Addressing Legacy Challenge of RTT Unfairness using Sender-Side Taxation

Kimberly Brown, Jasleen Kaur, PhD

Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

Introduction

Transmission Control Protocol (TCP) is an end-to-end transport protocol used to ensure reliable transmission of network packets. One major issue with TCP is that congestion control algorithms (CCAs) are not RTT fair, giving an unfair advantage to flows with smaller RTTs which receive ACKs quicker. CCAs are used to adjust the congestion window and regulate the sending rate of data transmitted based on network conditions like packet loss and delay. With this current implementation, it is possible for a flow to use a larger fraction of network capacity.

Cebinae [SIGCOMM '22] attempts to solve RTT unfairness by implementing in-router mechanisms that tax the sending rate of flows.

We hypothesized that sender-side taxation on hosts of network flows, rather than a tax on routers, like Cebinae, will have a greater effect on the fairness of flows on a network. We observed that sender-side taxation led to better results than Cebinae by allowing networks to maintain high total throughput and improve fairness between flows.

Methodology

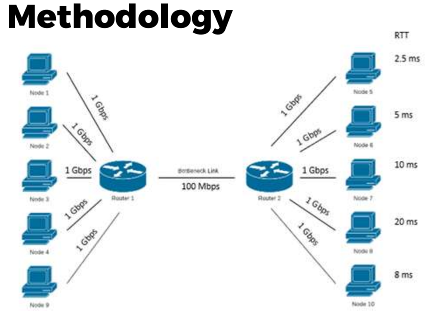

Figure 1. NS3 network simulator was used to simulate a dumbbell topology with ten nodes and two routers connected by a bottleneck link of 100 Mbps. Networks were simulated with New Reno and CUBIC CCAs

Results

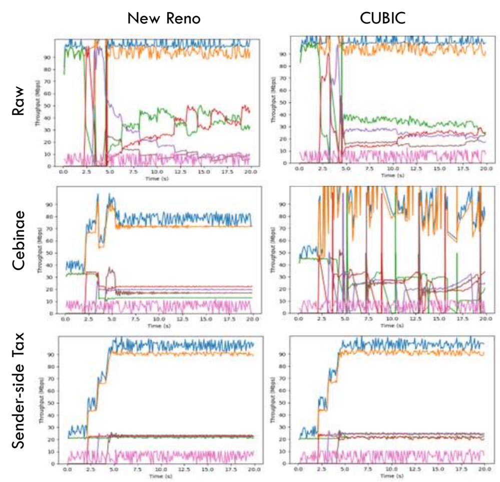

Figure 2. Time-series of throughput for a network of four flows using New Reno and CUBIC algorithms

Networks with Cebinae queuing experience a greater total throughput decrease than networks that with sender side taxation.

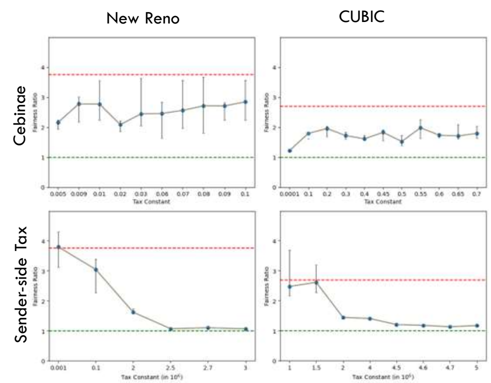

Figure 3. Median fairness ratio and tax constant relationship

Taxation of network flows leads better fairness than the default CCAs. Networks implemented with Cebinae are less fair than those with sender-side taxation and generally don't show a clear trend in fairness ratios.

	Raw New Reno	Cebinae (Tax: 0.005)	Sender-side Tax
Total Throughput (Mbps)	100.23	87.99	97.05
Fairness Ratio Time (s)	5.42	7.85	7.45
Fairness Ratio	3.85	2.16	1.08
Time to reach raw fairness ratio (s)	5.42	0.10	0.10

Table 1. Throughput, convergence time, and fairness ratio for Raw New Reno, Cebinae, and sender-side taxation of New Reno CCA

	Raw CUBIC	Cebinae (Tax: 0.5)	Sender-side Tax
Total Throughput (Mbps)	100.16	97.61	97.85
Fairness Ratio Time (s)	8.30	5.05	4.30
Fairness Ratio	2.88	1.53	1.14
Time to reach raw fairness ratio (s)	8.30	1.11	0.10

Table 2. Throughput, convergence time, and fairness ratio for Raw CUBIC, Cebinae, and sender-side taxation of CUBIC CCA

Time to reach raw fairness ratio is determined by the time required for flows to reach the Raw New Reno or CUBIC fairness ratio after all flows become active (at 4 seconds).

Convergence time of Cebinae and the sender-side tax is greater than Raw New Reno; the time to reach the Raw New Reno fairness ratio was 0.10 seconds for both Cebinae and the sender-side taxation. CUBIC algorithm with Cebinae and the sender-side tax both took less time to converge to fairness; CUBIC with sender-side taxation reached raw CUBIC fairness ratio faster than with Cebinae.

Conclusions

- Networks using the New Reno and CUBIC CCAs without taxation (default) result in flows that do not converge; networks do not demonstrate fairness
- Cebinae promotes fairness better than the default CCAs; flows on the network reach a fairness ratio earlier than in the default disciplines, but the taxation reduces the throughput of the flows
- Sender-side taxation results in flows that take less time to converge and reach fairness, compared to both default CCAs and Cebinae
- Sender-side taxation using the New Reno CCA correlates with lower fairness ratios than the CUBIC CCA
- Time to reach raw fairness ratios are similar between New Reno and CUBIC CCAs for flows experiencing sender-side taxation

Future Work

We will investigate the network fairness results of other popular congestion control algorithms, like BBR, which uses measurements of the link delivery rate, RTT, and packet loss rate to determine the congestion window. We will also examine the queue sizes of the various tax rates for each congestion control algorithm. We will explore how to incorporate congestion-detection to tax network flows.

Acknowledgements

Thank you to the Chancellor's Science Scholars program for their continual academic and financial support throughput my time at UNC.