

xDCTCP – Extended DCTCP

Introduction

Microsoft Research and Stanford University proposed a protocol called DCTCP[1] in 2010 to solve problems specific to data centers like high application latencies, queue buildup and limited buffer space of commodity switches. DCTCP achieves these goals primarily by reacting to congestion in proportion to the extent of congestion. Even though it solves the problem to an extent, there are certain parameters in the algorithm which are held constant. We plan on varying these parameters and studying the effect on DCTCP performance.

Approach

In DCTCP, switches mark ECN bit in packets when their queue length exceeds a certain threshold. The sender is made aware of these marked packets by the receiver using ECN-Echo. The controller at the sender adjusts its window size based on the fraction of packets marked (α) as per the following equation:

$$cwnd = cwnd \times (1 - \frac{\alpha}{n})$$

$$\text{where } \alpha = (1 - g) \times \alpha + g \times F$$

($n=2$, F = Fraction of packets marked in last window, g = Weight given to new samples against the past ones)

We plan on studying the performance of DCTCP on varying g and n .

Related Work

DCTCP is one of the most efficient ways of congestion control in data centers. Algorithms that provide equivalent or better feedback to congestion in a more reliable way than TCP is an active research topic. One of the experimental proposals is D2TCP[2] which varies the window size based on deadline imposed by the aggregator on the workers.

Plan of Action

We plan on determining optimal back-off factor n by varying the current window size as per:

$$cwnd = cwnd \times (1 - \frac{\alpha}{n})$$

We also plan on determining the optimal weightage to be given to new sample by varying the value of g which has been help constant in DCTCP to 1/16. Another factor that we would like to explore is prioritizing low latency short flows by using the 802.1Q priority field.

Tools: P4, Mininet

Schedule

Checkpoint 1 – Related literature survey and setup & familiarity with P4 programming in Mininet environment.

Checkpoint 2 – Implementation of original DCTCP using P4.

Final Poster Presentation – Performance analysis of xDCTCP by following the above mentioned plan of action.

References

- [1] Alizadeh, Mohammad, et al. "Data center tcp (dctcp)." ACM SIGCOMM computer communication review 41.4 (2011): 63-74.
- [2] Vamanan, Balajee, Jahangir Hasan, and T. N. Vijaykumar. "Deadline-aware datacenter tcp (d2tcp)." ACM SIGCOMM Computer Communication Review 42.4 (2012): 115-126.