The ISP Column
A column on things Internet

July 2024
Geoff Huston

Bytes from IETF 120 – BBR 1,2,3

During the recent IETF meeting it was pointed out to me that we got it all wrong when we called the end-to-end transport flow control algorithms "congestion control," as this was a term with negative connotations about the network and the quality of the user experience. If we had called this function something like "performance optimisation," then maybe the focus of the work would be not on how to avoid network congestion and packet loss, as such transient behaviours are intrinsic to most forms of packet networks, but how to interpret these signals to optimise the algorithm's adaptation to the network path characteristics. On this topic of performance optimisation I'd like to dwell on one particular presentation from the ACM/IRTF Applied Networking Research Workshop held at IETF 120, "BBRv3 in the public Internet: a boon or a bane?"

Some time back, well over a couple of decades ago now, there was one dominant TCP control algorithm used within the public Internet, characterised by the generic term "AIMD" (Additive Increase, Multiplicative Decrease). Each individual TCP flow would gently increase its sending rate in a linear manner (increasing the sending rate by one segment per Round Trip Time), and this rate inflation would continue until either the sender ran out of local buffer space, or one of more of the queues within the network had filled to the point of overflow and packets were dropped. In response to such packet loss the TCP flow rate would drop as soon as it was informed of the loss, halving the sending rate, and it would then resume this gentle rate increase. As long as all concurrent TCP sessions behaved in much the same manner, then a shared network path element would be more or less equally used by all these TCP sessions.

With the increasing diversity of network element characteristics in terms of bandwidth, stability, latency and reliability this approximate uniformity of TCP behaviour has been challenged. In the myriad of alternate TCP flow control algorithms, two other general approaches have emerged into the mainstream of adoption. The first enlists the assistance of the network to inform the TCP session of the onset of network load before packet loss and the associated loss of TCP's control signal. This Explicit Congestion Notification (ECN) is a marker placed on an IP packet when the network element is experiencing internal queue formation. The TCP packet receiver mirrors this congestion signal back to the sender in its ACK stream and the sender can then respond with a flow modification comparable to detection of packet loss. This allows a sender to perform adaptive rate control without deliberately pushing the network into overload and avoiding the extended hiatus that can be caused by receiver timeouts or inefficiencies due to packet loss. This approach of enlisting the network's cooperation in early congestion signalling lies behind the work on *L4S* (Low Latency, Low Loss, Scalable throughput) which seeks to improve the user experience through more responsive TCP flow behaviours. Another approach which has gathered momentum through widespread adoption has been in Google's *BBR* (Bottleneck Bandwidth and Round-trip propagation time) flow control algorithm, which replaces packet loss with round trip delay inflation as the signal of incipient network congestion and decreases the sensitivity of the session response from a continuous control function to a behaviour modification performed only at periodic intervals.

Various measurement exercises have shown very different TCP responses to different path characteristics. BBR is not that sensitive to packet loss and can perform far better in environments where elevated intermittent bit errors rates can cause non-congestion packet loss, often seen in mobile data scenarios, and more recently in Starlink services. ECN-aware controls are better in avoiding protracted control signal loss and can provide superior overall throughput results when compared to loss-based TCP sessions drive by congestion control protocols such as Reno and CUBIC. These various approaches not only have different behaviours in response to particular network path characteristics but respond differently to each other when sharing a common network path.

The initial version of BBR (v1) was able to exert greater flow pressure on concurrent loss-based TCP sessions (Figure 1).

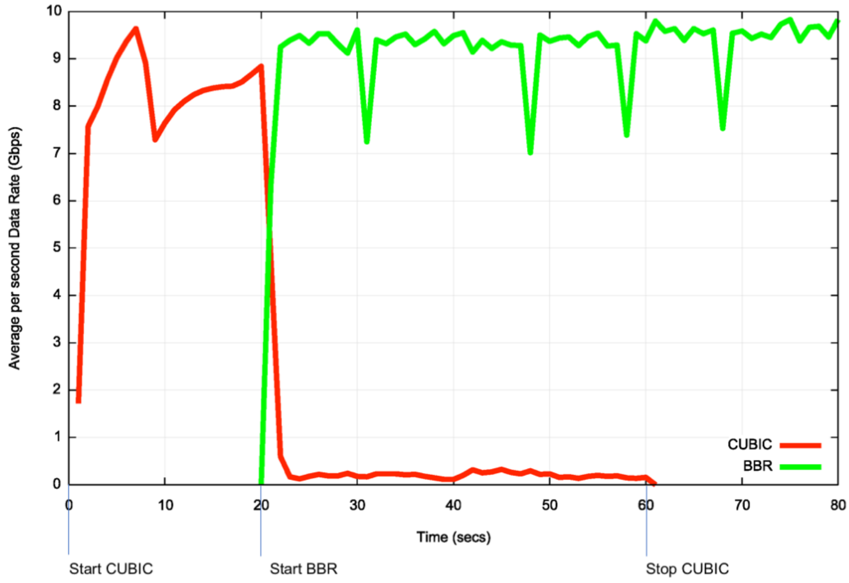

Figure 1 – BBRv1 and CUBIC session – 10Gbps link with 26ms RTT

Given this performance edge BBR was quickly adopted by many content delivery platforms. By 2019 22% of Alexa's top 20,000 web sites used BBR and BBR accounted for an estimated 40% or more of overall traffic volume on the public Internet.

BBR v2 attempted to redress some of this unfairness by reacting to both packet loss and ECN signals by moderating its sending rate in addition to BBR's probing profile. This revision to BBR also adjusted its probe response to be less aggressive in pre-empting network capacity. The result was a BBR behaviour that appeared to coexist on more or less equal terms with CUBIC (Figure 2).

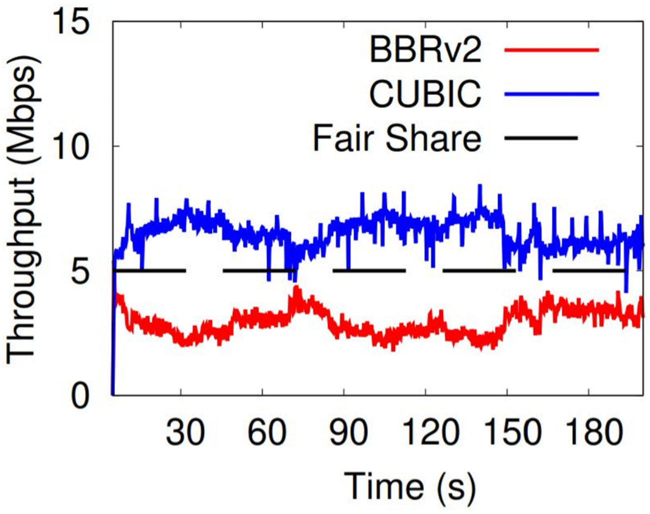

Figure 2 – BBRv2 and CUBIC session – 10Mbps link

Work has continued on BBR and in 2023 BBRv3 was released. This release is a tuning release that was intended to improve BBR's coexistence with CUBIC and RENO. The work presented at this workshop centred on a benchtop test using a bottleneck 100Mbps link with a 10ms delay and passing concurrent TCP sessions across the link. The basic result is that under such idealised conditions BBRv3 exerts a flow pressure equivalent to some 16 concurrent CUBIC sessions, which is much the same as the result for BBRv1, and when a single BBRv3 flow shares a link with a single CUBUC session the BBR session quickly occupies much of the available link capacity (Figure 3).

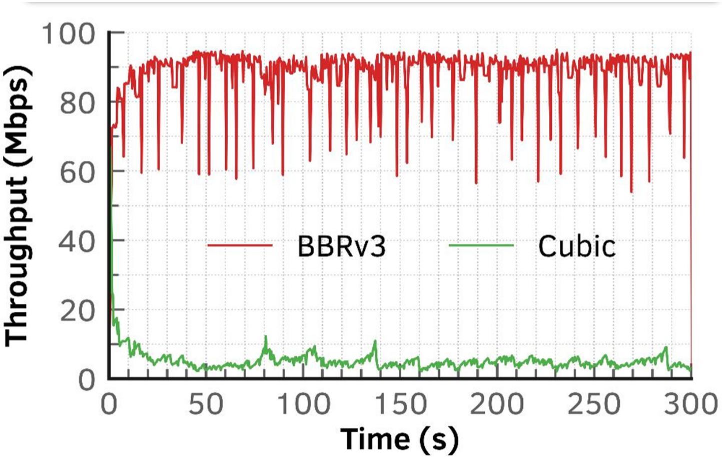

Figure 3 – BBRv3 and CUBIC session – 100Mbps link

If ECN signals are added to the network then the difference in responses of the two protocols is even greater (Figure 4).

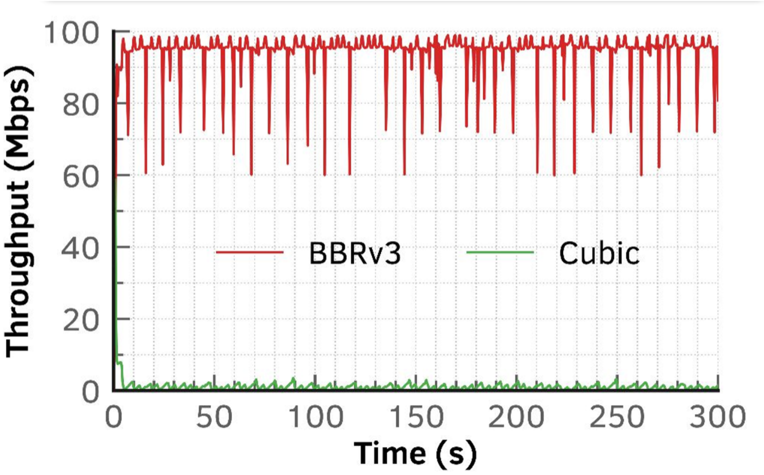

Figure 4 – BBRv3 and CUBIC session with ECN – 100Mbps link

It appears to me that our efforts in trying to optimise TCP flow behaviours by interpreting the extremes of network behaviour (packet loss) and managing the flow parameters on the basis of that interpretation, together with some assumptions about network queuing behaviours, have some fundamental limitations. The coarseness of the TCP response in loss-based algorithms assumes (and even requires) the presence of network buffers perform a form of behaviour adaptation between the host and the network.

The BBR approach is different in that it periodically provokes the network to react and interprets the network's response in relation to this provocation. This periodic probing has resulted in a control protocol that is evidently more capable of exerting continuous flow pressure on concurrent flows. It points to an observation that network queues can be driven to become a source of delay and inefficiency in terms of data transfer. The BBR model places more control into the control algorithms that operate on the end systems and try to optimise their behaviour in a manner that avoids using the network's queuing responses. A reasonable objective of any adaptive performance control protocol is to fill the network's circuits, but at the same time not to fill the network's internal buffers!

The widespread adoption of BBR on the internet points to the ongoing decline of use of loss-based congestion control protocols over the Internet, and I personally suspect that the more critical service metrics for these more recent adaptive rate control protocols are the extent to which they out-perform CUBIC and related loss-based congestion protocols, and the their ability to equilibrate their controls to fairly share the available network resource across concurrent sessions.

Disclaimer

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre.

Author

Geoff Huston AM, M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region.

www.potaroo.net