BBR bandwidth-based convergence

Aug 2018

The Google BBR team

This document describes a simple algebraic model for the dynamics of BBR's bandwidth-based convergence toward a fair share of bandwidth, when multiple BBR flows share a bottleneck.

This model applies to BBRv1, and also to BBRv2 when (a) a bottleneck is not providing loss or ECN signals, and (b) the experimental drain-to-target mechanism is disabled.

In a nutshell, the convergence happens because BBR bandwidth probing produces multiplicative increases in delivery rate that are larger for small flows than for big flows. Why is this? At an intuitive level, it's because when competing for a fixed pool of resources, it's hard for multiplicative growth for "big players" to result in proportional multiplicative gains. Whereas, by contrast, for small players it's easier to achieve larger multiplicative growth gains.

First, it may help to consider an analogy. The dynamics are a bit analogous to market share in markets for commodity goods. Consider a market with a fixed-rate set of sales transactions for a commodity good, where each seller offers an identical good at an identical price, and a buyer's selection of a seller is random, and in proportion to the shelf space occupied by a seller. We can consider how the number of goods on the shelves and the relative market share of sales evolves over multiple time intervals. If all the sellers are trying to increase their market share (sales rate) by stocking shelves with a number of items that is a multiplicative increase relative to their sales in the last time interval, the small start-up producers will find it much easier to grow their market share multiplicatively for a while, while the bigger players will end up noticing their market share is shrinking multiplicatively due to losing sales to the smaller players.

Now for an algebraic description. The following example considers an algebraic evaluatition with two flows with different initial bandwidth estimates. For simplicity we consider the bottleneck link bandwidth to be 1.0.

We model a given flow's bandwidth estimate, my_bw, and the other flow's bandwidth estimate in the system, other_bw, as two parameters. We consider the case of flows that currently fully utilize the link and initially occupy different fractions of the link bandwidth. Both flows will form their initial bandwidth estimates by measuring their current delivery rate; since they are together fully utilizing the link and cannot exceed the link bandwidth, their bandwidths will sum to 1.0.

We look at how much my_bw grows during one round of that given flow's bandwidth probing. We will see that as the flow's existing share of the bandwidth increases, its growth ratio during probing decreases.

In BBR, bandwidth probing is intentionally randomized, so that different flows will probe for bandwidth at different times. By what ratio does our flow's delivery rate go up if it probes for bandwidth while the other flow does not?

(a)
```
growth_rate = bw_during_probing / bw_during_baseline
            = (g*my_bw / (g*my_bw + other_bw)) /
              (  my_bw / (my_bw + other_bw))
```

The other flow's bandwidth is the bandwidth not allocated to me:

(b)
```
other_bw = (1 - my_bw)
```

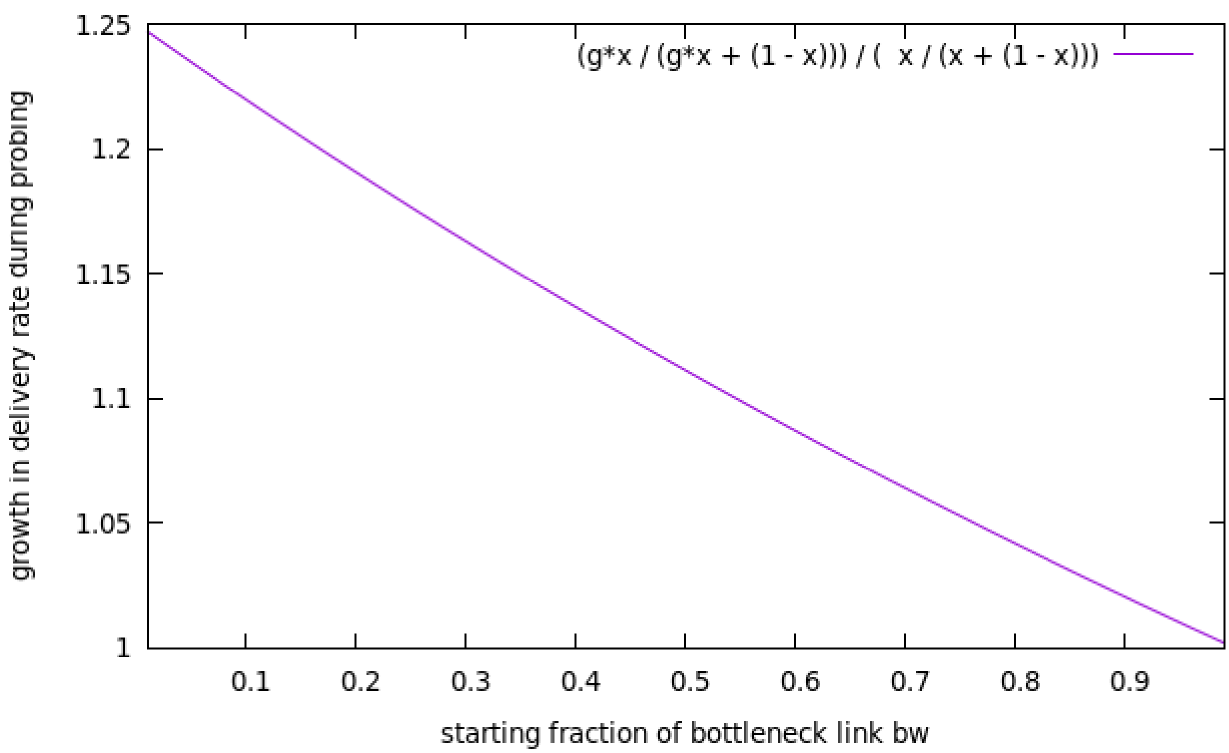
Substituting (b) into (a), this means that if we ask "by what ratio does my delivery rate go up if I probe for bandwidth and the other flow does not?" then the answer is:

(c)
```
growth_rate
= (g*my_bw / (g*my_bw + (1 - my_bw)))  /
  (  my_bw / (my_bw + (1 - my_bw)))
```

What does this look like, for g=1.25, and my_bw between .01 and 0.99? If we use gnuplot to plot this with:

```
set xrange [0.01:0.99]
g = 1.25
set xlabel 'starting fraction of bottleneck link bw'
set ylabel 'growth in delivery rate during probing'
plot  (g*x / (g*x + (1 - x))) / (x / (x + (1 - x)))
```

We get:



So the basic dynamic is that for flows with smaller existing shares of the bottleneck link bandwidth (closer to 0), multiplicative probing of a fixed magnitude results in larger proportional growth in the delivery rate, allowing smaller flows to increase their bandwidth estimates faster than large flows.

In each round of probing, a small flow is able to gradually increase its delivered share of the bottleneck bandwidth, and thereby increase its estimated bandwidth. As small flows grow their sending rate, the bandwidth remaining for large flows decreases, and in response the large flows decrease their estimated bandwidth.