بسمه تعالی

School of Computer Engineering

Advanced Computer Networks

Transport Layer and Congestion Control

Part 9

Seyed Hamed Rastegar

Fall 1401

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR

- BBR (Bottleneck Bandwidth and RTT) is a new TCP congestion control algorithm developed by researchers at Google.

- Like Vegas, BBR is delay based, which means it tries to detect buffer growth so as to avoid congestion and packet loss.

- Both BBR and Vegas use the minimum RTT and maximum RTT, as calculated over some time interval, as their main control signals.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR

- BBR also introduces new mechanisms to improve performance, including
  - Packet-pacing,
  - Bandwidth-probing
  - RTT-probing.

Advanced Schemes

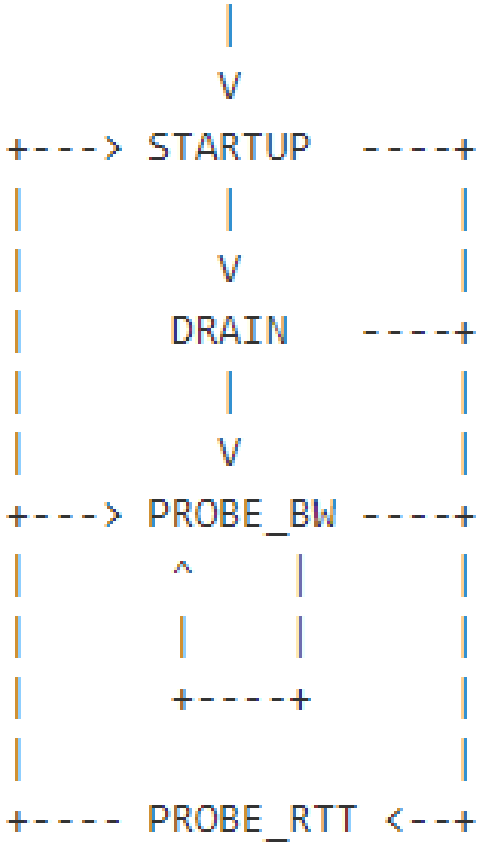
Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR

- **Packet-pacing** spaces the packets based on the estimate of the available bandwidth.
- This eliminates bursts and unnecessary queuing, which results in a better feedback signal.
- BBR also periodically increases its rate, thereby **probing the available bandwidth**.
- Similarly, BBR periodically decreases its rate, thereby **probing for a new minimum RTT**.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR >> Operation

- A description of the BBR (v1) algorithm was published in the September/October 2016 issue of ACM Queue.
- An implementation in the Linux kernel has been proposed as a patch as is according to the Figure.



Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR >> Operation

- In the **STARTUP** phase, BBR tries to quickly approximate the bottleneck bandwidth. It does so by increasing the sending rate until the estimated bottleneck bandwidth stops growing.
  - Bottleneck bandwidth is estimated from the amount of data ACKed over a given period, filtered through a "windowed max-filter".

- In the **DRAIN** phase, the sending (pacing) rate is reduced to get rid of the queue that BBR estimates to have created while probing the bottleneck bandwidth during STARTUP.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR >> Operation

- In **steady state**, BBR will pace to the estimated bottleneck bandwidth. Periodically it tries to improve its network model by doing **probing**:

- **PROBE_BW mode**: BBR probes for a bandwidth increase at the bottleneck by increasing the pacing rate, then decreasing the rate to remove temporary queuing in case the bottleneck bandwidth hasn't grown.

- **PROBE_RTT mode**: RTT is filtered through a windowed min-filter. Sometimes the algorithm will reduce the pacing rate to better approximate the base RTT in case queueing is in effect.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR

- BBR is actively being worked on and rapidly evolving.

- One major focus is **fairness**.
  - For example, some experiments show CUBIC flows get 100 × less bandwidth when competing with BBR flows, and other experiments show that unfairness among BBR flows is even possible.

- Another major focus is avoiding **high retransmission rates**, where in some cases as many as 10% of packets are retransmitted.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR

- Employed by Amazon CloudFront

- BBR works by observing how fast a network is already delivering traffic and the latency of current roundtrips. It then uses that data as input to packet-pacing heuristics that can improve performance.

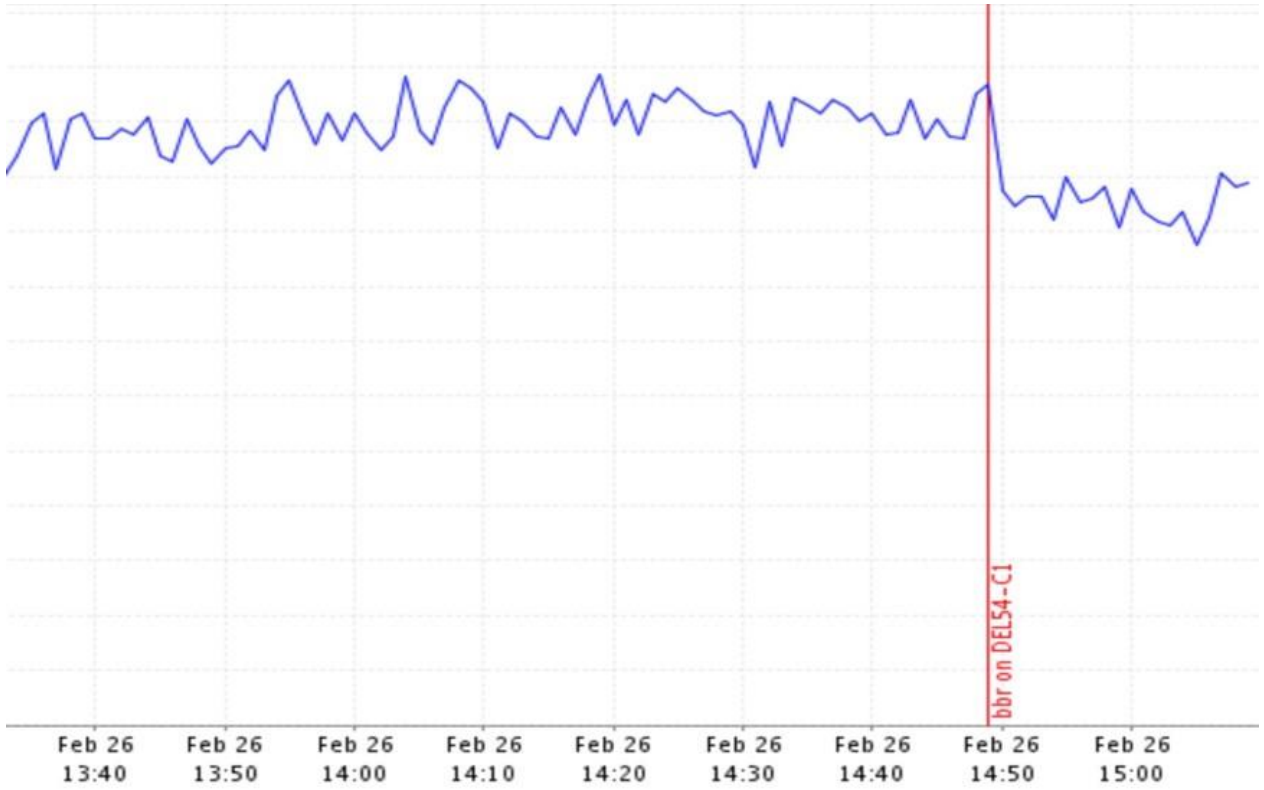
- Using BBR in CloudFront has been favorable overall, with performance gains of up to 22% improvement on aggregate throughput across several networks and regions.

- The following graph shows latency for a single POP in India before and after BBR be enabled during pre-deployment testing.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> BBR

- The graph shows the measurable improvement in latency—a 14% improvement—with BBR.



Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP

- We conclude with an example of a situation where a variant of the TCP congestion control algorithm has been designed to work **in concert** with ECN: **in cloud datacenters**.

- The combination is called **DCTCP**, which stands for **Datacenter TCP** invented by Alizadeh et al**.**

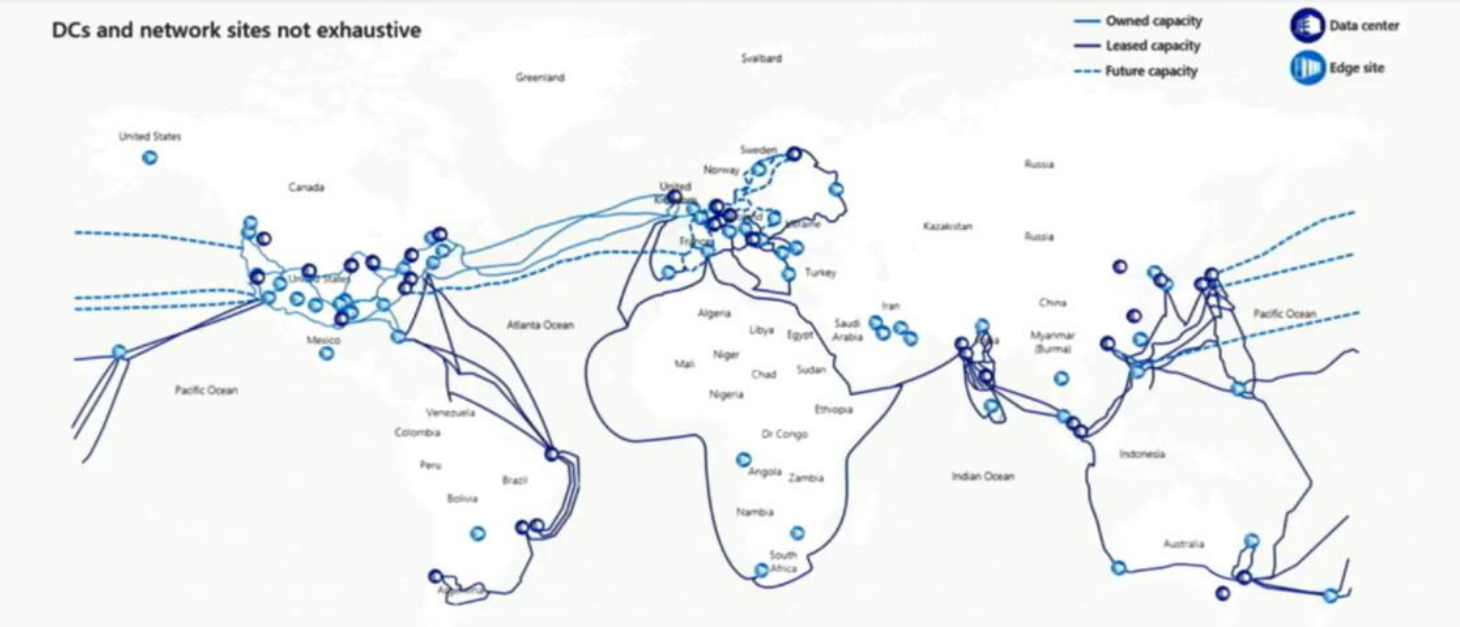
- **The situation is unique** in that a datacenter is self-contained
  - so it is possible to deploy a tailor-made version of TCP that does not need to worry about treating other TCP flows fairly.

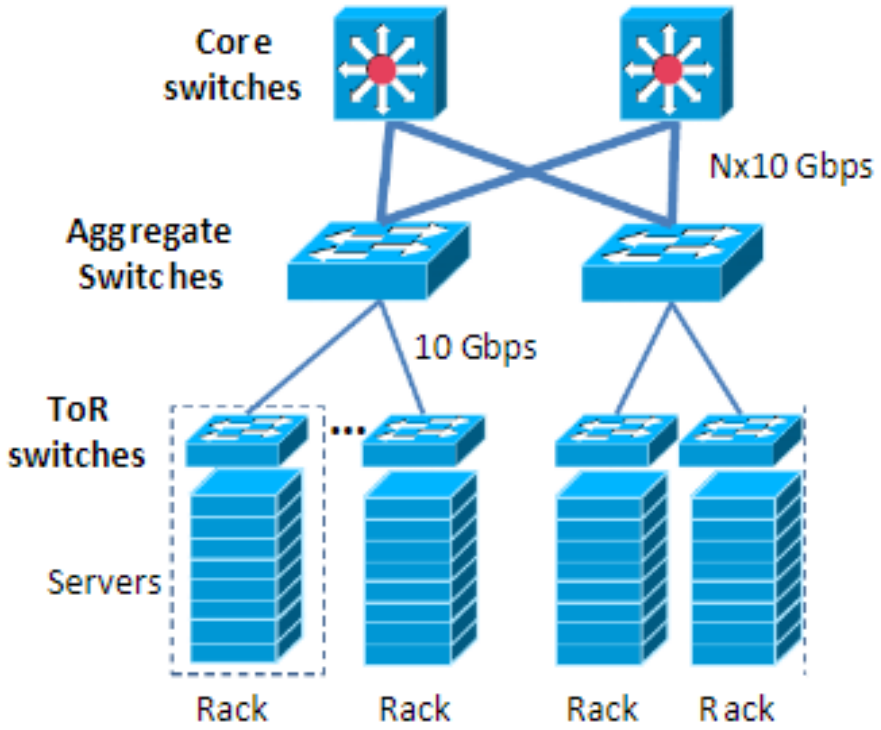
Alizadeh, Mohammad, Albert Greenberg, David A. Maltz, Jitendra Padhye, Parveen Patel, Balaji Prabhakar, Sudipta Sengupta, and Murari Sridharan. "Data center tcp (dctcp)." In *Proceedings of the ACM SIGCOMM 2010 Conference*, pp. 63-74. 2010.

Advanced Schemes

Reminder: A typical Data Center Network

- Inter Data center networks
- Intra Data Center Networks





Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP

- The idea is straightforward.
- DCTCP adapts ECN by estimating the fraction of bytes that encounter congestion rather than simply detecting that some congestion is about to occur.
- At the end hosts, DCTCP then scales the congestion window based on this estimate.
- The approach is designed to achieve high-burst tolerance, low latency, and high throughput.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP

- The key challenge DCTCP faces is to estimate the fraction of bytes encountering congestion.
- If a packet arrives and the switch sees the queue length (K) is above some threshold, e.g.,

$$K > (RTT \times C)/7,$$

where C is the link rate in packets per second, then the switch sets the CE bit in the IP header.

❑ **Note**: The complexity of RED is not required.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP

- The receiver then maintains a Boolean variable for every flow, which we denote SeenCE, and implements the following state machine in response to every received packet:
  - If the CE bit is set and SeenCE = False, set SeenCE to True and send an immediate ACK.
  - If the CE bit is not set and SeenCE = True, set SeenCE to False and send an immediate ACK.
  - Otherwise, ignore the CE bit.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP

- Finally, the sender computes the fraction of bytes that encountered congestion during the previous observation window (usually chosen to be approximately the RTT), as the ratio of the total bytes transmitted and the bytes acknowledged with the ECE flag set.

- The sender maintains an estimate of the fraction of packets that are marked, called α, which is updated once for every window of data as follows:

$$\alpha \leftarrow (1 - g)\alpha + gF$$

where 0 < g < 1, and *F* is the fraction of packets that were marked in the last window.

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP

- Features of TCP rate control such as Slow Start, additive increase during congestion avoidance, or recovery from lost packets are left unchanged.

- However, instead of cutting its window size by 2 in response to a marked ACK, DCTCP applies the following rule once every RTT:

$$W \leftarrow W + 1 \quad \text{if none of the packets are marked in a window}$$

$$W \leftarrow W\left(1 - \frac{\alpha}{2}\right) \quad \text{if one or more packets are marked per window}$$

Advanced Schemes

Source-Based Approaches (Vegas, BBR, DCTCP) >> DCTCP >> Remarks

- Due to use of ECN, DCTCP sometimes categorized as an AQM method.
- Before introducing DCTCP, RED (with ECN) was the major congestion control scheme in data centers.
- A switch that supports RED can be reconfigured to do this by setting both the low and high threshold to K and marking based on instantaneous rather than average queue length.