Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Prof. Laurent Vanbever
Networked Systems Group

Network-Assisted Congestion Control

Semester/Master thesis proposal

As cloud computing rises, the performance of data centers becomes increasingly important. Different applications and workloads require different guarantees such as small latency or large sustained throughput [1, 4]. Congestion control algorithms play an important role in delivering this performance and have been studied extensively [2, 7, 9, 11].

Thus far, researchers have been focusing mostly on how senders and receivers should react to networking signals such as packet loss, round-trip time (RTT), throughput and ECN [5] marking. These signals, however, do not have a linear relationship with performance and thus do not trivially translate to optimal changes in the sending rate. For instance, an increased RTT might be caused by either a micro burst of traffic that temporarily filled a buffer and increased queueing delay, or by an overloaded host. These cases call for different responses: While the rate should be reduced to stop overloading a host, a short burst of traffic is over as soon as it began, and reducing the rate would only decrease performance. Among others, this ambiguity is due to the aggregation of multiple sources of congestion into one signal. Indeed, RTT is affected by delays in *all* routers, the number of hops in the path, load of the end hosts, traffic scheduling and QoS.

We argue that the signals collected by current congestion control algorithms are sub-optimal and the focus of this thesis is to understand and evaluate how purer signals, especially those which could be provided by modern networking hardware, can benefit the senders.

Other than the queuing delays per hop, which would disambiguate the RTT feedback used by multiple TCP versions [8, 10], we expect signals related to the buffer management, the traffic distribution and the queue dynamics to be also beneficial. While the queuing delay already gives an idea of the queue occupancy of each device to the sender, it is not a reliable indicator of congestion and packet loss [3] due to the existence of different management policies and buffer sizes. On the other hand, signals used by the Memory Management Unit to dynamically adapt the limits per queue such as the remaining buffer in the device or the incoming packet rate in the queue of interest are factors directly affecting future drops. For example, a sender likely causes packet loss in a device with small remaining buffer, unless it decreases its rate. On the contrary, the sender should increase the rate of a short flow traversing a device with very large remaining buffer, even if the instantaneous queue is long. Moreover, the optimal rate for a sender depends on the elasticity of other flows using the same bottleneck [6]. Indeed, different TCP versions will react differently to the same congestion signals.If the sender knows the composition of protocols it shares the buffer with they can act accordingly. For example, if most of the competing flows are inelastic, the sender should maintain its rate or even increase it upon congestion, to get its fair share of the link capacity. On the other hand, if flows are elastic, then the sender should react to congestion by decreasing its rate, as this will result in smaller queues and thus lower delay for all flows. The network could provide this information, as it has visibility of all flows. Finally, not all congestion incidents are worth rate limiting. Indeed, congestion incidents can be caused by sudden bursts of traffic, or can be long-lived and recurrent. Knowing the difference can help the sender to gauge the potential future congestion and adapt its window accordingly.

This proposal may be loosely separated into three stages: First, one needs to understand the diffident congestion control algorithms and the intuitions behind the used signals. Second, one needs to implement a simulated environment to test known and new signals with respect to how useful they are to a sender in adapting its rate. Finally, one needs to design or extend a congestion control algorithm that makes use of the additional signals.

Requirements

- Some familiarity with communication networks, in particular with the basics of congestion control.
- Skills in programming, simulation and data analysis.

Contact

- Alexander Dietmüller, adietmue@ethz.ch
- Maria Apostolaki, apmaria@ethz.ch
- Prof. Dr. Laurent Vanbever, lvanbever@ethz.ch
- Prof. Dr. Keon Jang, keonjang@gmail.com (Max Planck Institute)

References

[1] T. Benson, A. Akella, and D. A. Maltz. Network traffic characteristics of data centers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 267–280. ACM, 2010.

[2] I. Cho, K. Jang, and D. Han. Credit-scheduled delay-bounded congestion control for datacenters. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, pages 239–252. ACM, 2017.

[3] A. K. Choudhury and E. L. Hahne. Dynamic queue length thresholds for shared-memory packet switches. *IEEE/ACM Transactions on Networking*, 6(2):130–140, April 1998.

[4] J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[5] S. Floyd, D. K. K. Ramakrishnan, and D. L. Black. The Addition of Explicit Congestion Notification (ECN) to IP. RFC 3168, Sept. 2001.

[6] P. Goyal, A. Narayan, F. Cangialosi, D. Raghavan, S. Narayana, M. Alizadeh, and H. Balakrishnan. Elasticity detection: A building block for delay-sensitive congestion control. *CoRR*, abs/1802.08730, 2018.

[7] A. Kabbani, M. Alizadeh, M. Yasuda, R. Pan, and B. Prabhakar. Af-qcn: Approximate fairness with quantized congestion notification for multi-tenanted data centers. In *2010 18th IEEE Symposium on High Performance Interconnects*, pages 58–65. IEEE, 2010.

[8] D. Katabi, M. Handley, and C. Rohrs. Congestion control for high bandwidth-delay product networks. In *Proceedings of the 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '02, pages 89–102, New York, NY, USA, 2002. ACM.

[9] 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 *2013 Proceedings IEEE INFOCOM*, pages 2157–2165. IEEE, 2013.

[10] K. Winstein and H. Balakrishnan. Tcp ex machina: Computer-generated congestion control. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, pages 123–134, New York, NY, USA, 2013. ACM.

[11] H. Wu, Z. Feng, C. Guo, and Y. Zhang. Ictcp: Incast congestion control for tcp in data-center networks. *IEEE/ACM Transactions on Networking (ToN)*, 21(2):345–358, 2013.