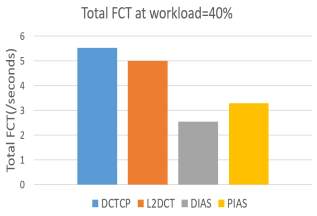

Fig. 2: Comparison with existing schemes

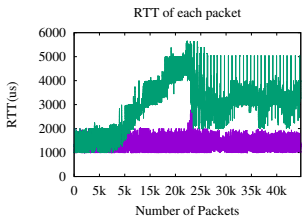

Fig. 3: RTT of each packet in a latency-sensitive long flow

B. Performance with latency-sensitive long flows

To evaluate the effect of DIAS and PIAS on the FCT of latency-sensitive long flows, we generate a latency-sensitive long flow into the network and compare the RTT of packets in the flow. In Fig.3, the green part represents RTT when applying PIAS and the purple part denotes RTT when applying DIAS, we can see that the delay of these flow packets in DIAS are much smaller than PIAS.

IV. Conclusion

Minimizing the FCT in datacenter network is a popular research. Information-aware flow scheduling schemes (e.g. D^2TCP, L^2DCT) are hard to use in practice since most applications may not know the flow sizes as a priori. As for the information-agnostic flow scheduling scheme PIAS, its scalability in large-scale networks may be poor because it relies on a central entity to collect and manage traffic load information. We propose a distributed information-agnostic flow scheduling scheme (DIAS) based on the wait-time in send buffer to both minimize the FCT of short flows and latency-sensitive long flows. Evaluations show that DIAS reduces FCT by up to 54.7% and 50.1% over DCTCP and L^2DCT, respectively. Compared with PIAS, DIAS ensures a smaller FCT of latency-sensitive long flows.

References

[1] M. Alizadeh, S. Yang, M. Sharif, S. Katti, N. McKeown, B. Prabhakar, and S. Shenker, "pfabric: Minimal near-optimal datacenter transport," in *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4. ACM, 2013, pp. 435–446.

[2] C.-Y. Hong, M. Caesar, and P. Godfrey, "Finishing flows quickly with preemptive scheduling," in *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*. ACM, 2012, pp. 127–138.

[3] A. Munir, G. Baig, S. M. Irteza, I. A. Qazi, A. X. Liu, and F. R. Dogar, "Friends, not foes: synthesizing existing transport strategies for data center networks," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4, pp. 491–502, 2015.

[4] B. Vamanan, J. Hasan, and T. N. Vijaykumar, "Deadline-aware datacenter tcp (d2tcp)," in *ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 2012, pp. 115–126.

[5] A. Munir, I. A. Qazi, Z. A. Uzmi, A. Mushtaq, S. N. Ismail, M. S. Iqbal, and B. Khan, "Minimizing flow completion times in data centers," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 2157–2165.

[6] W. Bai, K. Chen, H. Wang, L. Chen, D. Han, and C. Tian, "Information-agnostic flow scheduling for commodity data centers." in *NSDI*, 2015, pp. 455–468.

[7] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, "Data center tcp (dctcp)," in *ACM SIGCOMM computer communication review*, vol. 40, no. 4. ACM, 2010, pp. 63–74.