Testing the Deadline-Aware Queue Transport Scheme for Data Center Flows

Roberto Rojas-Cessa
Department of Electrical
and Computer Engineering
New Jersey Institute of Technology
rojas@njit.edu

Abstract

In this document, we summarize our proposed approach for the transport of data-center flows, called Deadline-Aware Queue (DAQ), and describe a possible approach to test DAQ on the infrastructure of the NSFCloud program. Our proposed testing environment requires modifications on the protocols for the transport of flows to accommodate monitoring tools, and flexibility in the modification of queue management at the switches of the data center network.

I. Introduction

We have recently proposed the Deadline-Aware Queue (DAQ) scheduling scheme for addressing the needs of transport of flows in a data center [1]. Quickly responding to the users' requests is essential for the operation of a data center. The increasing number of data center applications relies on interactive soft real-time workloads that generate a large number of small requests. These requests generate partial responses that are collected and aggregated as a digest for the issuing users, in a timely fashion. These applications demand low latency for completion of short request/respond flows. Moreover, they have a unique and stringent deadline for receiving a response from the data center. For example, a typical data center application, Online Data Intensive (OLDI), has a soft-real-time constraint (e.g., 300 ms latency) and this specific requirement is recorded on service level agreements (SLAs), directing the application operation [2].

DAQ is a transport-layer protocol that keeps high application throughput of short flows while guaranteeing sufficient throughput for long flows. DAQ also requires minimum infrastructure support and keeps no flow state at switches. DAQ satisfies the following performance requirements: (1) To meet deadlines for specific applications without recording the state of each flow, even at the times of bursty traffic in the network [3]. (2) To achieve high throughput for long background flows, independently of the load of the incoming short flows. Furthermore, the queue management is designed to assign priorities for deadline-constrained flows so as to avoid deadline inversion. This mechanism also improves the application throughput.

We studied the performance of DAQ by modeling it for NS-2 [4] simulator, where we estimated the scheme and tested different design parameters under different traffic conditions. These conditions include the so-called short and long flows. Short and long flows comprise the two major groups of traffic in DCNs [5]–[7]. Tasks going through the partition-aggregate model generate short flows. Short flows have a size of a few kilobytes and are associated with user's tasks, including server requests and responses. Long flows have a size of several megabytes and can be considered as information stored in a data center or used for maintenance. Short and long flows not only differ in size and the nature of the data, but in their service requirements. Short flows are time sensitive and long flows are throughput sensitive. In the instances where short flows are associated with deadlines for producing a response, a data center must comply with the largest number of deadlines. In the case where short flows are not associated to deadlines, the transmission of flows must be finished as soon as possible.

The time it takes to transmit a flow is referred to as flow completion time (FCT). On the other hand, long flows must be transmitted at a satisfactory throughput to maintain the data center and information up to date. The requirements of these two groups of flows must be satisfied without conflict with each other.

II. Tests in Actual Data Center

In our evaluations, we have estimated the average FCT (AFCT) and the ratio of flow completed within their deadlines among all those processed by the data center for flows associated with a deadline. This ratio is also known as application throughput.

Because our obtained results are quite satisfactory, we are interested in performing actual tests on a data center. The infrastructure of the NSCloud program seems to be a potential option for achieving this experimentation. Some of the environmental support that may be needed are reservation and assignment of guaranteed resources to perform a controlled experiment. This would involve control on the amount of resources used in one or multiple clusters (as a slice). We may require to have flexibility for medication of the transport layer on the protocol stack of servers and controlling queuing models at the switches of the data-center network. The transport layer may be achievable by modifying the kernel of the Linux operating system of the servers used as workers and aggregators.

On another hand, running experiments using existing protocols on data centers could be useful for creating traces on which we could use to run future evaluations for simulation [8]. The data centers of the NSFCloud program would be appropriate environments where to run these experiments.

References

[1] C. Ding and R. Rojas-Cessa, "Daq: Deadline-aware queue scheme for scheduling service flows in data centers," in *Communications (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2989–2994.

[2] A. Munir, I. A. Qazi, and S. B. Qaisar, "On achieving low latency in data centers," in *Proceedings of International Conference on Communications*. IEEE, 2013, p. 6.

[3] C. Wilson, H. Ballani, T. Karagiannis, and A. Rowtron, "Better never than late: Meeting deadlines in datacenter networks," *SIGCOMM-*, vol. 41, no. 4, p. 50, 2011.

[4] http://www.isi.edu/nsnam/ns/, "ns-2 network simulator," 1989.

[5] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of data center traffic: measurements & analysis," in *Proc. of SIGCOMM conference on Internet measurement*. ACM, 2009, pp. 202–208.

[6] T. Benson, A. Anand, A. Akella, and M. Zhang, "Understanding data center traffic characteristics," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 92–99, 2010.

[7] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta, "Vl2: a scalable and flexible data center network," in *Proc. ACM SIGCOMM*, vol. 39, no. 4. ACM, 2009, pp. 51–62.

[8] Z. Dong, W. Zhuang, and R. Rojas-Cessa, "Energy-aware scheduling schemes for cloud data centers on google trace data," in *the 2014 IEEE Online Conference on Green Communications*, November 2014, pp. 1–6.