Are we heading towards a BBR-dominant Internet?

Ayush Mishra, Wee Han Tiu, and Ben Leong
National University of Singapore

ABSTRACT

Since its introduction in 2016, BBR has grown in popularity rapidly and likely already accounts for more than 40% of the Internet's downstream traffic. In this paper, we investigate the following question: given BBR's performance benefits and rapid adoption, *is BBR likely to completely replace CUBIC* just like how CUBIC replaced New Reno?

We present a mathematical model that allows us to estimate BBR's throughput to within a 5% error when competing with CUBIC flows. Using this model, we show that even though BBR currently has a throughput advantage over CUBIC, this advantage will be diminished as the proportion of BBR flows increases.

Therefore, if throughput is a key consideration, it is likely that the Internet will reach a stable mixed distribution of CUBIC and BBR flows. This mixed distribution will be a *Nash Equilibrium* where none of the flows will have the performance incentive to switch between CUBIC and BBR. Our methodology is also applicable to other recently proposed congestion control algorithms, like BBRv2 and PCC Vivace. We make a bold prediction that BBR is unlikely to completely replace the CUBIC on the Internet in the near future.

CCS CONCEPTS

• **Networks** → **Transport protocols**; **Network performance modeling**.

KEYWORDS

Congestion Control, Measurement

ACM Reference Format:
Ayush Mishra, Wee Han Tiu, and Ben Leong. 2022. Are we heading towards a BBR-dominant Internet?. In *Proceedings of the 22nd ACM Internet Measurement Conference (IMC '22), October 25–27, 2022, Nice, France.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3517745.3561429

1 INTRODUCTION

First introduced by Google in 2016, BBR [4] offers better throughput and consistently lower delays than other loss-based TCP variants. BBR has since become widely deployed on the Internet. A recent measurement study in late 2019 estimates that BBR likely already accounts for more than 40% of the Internet's downstream traffic [20]. In this paper, we try to answer the following research question: *should we therefore expect the majority of Internet users to switch to BBR in the near future?*

This is an important question because the stability of the Internet depends on the competing flows interacting well with one another. We have not experienced a *congestion collapse* [17] for many years likely because the vast majority of flows have been well-understood AIMD/MIMD-window-based TCP flows [9]. The last major change in the Internet congestion landscape happened when CUBIC replaced New Reno [22, 31]. That transition was however relatively incremental because both CUBIC and New Reno are loss-based and `cwnd`-based. Therefore, all existing in-network solutions, policing algorithms, and AQMs already deployed on the Internet could largely remain unchanged.

On the other hand, if BBR were to replace CUBIC as the dominant congestion control algorithm for the Internet, it represents a fundamental paradigm shift. Many classic networking questions that have supposedly been settled would have to be re-evaluated. For example, it was said that router buffers ought to be sized inversely proportional to $\sqrt{N}$, where N is the number of flows [2]. Later, it was shown that even tiny buffers might suffice under certain conditions [11]. However, these rules of thumb assumed that flows were loss-based. BBR is loss-agnostic [27]. In other words, a BBR-dominant future Internet [20] could have potentially wide-ranging consequences and even fundamental issues like buffer sizing will need to be revisited [19].

The first step toward predicting the future composition of the Internet's congestion control landscape is to understand the incentive(s) for switching to BBR. Companies like Dropbox [16], YouTube [5], and Spotify [7] that have adopted BBR have cited better throughput as the most common reason for making the switch. To determine if switching to BBR would continue to yield better throughput, we need to understand how BBR flows interact with CUBIC flows. While a model was earlier proposed by Ware et al [30], we found that some of the assumptions made were not realistic and verified experimentally that their model does not make accurate predictions. To address this gap, we developed a new model that is able to accurately model BBR's performance to within 5% error for most realistic buffer sizes.

With an accurate model of competing CUBIC and BBR flows, we formulate the interactions between CUBIC and BBR flows as a *normal form game*. By doing so, we can abstract the setting in which websites chose their congestion control algorithms as a simple game in which some players (websites) try to maximize a utility (throughput) by selecting some pre-defined strategies (i.e. either running CUBIC or BBR). We can then apply standard game-theoretic analysis to determine if a Nash Equilibrium exists. A Nash Equilibrium (NE) is a strategy distribution in which none of the players stand to gain anything by switching strategies given that the strategies of all the other players remain the same. In the context of our problem, a Nash Equilibrium is a stable distribution of CUBIC and BBR flows such that none of the flows have any incentive to switch congestion control algorithms.

IMC '22, October 25–27, 2022, Nice, France

ACM ISBN 978-1-4503-9259-4/22/10.
https://doi.org/10.1145/3517745.3561429

By solving the normal form game, we show that a NE distribution of CUBIC and BBR flows will exist in most networks with realistic configurations. Our model is also able to predict these NE distributions and we verify empirically with a large number of experiments that these predictions are accurate. Our contributions in this paper can be summarized as follows:

(1) We present a new mathematical model for predicting the bandwidth shares of competing CUBIC and BBR flows. We verify our model with extensive experiments to show it is demonstrably more accurate at predicting bandwidth shares than the current state-of-art model by Ware et al. [30].
(2) With our model, we are able to predict that BBR's throughput gains over CUBIC steadily reduce as the proportion of BBR flows increases.
(3) We adapt our model to determine this Nash Equilibrium (NE) distribution for multiple flows with the same RTT and show (both mathematically and empirically) that this NE is at a mixed distribution of CUBIC and BBR flows. We also show that our analysis seems to hold for BBRv2, an upgraded version of BBR that is currently being developed at Google [6].

In summary, we present a mathematical model for understanding how CUBIC and BBR compete with each other and use its results to predict the future of the Internet's congestion control landscape. We predict that while BBR, or perhaps BBRv2, is likely to become more popular, *it is not likely that the majority of the Internet will fully switch from CUBIC to BBR if better throughput is the key consideration.* We observed that there are diminishing returns in BBR's throughput advantage over CUBIC as the proportion of BBR flows increases. As more and more flows switch to BBR on the Internet, CUBIC is likely to become more competitive until the point where there is no longer an incentive to switch between the two.

While a limitation of our model is that we assume that all flows have the same RTTs, we argue that this assumption is plausible, since a majority of today's Internet traffic is served via CDNs [24]. This means that it is quite plausible for the flows at local bottlenecks to have similar RTTs. We augment our model to predict a Nash Equilibrium distribution of CUBIC and BBR flows in a network and show that the ultimate mix of CUBIC and BBR on the Internet will mainly depend on the bottleneck buffer size and RTTs of the competing flows.

2 MODELLING INTERACTIONS BETWEEN BBR AND CUBIC

In this section, we first describe a basic model that will allow us to predict the throughput shares when a CUBIC flow competes with a BBR flow at a common bottleneck. Next, we extend this model to a setting with multiple CUBIC and BBR flows. For simplicity, we will assume that all the competing flows have the same base RTT. We note that the majority of today's Internet traffic is served via CDNs [24], so it would not be uncommon for the majority of flows at local bottlenecks to have similar RTTs. In this light, we argue that this assumption is likely applicable in many contexts.

2.1 Background

BBR Overview. BBR (**B**ottleneck **B**andwidth and **R**ound-trip propagation time) is a rate-based congestion control algorithm proposed by Google in 2016 [4]. BBR estimates its share of the bottleneck bandwidth and the minimum round-trip time (RTT) of the path to regulate the TCP send rate. While doing so, BBR maintains a cap on its in-flight data at twice the bandwidth-delay product (BDP). BBR is implemented as a state machine with the following 4 states to make periodic and sequential measurements to keep its estimates up-to-date:

(1) **Startup**: To quickly learn the bottleneck bandwidth, BBR performs an exponential search by doubling its sending round every iteration. By doing so, it is able to find bottleneck bandwidth in $O(\log_2(\text{BDP}))$ round trips. BBR transitions to the Drain phase once it detects that the pipe is full by looking for a plateau in bandwidth estimates.
(2) **Drain**: BBR drains packets it has accumulated in the queue during the aggressive Startup phase by reducing its in-flight packets to 1 BDP. Once it estimates that the queue is fully drained, but the pipe remains full, BBR enters the ProbeBW phase.
(3) **ProbeBW**: BBR spends a majority of time in the ProbeBW state, and probes for bottleneck bandwidth using a technique called gain cycling. BBR undergoes a cycle of 8 RTTs: it first sends packets at 1.25 times the maximum receive rate to probe for extra bandwidth at the bottleneck. To compensate for this aggression. it sends packets at 0.75 times for the next RTT, and for the remaining 6 RTTs, it maintains its sending rate at the maximum receive rate.
(4) **ProbeRTT**: BBR needs to empty the bottleneck buffer in order to accurately estimate the minimum RTT (RTT_{min}). BBR enters the ProbeRTT phase once every 10 seconds, reducing its in-flight packets to 4 in an attempt to drain the buffer.

CUBIC Overview. TCP CUBIC [13] is a loss-based algorithm, which means that when it encounters a packet loss, it shrinks its congestion window (cwnd) by a factor of 0.7. Otherwise, it increases cwnd using Equation (1).

$$\text{cwnd}(t) = C \times (t - K)^3 + W_{max} \tag{1}$$

where W_{max} is the window size just before the window is reduced and $K = \sqrt[3]{W_{max} \times (1 - \beta_{cubic})/C}$. CUBIC's implementation in the Linux kernel sets $C = 0.4$, $\beta_{cubic} = 0.3$. For the purposes of our model, the key aspect of a CUBIC flow is that it reduces to 0.7 times W_{max} after it experiences a packet loss.

2.2 Issues with Model by Ware et al.

To the best of our knowledge, the current best state-of-art model for the interactions between CUBIC and BBR is the model by Ware et al. [30]. Their model predicts the aggregate bandwidth of the competing BBR flows as

$$BBR_{frac} = (1 - p)\left(\frac{d - Probe_{time}}{d}\right) \tag{2}$$

where

$$p = \frac{1}{2} - \frac{1}{2X} - \frac{4N}{q} \tag{3}$$

$$Probe_{time} = \left(\frac{q}{c} + .2 + l\right)\left(\frac{d}{10}\right) \tag{4}$$

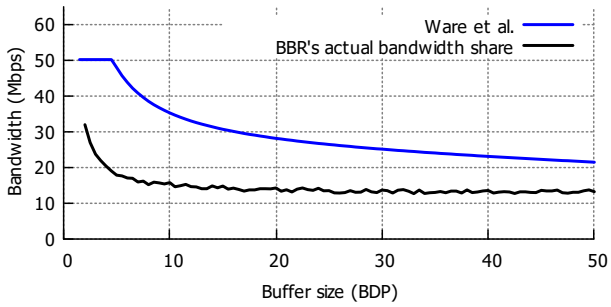


Figure 1: BBR bandwidth share for 50-Mbps bottleneck link at 40 ms RTT.

where p represents the competing CUBIC flows' aggregate fraction of the bottleneck bandwidth c. N is the number of competing BBR flows, q is the average queuing delay in the bottleneck buffer and X is the size of the bottleneck buffer in BDP. l is the base RTT of all the flows and d is the duration of the time the flows compete.

Their model predicts that BBR flows get a fixed share of the bottleneck bandwidth regardless of the number of competing CUBIC flows. While qualitatively this model does make some interesting observations (for example, Ware et al. were the first to highlight that BBR's in-flight cap is key in determining how it competes with other CUBIC flows), we have found their model to deviate significantly from actual BBR performance in experiments. We can see this from the results in Figure 1 for a simple experiment with a CUBIC flow competing with a BBR flow at a 50-Mbps bottleneck link, with each flow lasting for 2 minutes and having a base RTT of 40 ms.

With some analysis, we found that the inaccuracies in Ware et al.'s model [30] are due to the following assumptions:

(1) The most problematic assumption is that the buffer is always full. This assumption was most likely made in the interest of simplicity since even the experiments in [30] demonstrate that this assumption is not true.
(2) The second assumption is an over-simplification that BBR's RTT is the base congestion-free RTT plus $p \times q$, where p is CUBIC's share of the link capacity and q is the size of the bottleneck buffer. Since a flow's throughput is directly proportional to its average buffer occupancy, this calculation implies that CUBIC's *average* buffer occupancy is responsible for bloating BBR's RTT_{min} estimate. However, since BBR measures the *minimum* RTT during the ProbeRTT phase, it stands to reason that this bloating of the RTT_{min} should be affected by CUBIC's *minimum* buffer occupancy, and this is not the average buffer occupancy. This problem is further exacerbated when compounded with their first assumption that the buffer is always full. This effectively fixes CUBIC's buffer occupancy and uses what is in reality CUBIC's *maximum* buffer occupancy to calculate BBR's bloated RTT_{min} estimate.

We make none of these assumptions in our new model. When there are multiple CUBIC flows involved, we derive a confidence interval instead of assuming CUBIC's share of the link capacity to be fixed. This interval not only accurately predicts the actual average bandwidth of the competing flows, but also captures the stochasticity of these interactions caused by the varying degrees of synchronization between the CUBIC flows across trials and network conditions.

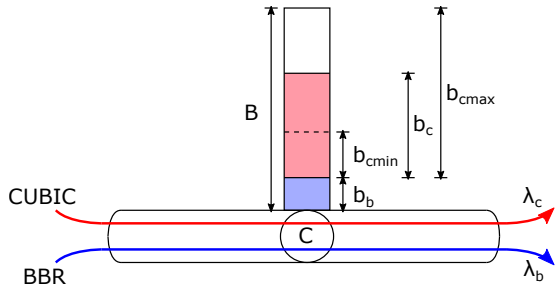


Figure 2: Network model.

2.3 Basic 2-Flow Model

In this section, we describe a simple model that can predict the bandwidth shares of two competing CUBIC and BBR flows passing through a simple drop-tail queue. In this model, both the flows have the same base/minimum RTT and compete at a bottleneck with link capacity C and buffer size B.

Assumptions. Our model makes the following assumptions:

(1) **The link is always fully utilized.** Since our analysis is centered around the bottleneck of the connection, we assume that the link is always utilized and there are always a non-zero number of packets in the buffer. To allow this assumption to hold in the presence of loss-based flows like CUBIC, we also assume the buffer is sufficiently sized [19] (at least 1 BDP) and that the CUBIC flows do not suffer any premature packet loss.
(2) **The BBR flows always maintain 2 BDP packets in flight.** This assumption is in line with the observations made by Ware et al. [30], where they showed that BBR becomes `cwnd`-bound when it competes with CUBIC. The standard implementation of BBR has a `cwnd` twice its estimated BDP. To allow this assumption to hold, we consider buffers that are at least 1 BDP in size.
(3) The packets from the two flows are uniformly distributed in the buffer.
(4) The BBR flows are mostly loss-agnostic (This is true for BBRv1 [4])
(5) **The reduction in BBR's bandwidth during the ProbeRTT phase is negligible.** We make this assumption because BBR's ProbeRTT phase lasts only for around 200 ms, which is negligible compared to its 10 s long ProbeBW phase.
(6) All flows have the same base/minimum RTT.

The notation used for our model is listed in Table 1 for convenient reference.

Modeling throughput. Consider the bottleneck illustrated in Figure 2. At any given point in time, the throughput achieved by a flow is going to be the number of bytes it has in flight divided by its round-trip time. The in-flight bytes for both the CUBIC and BBR flows will be their respective buffer shares b_b and b_c plus the amount of data they have on the wire. Since both the flows see

Table 1: Model Notation

Symbol	Meaning
C	Bottleneck link capacity
B	Bottleneck buffer size
RTT	Base RTT (propagation delay)
RTT^+	BBR's over-estimate of the RTT
b_c	CUBIC's average buffer occupancy
b_b	BBR's average buffer occupancy
Q_d	Queuing delay
b_{cmin}	CUBIC's minimum buffer occupancy
b_{cmax}	CUBIC's maximum buffer occupancy
λ_b	BBR flow's bandwidth
λ_c	CUBIC flow's bandwidth
λ_{cmin}	CUBIC's smallest bandwidth share
λ_{cmax}	CUBIC's largest bandwidth share
W_{max}	CUBIC's largest cwnd

the same bottleneck queuing delay because they share the same bottleneck, we can write their bandwidths as follows:

$$\lambda_c = \frac{\lambda_c RTT + b_c}{RTT + Q_d} \tag{5}$$

$$\lambda_b = \frac{\lambda_b RTT + b_b}{RTT + Q_d} \tag{6}$$

However, since we know that BBR is limited by its cwnd (which is capped by $2 \times$ BDP) when competing with a CUBIC flow [30], we can rewrite Equation (6) as follows:

$$\lambda_b = \frac{2\lambda_b RTT^+}{RTT + Q_d} \tag{7}$$

where RTT^+ is BBR's overestimate of the minimum RTT.

The queuing delay Q_d is the total number of bytes both the flows have in the buffer divided by the rate at which these bytes are drained (link capacity C)

$$Q_d = \frac{b_b + b_c}{C} \tag{8}$$

Relating RTT^+ to CUBIC's buffer occupancy. BBR flows see a bloated RTT because during its ProbeRTT phase, the bottleneck buffer is not completely empty and there are still some CUBIC packets that have not drained. Then, RTT^+ can be written as:

$$RTT^+ = RTT + \frac{b_{cmin}}{C} \tag{9}$$

here, b_{cmin} is the smallest number of packets a CUBIC flow has in the buffer during BBR's ProbeRTT phase. We will assume this to be CUBIC's buffer share when it backs off after a packet loss. Combining Equations (7) and (9) and simplifying, we get:

$$b_b + b_c = 2b_{cmin} + C \cdot RTT \tag{10}$$

where $b_b + b_c$ is effectively the average buffer occupancy. This is assuming the buffer size is greater than 1 BDP (or $C \times RTT$). We can use this result together with Equation (8) to rewrite Equation (5) as follows:

$$\lambda_c = \frac{\lambda_c RTT + b_c}{2RTT + \frac{2b_{cmin}}{C}} \tag{11}$$

CUBIC Minimum Buffer Occupancy. b_{cmin} is CUBIC's minimum buffer occupancy, which occurs when the CUBIC flow backs off after a packet loss. Since we know that CUBIC backs off to 0.7 times its maximum buffer occupancy after a packet loss, we can calculate b_{cmin} as follows:

$$b_{cmin} = (0.7W_{max}) - (\lambda_{cmin} RTT) \tag{12}$$

where λ_{cmin} represents the share of the bottleneck bandwidth the CUBIC flow receives during backoff, and $(\lambda_{cmin} \times RTT)$ is the number of bytes on the pipe after this backoff. Since the relevant buffer shares of BBR and CUBIC are an indicator of how much bottleneck bandwidth they are receiving at any point in time, we can write λ_{cmin} as:

$$\lambda_{cmin} = \frac{b_{cmin}}{b_{cmin} + b_b} C \tag{13}$$

To calculate b_{cmin} in Equation (12), we need to calculate the largest window size W_{max} for the competing CUBIC flow. We estimate W_{max} as follows:

$$W_{max} = b_{cmax} + \lambda_{cmax} RTT \tag{14}$$

where b_{cmax} is simply the buffer occupancy a CUBIC flow has when it completely fills the bottleneck buffer:

$$b_{cmax} = B - b_b \tag{15}$$

and λ_{cmax} is the bandwidth CUBIC gets when it puts b_{cmax} packets in the buffer:

$$\lambda_{cmax} = \frac{b_{cmax}}{b_{cmax} + b_b} C \tag{16}$$

From the results from Equations (13) to (15), we can expand Equation (12) to calculate b_{cmin} as follows:

$$b_{cmin} + \frac{b_{cmin}}{b_{cmin} + b_b} C \cdot RTT = 0.7 \times (B - b_b + \frac{B - b_b}{B} C \cdot RTT) \tag{17}$$

Putting it all together. Using the relation from Equation (10) and approximating $b_b + b_c = B$, we can write Equation (17) as:

$$\frac{B - C \cdot RTT}{2} + \frac{\frac{B - C \cdot RTT}{2}}{\frac{B - C \cdot RTT}{2} + b_b} C \cdot RTT = 0.7 \times (B - b_b + \frac{B - b_b}{B} C \cdot RTT) \tag{18}$$

Since B, C, and RTT are known quantities, we can solve Equation (18) to get a BBR flow's buffer occupancy when competing with another CUBIC flow. This b_b value can then be plugged into a simplified version of Equation (11) to solve for λ_c and λ_b:

$$\lambda_c \left(RTT + \frac{2b_{cmin}}{C} \right) = 2b_{cmin} + C \cdot RTT - B_b \tag{19}$$

$$\lambda_b = C - \lambda_c \tag{20}$$

2.4 Modelling Multiple Flows

For a network with multiple CUBIC and BBR flows with the *same* RTT, we make the following observations:

(1) First, the 2-flow model described in §2.3, only needs to take into account a CUBIC flow's maximum and minimum buffer occupancy. Hence, for a bottleneck with a total N flows with N_c of them being CUBIC flows and N_b of them being BBR flows, we can model all the CUBIC flows as one *aggregate* CUBIC flow with a combined bandwidth $\hat{\lambda}_c$.

(2) Similarly, we model all the BBR flows as another *aggregate* BBR flow with the bandwidth $\hat{\lambda}_b$. This is because we assume that the behavior of the aggregate BBR flow is practically identical to a single BBR flow when all the participating BBR flows will be cwnd bound. This is because we expect the BBR flows to be synchronized even while competing with other CUBIC flows and be fair to each other because of having similar RTTs [4].
(3) Next, while $\hat{b}_{cmax}$ remains largely unchanged (since CUBIC flows always attempt to fill the buffer, regardless of whether there is one flow or many flows), $\hat{b}_c min$ can vary for the *aggregate* CUBIC flow. This is because with multiple flows, depending on the loss pattern and start times of the competing flows, they can have varying levels of synchronization between the multiple CUBIC flows. We consider the maximum and minimum levels of synchronization separately.

In other words, to model a network with multiple CUBIC and BBR flows, we use the same model described in §2.3 but replace λ_b and λ_c with $\hat{\lambda}_b$ and $\hat{\lambda}_c$, respectively. The one key difference is that instead of using Equation (12), we consider 2 boundary cases:

(1) **CUBIC Synchronized**. If all the CUBIC flows are synchronized, the lower bound for $\hat{b}_c min$ would be given by:

$$\hat{b}_{cmin} = (0.7\hat{W}_{max}) - (\hat{\lambda}_{cmin} RTT) \tag{21}$$

(2) **CUBIC De-Synchronized**. On the other hand, if only one of N_c CUBIC flows back-off at any time, i.e. all the flows are perfectly de-synchronized, the upper bound for $\hat{b}_{cmin}$ would be given by:

$$\hat{b}_{cmin} = \left(\frac{(N_c - 0.3)}{N_c}\hat{W}_{max}\right) - (\hat{\lambda}_{cmin} RTT) \tag{22}$$

Solving the model for these 2 scenarios will provide us with a good estimate for the bandwidth shares of the BBR and CUBIC flows. In practice, we find that the empirical results are generally much closer to the case where CUBIC flows are synchronized (i.e. Equation (21)). The average bandwidths of the individual flows can be obtained as follows:

$$\lambda_c = \frac{\hat{\lambda}_c}{N_c} \tag{23}$$

$$\lambda_b = \frac{\hat{\lambda}_b}{N_b} \tag{24}$$

3 MODEL VALIDATION

In this section, we validate our models in §2.3 and §2.4 using real experiments. Since all the flows in our model have the same RTTs, we normalize the buffer size to the bandwidth-delay product (BDP) in the graphs in this section to make it easy to compare across different network conditions.

3.1 Basic 2-Flow Model

We first evaluate the accuracy of our simple model that predicts the bandwidth shares of two competing CUBIC and BBR flows. To this end, we launched a CUBIC and BBR flow through a 50 Mbps bottleneck link. The buffer size was varied from 1 BDP all the way up till 30 BDP in steps of 0.5 BDP. We repeated the same experiment with a 100 Mbps bottleneck link.

In Figure 3, we plot the observed throughput share of the BBR flow against buffer size and compared it to the values predicted by Ware et al. [30]. Over a large range of buffer sizes, our model can predict the throughput achieved by a BBR flow competing with a CUBIC flow within 5% of the actual value. In contrast, Ware et al.'s model has an error of at least 30% error, and this is for shallower buffers. As discussed in §2.2, this is because they made assumptions that do not hold in shallow to moderately sized bottleneck buffers.

An interesting observation from Figure 3 is that both the predictions of our model as well as Ware et al.'s model are relatively stable across different link speeds and RTTs, i.e. the plots for other link speeds and RTTs have a similar shape and error.

3.2 Multiple Flows

To evaluate the accuracy of our model for multiple flows, we launched 10 flows (5 CUBIC flows vs. 5 BBR flows) through a 100 Mbps bottleneck link with all the flows having a base RTT of 40 ms. The buffer size was varied from 1 BDP to 30 BDP in steps of 1 BDP. We repeated the same experiments with 20 flows (10 CUBIC flows vs. 10 BBR flows). All the flows in these experiments were started simultaneously and lasted for 2 minutes.

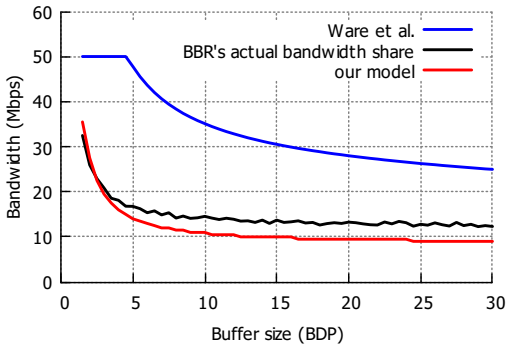
In Figure 4, we plot the per-flow average throughput against the confidence interval predicted by our multi-flow model. We see from our results that the per-flow average throughput for BBR falls within the confidence interval predicted by our model. In fact, we found the measured per-flow average throughput to be very close to the boundary where the CUBIC flows are de-synchronized. We checked the traces of our experiments and verified that the CUBIC flows were indeed generally not found to be synchronized in these experiments. It should be noted here that while it *looks* like the model by Ware et al. matches our model's 'Synch' bound in deeper buffers, it is not because their model assumes that the competing CUBIC flows are perfectly synchronized. In fact, they assume that the buffer occupancy of the competing loss-based flows does not vary at all - therefore CUBIC flows being either synchronized or not has no impact on the assumptions of their model.

3.3 Varying the Proportion of Flows

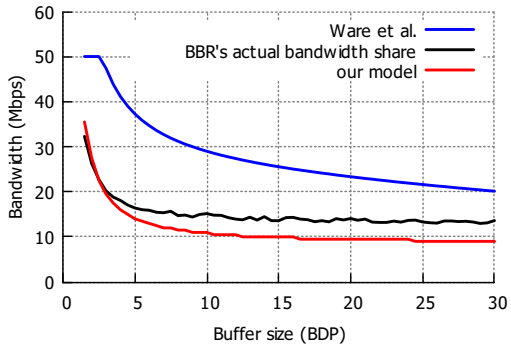
Since our goal is to understand the evolution of the Internet's congestion control landscape, it is also important to understand how BBR's average per-flow bandwidth share will change as the share of BBR flows at the bottleneck increases.

To this end, we launched 10 flows through a 100 Mbps bottleneck for buffer sizes of 3 and 10 BDP. Each of these flow were either CUBIC or BBR. Over multiple runs, we increased the share of the flows running BBR and measured the average bandwidth achieved by BBR flows over a duration of 2 minutes. All the flows were launched simultaneously. This experiment was then repeated for 20 flows.

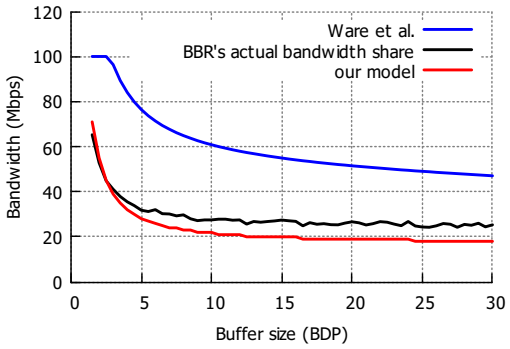
In Figure 5, we plot the average per-flow throughput against the number of BBR flows (out of 10 or 20). We see that the measured average per-flow throughputs indeed fall within the upper and lower bounds predicted by our multi-flow model. We note that in some cases, the measured values are closer to the boundary where



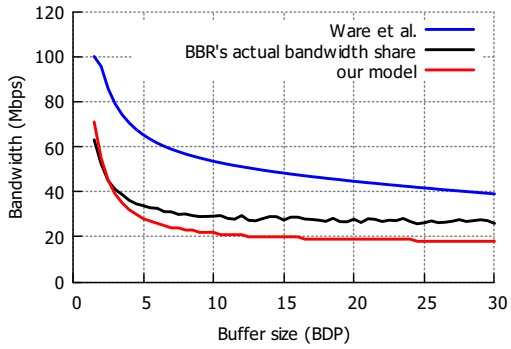
(a) 50 Mbps link, 40 ms RTT



(b) 50 Mbps link, 80 ms RTT

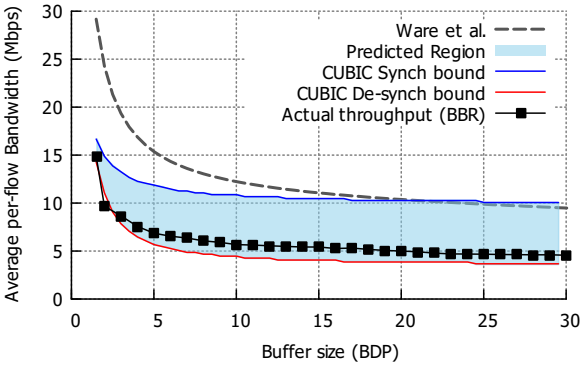


(c) 100 Mbps link, 40 ms RTT

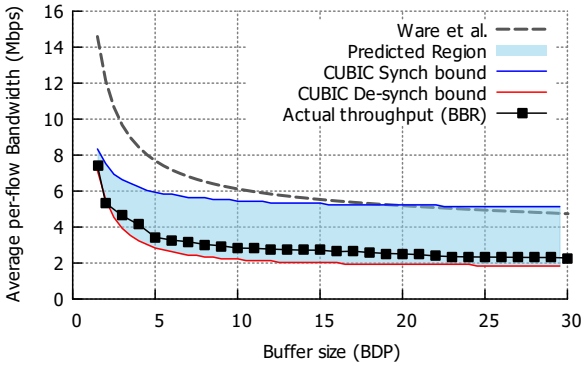


(d) 100 Mbps link, 80 ms RTT

Figure 3: Predicted throughput vs. actual throughput when a CUBIC flow competes with BBR.



(a) 5 CUBIC vs. 5 BBR, 100 Mbps link, 40 ms RTT

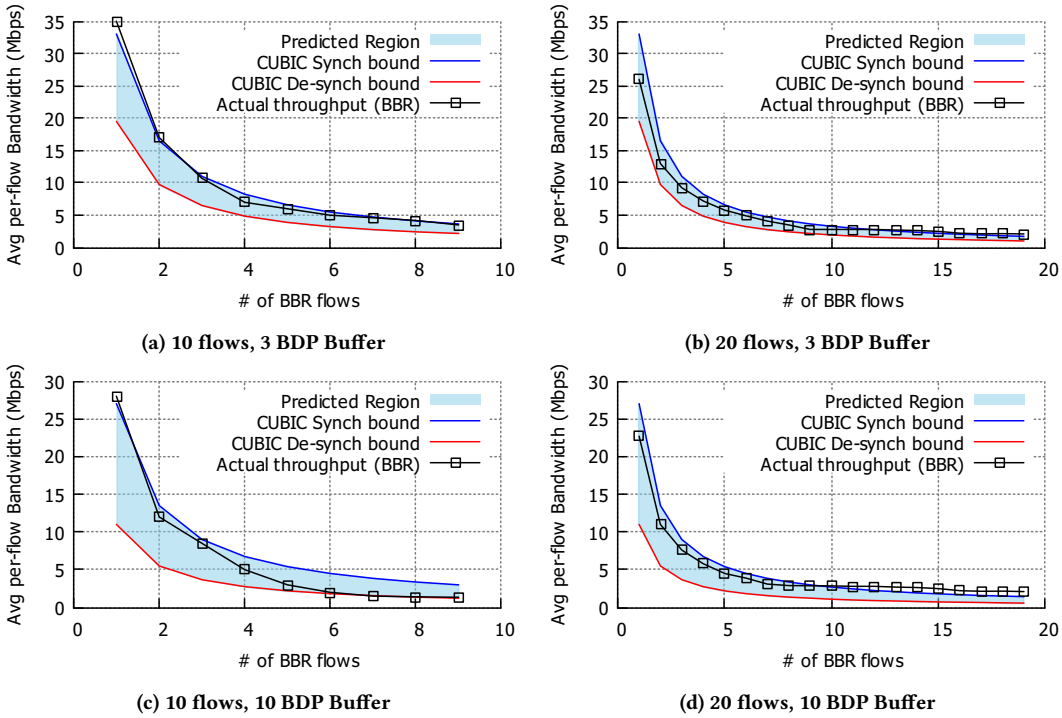


(b) 10 CUBIC vs. 10 BBR, 100 Mbps link, 40 ms RTT

Figure 4: Predicted vs. actual throughput when multiple CUBIC and BBR flows compete.

the CUBIC flows are synchronized and in other cases where they are not. Again, we checked the traces of our experiments and verified that the behavior of the CUBIC flow did correspond to the closer line in the experiments.

The most important takeaway from these experiments is that BBR's average per-flow bandwidth reduces as the proportion of flows running BBR at the bottleneck increases. This suggests that as more and more users on the Internet start using BBR as their congestion control algorithm, the throughput advantage currently enjoyed by BBR over CUBIC will be reduced. At some stage, the average throughput for BBR could fall below that of CUBIC! We will use this observation in §4 to show that a Nash Equilibrium distribution of CUBIC and BBR must exist when multiple flows with the same base RTT compete at a common bottleneck link. This trend also suggests that as more BBR flows join the bottleneck, their collective buffer occupancy increases only sub-linearly.



(a) 10 flows, 3 BDP Buffer

(b) 20 flows, 3 BDP Buffer

(c) 10 flows, 10 BDP Buffer

(d) 20 flows, 10 BDP Buffer

Figure 5: Diminishing returns for BBR.

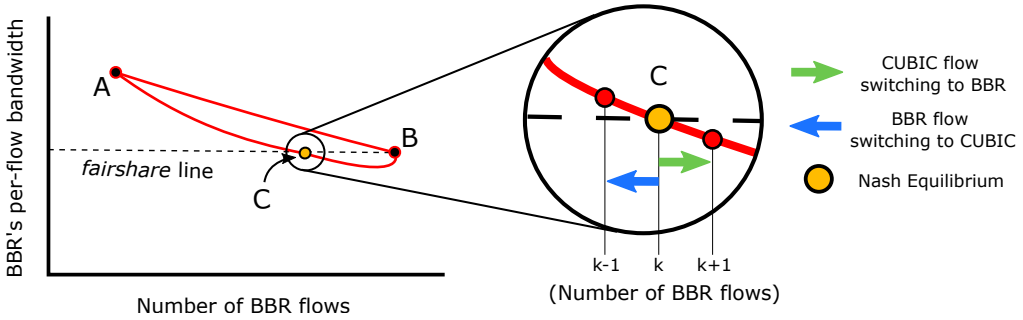


Figure 6: Nash Equilibrium for flows with similar RTTs.

4 APPLYING GAME THEORY

In this section, we apply game theory using our model in §2.4 to predict how the Internet might evolve in the near future.

Performance is multi-faceted and context-dependent. For simplicity, we focus on throughput and assume the flows will choose the congestion control algorithm that offers them the highest throughput, since better throughput is often cited as the reason for switching congestion control algorithm [5, 7, 16].

We model users (websites) as agents that currently choose either CUBIC or BBR as their congestion control (CC) algorithm. If a user can enjoy higher throughput by switching to the other CC algorithm, then there would be an incentive to switch. A Nash Equilibrium (NE) distribution of CUBIC and BBR flows occurs when none of the users have *any* incentive to switch, either because doing so will not result in increased throughput, or worse, will result in lower throughput.

We show that *a NE must exist when flows with similar RTTs compete at a bottleneck* and discuss how this analysis applies to more complicated setting with other (non-BBR) congestion control algorithms and more complex utility functions. Networks with flows with different RTTs remain future work.

4.1 NE for flows with similar RTTs

Consider a network with n flows sharing a common bottleneck, each running either CUBIC or BBR as their congestion control algorithm. In this network, we define a given distribution of CUBIC and BBR flows to be a NE, if none of the flows have any incentive to switch from CUBIC to BBR or vice versa to achieve better performance. Since all the flows have the same RTT and are essentially symmetric, there are a total of $n + 1$ possible distributions for n flows. For each of these distributions, we can measure the average bottleneck bandwidth received by all the BBR flows and plot them on a graph

as shown in Figure 6. The dotted line on this graph represents the *fair-share* line, where the average bandwidth of the BBR flows is equal to the fair-share bandwidth (i.e. link capacity divided by the total number of flows).

It has been observed that when a small number of BBR flows compete with a large number of CUBIC flows, they are able to get a disproportionately large share of the bottleneck bandwidth [30]. Given this result, we know that there exists a distribution (with a small number of BBR flows) that lies above the *fair-share* line in Figure 6, which we label as *A*. We also know that when *all* the flows at the bottleneck run BBR, they will take up all the bottleneck bandwidth. The average bandwidth for the BBR flows will then, by definition, be the fair-share bandwidth. We can use this observation to plot point B in Figure 6. We expect all the distributions between these two distributions to lie on a line connecting points A and B.

Next, we consider the distributions along the line from A to B. There are one of 2 possibilities: (i) either the line AB is always lies above the fair-share line, or (ii) the line AB intersects the fair-share line at some point C. We note that for the points above the fair-share line, BBR flows will have on average higher throughput; for the points below this line, the CUBIC flows will have on average higher throughput. What this also implies is that for the points above the fair-share line, some CUBIC flow will have an incentive to switch to BBR; the converse would be true for the points below the fair-share line.

Case 1: AB is above the fair-share line. For any point between A and B, some CUBIC flow would be incentivized to switch to BBR. As more flows switch from CUBIC to BBR, we move along the AB line until we reach B. Point B, where all flows are BBR, is then the Nash Equilibrium distribution. This is because no flows have the incentive to switch to CUBIC because it would result in a loss of throughput.

Case 2: AB intersects the fair-share line at C. We claim that C is a Nash Equilibrium distribution. To understand why we can zoom in and examine what happens at C (Figure 6). There are 2 possibilities:

(1) **A CUBIC flow can switch to BBR**. This would correspond to the current state of the network moving to the right of C. This means the average throughput of the BBR flows will drop below that of the CUBIC flows. Therefore, a CUBIC flow will never switch to BBR.
(2) **A BBR flow can switch to CUBIC**. On the other hand, a BBR flow switching to CUBIC would move the current state of the network to the left of C, which will result in the average throughput of the CUBIC flow dropping below that of the BBR flows. So again, this switch is also not tenable.

Since it is not tenable to move either left or right, point C is a stable Nash Equilibrium distribution.

Estimating the Nash Equilibrium distribution. Following the observations in §2.4, the Nash Equilibrium distribution exists when the combined bandwidth of all the BBR flows intersects with the *fair-share* line. In other words, the Nash Equilibrium distribution is the value for N_b at which:

$$\lambda_b = \frac{\hat{\lambda}_b}{N_b} = \frac{C}{N} \tag{25}$$

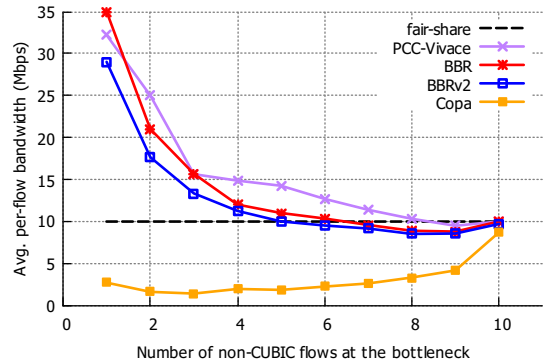


Figure 7: Combined bandwidth vs. number of flows for various congestion control algorithms.

We can use Equation (25) in conjunction with Equations (20) and (24) of our throughput model to predict the Nash Equilibrium distribution of CUBIC and BBR in any given fixed capacity network where all the flows have the same base RTT!

4.2 Other Congestion Control Algorithms

The results for CUBIC and BBR in §4.1 are based on *only* two assumptions: (i) BBR is able to obtain a disproportionately large share of the bottleneck bandwidth for at least one distribution; and (ii) when all the flows are BBR, the BBR flows will take up the available bottleneck bandwidth. The latter is self-evident if we replace BBR with another congestion control algorithm. If we can show that the former property is also true for another congestion control algorithm X, then an NE distribution of CUBIC and X flows *must* also exist when flows with similar RTTs compete at a bottleneck.

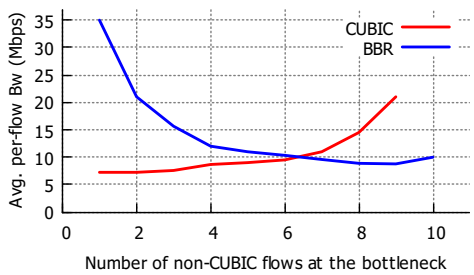
To verify if the former property holds for the following congestion control algorithms that were proposed after BBR: (i) BBRv2 [6], (ii) Copa [3], and (iii) PCC Vivace [10], we launched an experiment with 10 flows in a network with a 100 Mbps bottleneck and a 2 BDP bottleneck buffer for each algorithm X. All flows in this network ran either CUBIC or X. We measured the per-flow average throughputs for all 11 possible distributions of CUBIC and X flows.

We plot the average per-flow throughput against the number of non-CUBIC (X) flows in Figure 7. We found that PCC-Vivace [10], BBR [4] and BBRv2 [6] are able to get a disproportionately large share of the bottleneck bandwidth when there are a small number of flows. On the other hand, Copa [3] obtains lower average throughput for all congestion control algorithm distributions. Therefore, we expect a Nash Equilibrium distribution to exist for BBRv2 and PCC Vivace as well, but perhaps not for Copa.

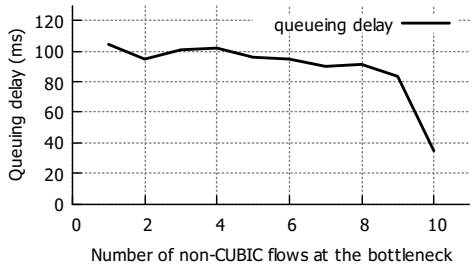
We note that our result is valid only for the case of two competing algorithms competing at a common bottleneck. Scenarios where more than two CC algorithms compete at a common bottleneck remain future work.

4.3 Complex Utility Functions

In the real world, it is likely there are senders that care not only about throughput but also delay. For video streaming, the metric of import would become even more complicated. However, we argue



(a) Average per-flow throughput.



(b) Average queuing delay (shared by CUBIC and BBR).

Figure 8: Variation in throughput and queuing delay as a function of the congestion control algorithm distribution.

that even in such scenarios, throughput will likely still be the metric that drives senders to switch between CUBIC and BBR.

To illustrate this, we plot the average throughput per algorithm and average queuing delay of a 10 flow evolution experiment discussed in §4.2. The 10 flows in this experiment pass through a bottleneck link of 100 Mbps, with a 2 BDP buffer and 40 ms base RTT. The two lines in Figure 8a represent the average throughput and CUBIC and BBR flows receive in a given trial. The single line in Figure 8b represents the average queuing delay, which is a metric shared between the flows regardless of the congestion control algorithm they run in that trial. What these graphs illustrate is that even though both throughput and delay are dependent on the congestion control algorithm distribution at the bottleneck, throughput is likely the only metric that is asymmetric enough to drive flows to switch between CUBIC and BBR. For a flow that cares about queuing delay, a switch between CUBIC and BBR likely leads to a marginal gain in utility (since it is clear from Figure 8b that increasing the proportion of BBR flows has hardly any effect on queueing delay unless all flows were BBR.)

Therefore, we conjecture that under simple utility functions that are linear combinations of throughput and delay, a Nash Equilibrium distribution will still exist. This is because we expect the decision to select between different congestion control algorithms to still be dominated by the throughput gains in such settings. That said, it is still unclear how the flows will react where all the participating flows have drastically different utility functions. Investigating whether Nash Equilibria will exist and what the distributions would look like for complex utility remains future work.

4.4 Experimental Verification

In this section, we present the results of our testbed experiments to validate the accuracy of the NE distributions predicted by our results in §4.1.

Methodology. For each network setting, we run 10 trials of all the $n+1$ possible combinations of the n senders running either CUBIC or BBR. In each trial, the senders send data for 2 minutes and we record their average per-flow throughput. To identify the NE, we enumerate all the combinations and check if there is any combination such that no individual flow in that combination can achieve higher throughput if it switches to the other TCP variant (all other flows remaining fixed). It is common for multiple distributions to satisfy this condition.

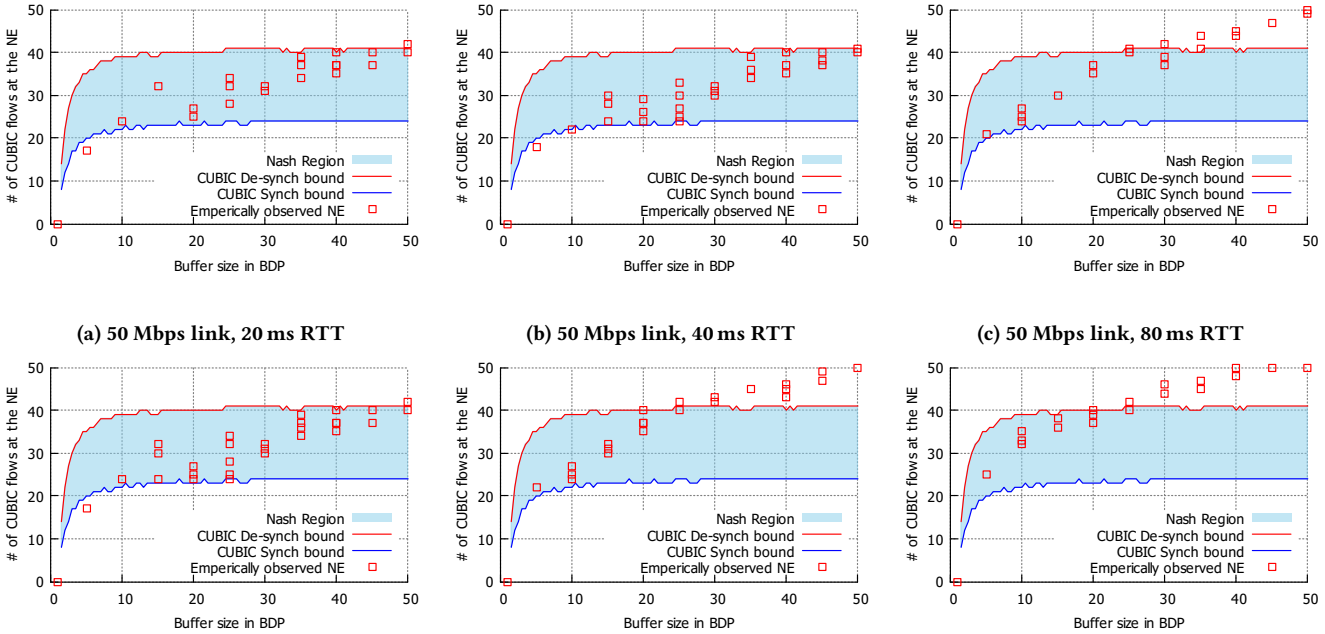
Nash Equilibria Found. We plot the results of NE found for 50 competing flows in Figure 9. The bottleneck bandwidth was set to 100 Mbps and 50 Mbps with the buffer size varying from 0.5 to 50 times the BDP. All these flows had the same base RTT which were 20, 40, and 80 ms across different trials. All the NE found empirically were in the interval predicted by our model, except those at high BDPs. BBR is not cwnd-limited in these regions and hence our model does not work well, which explains why the actual NE deviates from our predictions.

Aside from the trend that there tend to be more CUBIC flows at the NE in deeper buffers as compared to shallower buffers, the results in Figure 9 present two more interesting trends. The first is that we found *multiple* NE over different iterations of the same experiment. This is down to the throughput gains from switching between CUBIC and BBR being marginal around the Nash Equilibrium distribution. Therefore, any stochasticity across the trials can result in the NE shifting to neighboring distributions. We observe this phenomenon as multiple NE distributions across multiple trials.

The other trend is that when the buffer size is normalized by the BDP, the region predicted by the model is exactly the same regardless of the base RTT or the bottleneck link capacity. This is evident in Figure 9, where the predicted regions in all the six different network settings tested have the exact same shape. The empirically observed NE also follow this trend, with all of them roughly following the same curve across experiments with different base RTTs and link speeds. This would suggest that where the NE lies does not independently depend on either the link capacity C or the RTT, but the BDP. This makes sense, since the key indicator in what bandwidth competing BBR and CUBIC flows get in §2 is the extra packets that BBR keeps in the buffer, which depends on the BDP ($C \times RTT$). We saw similar trends when we repeated these experiments for bottlenecks with 25 flows.

4.5 Flows with different RTTs

While our model assumes that all the flows have similar RTTs, the RTT distribution on the Internet can be quite diverse [26]. Even though our model cannot easily be extended to a multi-RTT setting, we conducted experiments to investigate the NE for flows that had different RTTs. In particular, we simultaneously launched 30 flows comprising of three groups of 10 flows with RTT of 10 ms, 30 ms, and 50 ms respectively. These flows shared a 100 Mbps bottleneck link with buffer sizes varying as multiples of the BDP (bandwidth-delay product) of the flow with the shortest RTT. We ran all 2^n



(a) 50 Mbps link, 20 ms RTT (b) 50 Mbps link, 40 ms RTT (c) 50 Mbps link, 80 ms RTT

(d) 100 Mbps link, 20 ms RTT (e) 100 Mbps link, 40 ms RTT (f) 100 Mbps link, 80 ms RTT

Figure 9: Predicted Nash Equilibrium vs. observed Nash Equilibrium points for a bottleneck with 50 flows.

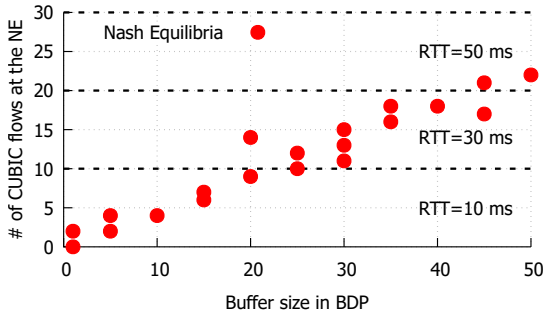


Figure 10: Nash Equilibrium distributions between CUBIC and BBR flows with different RTTs.

possible combinations of CUBIC and BBR flows for these thirty flows for three trials and then computed the Nash Equilibrium distributions just like in our previous experiments. The purpose of this experiment is not to quantitatively verify the predictions of our model, but to verify that Nash Equilibrium distributions of CUBIC and BBR can exist in multi-RTT networks as well. We plot the results in Figure 10. We noticed two key trends in NE between flows with different RTTs:

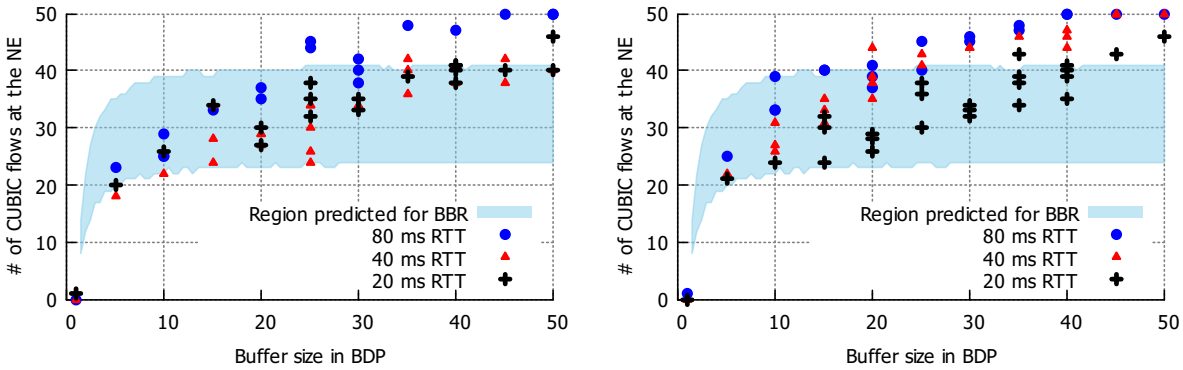
(1) **Existence of NE.** For all the buffer sizes tested, we were able to find at least one Nash Equilibrium distribution of CUBIC and BBR flows. In many instances, there were multiple NE distributions across trials, but all these distributions roughly had the same percentage of flows running CUBIC.

(2) **Nature of the NE.** In all Nash Equilibrium distributions, all the flows choosing to run CUBIC were the flows with the shortest RTTs. In other words, if a Nash Equilibrium distribution had 15 out of 30 flows running CUBIC, these 15 flows would comprise of all the ten flows with 10 ms RTT, and five 30 ms RTT flows.

Our results suggest that flows with larger RTTs benefited by switching to BBR more than flows with shorter RTTs; the reverse is true for CUBIC. This is expected from our understanding of RTT-fairness for CUBIC and BBR. Loss-based congestion control algorithms like CUBIC in general tend to favor flows with shorter RTTs [18], because flows with shorter RTTs are able to get quicker feedback and probe for bandwidth more frequently. With BBR, the opposite is true, i.e., flows with larger RTTs obtain a larger share of the bottleneck bandwidth than flows with smaller RTTs [14], because BBR flows become cwnd-limited and maintain a buffer share directly proportional to their RTT. When CUBIC and BBR flows with different RTTs compete, it is only natural that these two opposing trends would complement each other to give rise to NE distributions with shorter-RTT CUBIC flows and longer-RTT BBR flows.

4.6 BBR Predictions applied to BBRv2

Google is working to replace BBR with BBRv2 in the near future [6]. BBR has been found to be unfair to CUBIC flows in smaller buffers and can take up to half the total available bandwidth at the bottleneck regardless of flow-wise share [30]. To mitigate this issue, BBRv2 is designed to be a less aggressive alternative to BBR. At a high level, BBRv2 behaves like BBR, but because it has a variable



(a) 50 flows, 50 Mbps

(b) 50 flows, 100 Mbps

Figure 11: Nash Equilibrium distributions between competing CUBIC and BBRv2 flows.

cwnd, it is able to react to packet loss. We repeated the experiments in §4.4 for BBRv2 to determine if Nash Equilibrium distributions exist, and if so, how they would compare to our predictions for BBR.

Our results in Figure 11 suggest that multiple Nash Equilibria also exist when CUBIC and BBRv2 flows compete at a common bottleneck. This is in line with our observations in §4.2. Because BBRv2 is less aggressive than BBR, the Nash Equilibria for BBRv2 generally had a higher share of CUBIC flows for the same buffer size when compared to BBR (see Figure 7). Our results also suggest that our current model for BBR works well for BBRv2 when the RTT is relatively small. Augmenting the model to improve throughput predictions for BBRv2 remains future work.

5 DISCUSSION

Nash Equilibrium for networks with different RTTs. Nash Equilibria were earlier observed in settings where the competing CUBIC and BBR flows have different base RTTs [21]. However, one limitation of our model is that the analysis presented in this paper does not apply to networks that have flows with different RTTs. This is because with different RTTs, flows will no longer be symmetric and the state space with all the possible distributions will grow exponentially. Our proof in §4.1 requires that we linearize the state space of all the possible CCA distributions in a way that the two conditions discussed in §4.2 are met. We have not found a way to do so for a network where flows have different base RTTs. While we have results that suggest that Nash Equilibria generally exist for networks with different RTTs (see §4.5), we have not been able to extend the proof in §4.1 to this setting.

Implications on Internet Buffer Sizing. Router buffer sizing is a long-standing problem [2, 11, 19]. Rules of thumb have been derived over the years and trends have been moving towards "tiny" buffers [11], to avoid Buffer Bloat [12]. However, given that BBR keeps 2×BDP packets in flight, CUBIC flows may face starvation if BBR becomes the dominant TCP variant on the Internet.

Model Performance for large numbers of flows. While our experiments have validated our model for up to 50 concurrent flows, 50 is still orders of magnitude smaller than the number of concurrent flows passing through the bottleneck links on the Internet. It remains to be seen how our predictions will scale to the Internet. However, we see no reason why qualitatively our predictions would not apply to larger networks with hundreds of concurrent flows.

More diverse workloads and more complicated metrics. One gap in the evaluation of our model is that we have only run experiments for long flows. Real Internet workloads are more diverse, and consist of not only long flows, but also chunky video traffic, short flows generated by ad services, and latency-sensitive traffic from live streaming and video calling. Different application traffic is likely to care about more complex metrics than just throughput. It is also unlikely that the mathematical model presented in this paper (which models the steady state behavior of CUBIC and BBR) will be able to accurately replicate the interactions between short flows. Improving our model and evaluation to handle more diverse and realistic workloads remains future work.

Forced synchronization among CUBIC flows. We observed that the actual bandwidth share for the NE found is often closer to the CUBIC-Synched bound of our model (see Figure 5). This suggests that for multiple competing CUBIC and BBR flows, the CUBIC flows can get synchronized. We believe that this is likely because when all the BBR flows transition (together) from ProbeRTT to ProbeBW, they collectively add many packets to the buffer causing the buffer to overflow and the majority of the CUBIC flows to experience packet losses at the same time, and they end up synchronizing.

Poor Performance for ultra-deep buffers. We do not expect the model described in §2 to be applicable in very deep buffers (more than 100 times the BDP). This is because in such buffers, BBR is not always cwnd limited. When BBR exits its ProbeRTT phase, it starts with keeping only 1 BDP packets in flight. However, as the flow scavenges more and more bandwidth during its periodic ProbeBW cycles, it slowly grows the number of packets it has in flight till it is capped by the cwnd. Since these ProbeBW cycles happen every 8 RTTs, and in deep buffers these RTTs are much larger than in shallow buffers, BBR becomes cwnd limited at a much slower rate [30]. Therefore, our model would always overestimate how well BBR would perform against CUBIC flows in deep buffers.

To demonstrate this, we conducted a simple experiment with a single CUBIC flow competing with a single BBR flow at a 50-Mbps

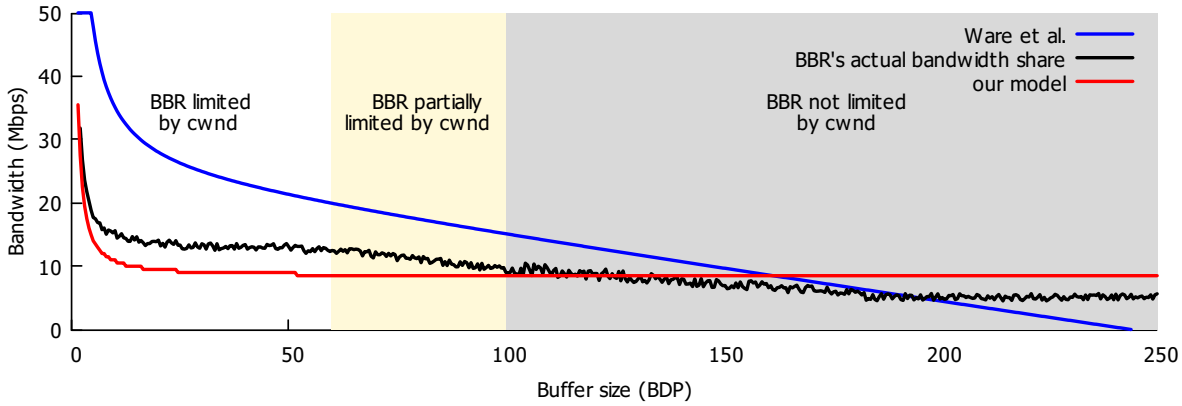


Figure 12: Performance of the model in ultra-deep (>100*BDP) buffers.

bottleneck link, with each flow lasting for 2 minutes and having a base RTT of 40 ms. We then plotted the bandwidth received by the BBR flow in buffer sizes ranging from just 1 BDP up to 250 times the BDP. We can see the extent of our model's over-estimation in ultra-deep buffers in Figure 12. It is clear that BBR's average throughput gradually decreases as the buffer size increases beyond 60 times the BDP. The actual throughput will dip below our model's predicted value when the buffers are deeper than 100 times the BDP. We verified that the BBR flow was not cwnd-limited at these buffer ranges in these experiments.

Assumption of 2 BDP packets in flight. Our model in §2 assumes that BBR flows always maintain 2 BDP worth of packets in flight. In practice, the actual number of packets would vary between 1 and 2 BDP, since each ProbeBW phase starts with approximately 1 BDP of packets. As the ProbeBW phase progresses, the overestimation of the minimum RTT will cause BBR to increase the number of packets in flight, and the rate of increase depends on the RTT. For higher RTT values, the number of bandwidth probes in the 10-second ProbeBW phase will be smaller and so the average number of packets in flight will be closer to 1 BDP. Our assumption of 2 BDP packets in flight allows us to achieve good accuracy while keeping the model simple. Nevertheless, it is likely that it is possible to improve our model by estimating the number of packets in flight during the ProbeBW phase more accurately.

Taming the Zoo. Given that the majority of Nash Equilibrium distributions that we have found are *mixed* distributions of CUBIC and BBR, it is likely that these two algorithms will have to co-exist on the Internet for the foreseeable future. We therefore need to work on networking solutions that work well with not just one class of congestion control algorithms, but a diverse mix of both of them, and probably other TCP variants [20].

Incentives to switch to better congestion control. The fact that we can see mixed distribution Nash Equilibria for most buffer sizes suggests that it might be hard to incentivize switching to *better* congestion control on the Internet while playing *nice* with existing flows. Looking back, CUBIC was clearly superior to New Reno [23] on high BDP network paths. Therefore one of the reasons CUBIC was able to largely replace New Reno was because it was more aggressive and not very friendly to existing Reno flows, so operators who cared about throughput had little option but to switch. Moving forward, the situation between BBR and CUBIC is much less straightforward. While BBR is able to achieve better throughput when there are a small number of flows, the advantage that BBR has over CUBIC diminishes as the proportion of BBR flows increases. In this light, the incentives to pick BBR over CUBIC will likely be very different from those to pick CUBIC over New Reno.

6 RELATED WORK

Since the introduction of BBR in 2016, there have been a number of works dedicated to investigating BBR [15, 25, 29]. Hock et al. conducted the first independent study into understanding how BBR interacts with CUBIC flows [15]. They observed that in shallow buffers of less than 1BDP, BBR flows take a larger share of the bandwidth compared to competing CUBIC flows. They argued that BBR's bandwidth probing causes buffer overflows and bursty losses for competing CUBIC flows. The resulting packet loss causes CUBIC to reduce its cwnd, which in turn allows BBR to take a larger share of the bandwidth. This cycle is perpetuated with every bandwidth probe, leading to CUBIC starving for bandwidth. This observation is corroborated by other studies [25, 29]. Scholz et al. conducted experiments with up to 10 BBR flows competing with 10 CUBIC flows [25] and showed that BBR flows are always able to claim at least 35% of the total bandwidth. Dong et al. also made a similar observation that when a single BBR flow competes with an ever-increasing number of CUBIC flows, BBR's fraction of the bandwidth remains the same [10].

To the best of our knowledge and as stated in §2.2, the current best state-of-art model for the interactions between CUBIC and BBR is the model by Ware et al. [30]. Ware et al. demonstrated that BBR's performance is governed by its cwnd in deep buffers. They claimed that for *very deep* buffers, BBR flows collectively take up a fixed share of the bottleneck buffer. Unfortunately, these assumptions are not very good and hence their model is not accurate over a large range of buffer sizes. One of its key shortcomings is that it assumes that bottleneck buffers are always full. Our model makes none of these assumptions.

Our model builds upon our earlier work [21], where we showed that a NE distribution of CUBIC and BBR flows must exist in 2-flow games. We had earlier also empirically demonstrated that NE

distributions exist in networks with up to 12 flows. Our earlier experiments suggested that the share of CUBIC and BBR flows at the NE primarily depended on the buffer size. Other network parameters such as the bottleneck bandwidth and the number of flows have a marginal effect on the NE distribution. Our new model proves that the conjecture that a NE must exist between CUBIC and BBR flows is true for the case where all the flows have the same base RTT. While our experience seems to suggest that an NE must exist even for flows with different RTTs, whether this hypothesis is true remains an open question.

Game theory had previously also been applied to congestion control [1, 8, 28], albeit in different settings and contexts. Chien and Sinclair were the first to study the interactions between modified AIMD congestion control algorithms and evaluate the efficiency of the Nash Equilibrium bandwidth distributions between them [8]. They showed that the Nash Equilibrium between Reno and Tahoe flows can be efficient in drop-tail buffers and inefficient with RED-enabled buffers. The main difference between their work and ours is that their strategies (congestion control algorithms) for the players (individual flows) are fixed. Chien and Sinclair attempted to calculate the Nash Equilibrium *bandwidth distribution*, while we are focused on predicting the Nash Equilibria in terms of the distribution of congestion control algorithms. In the remaining two works [1, 28], the focus was on investigating the Nash Equilibrium bandwidth distributions between 2 flows running Reno and Vegas.

7 CONCLUSION

In this paper, we showed that BBR sees diminishing returns in its throughput advantage over CUBIC as the proportion of BBR flows increases. As BBR flows become more numerous, the average per-flow bandwidth of the BBR flows will drop. This dynamic suggests that for most realistic network scenarios, there will likely always be a Nash Equilibrium distribution of CUBIC and BBR flows, where no flows have any incentive to switch. We thus make a bold prediction that it is unlikely that BBR will completely replace the CUBIC flows on the Internet in the near future. Even as BBR (or BBRv2) continues to grow in dominance, we believe that some flows always continue to be CUBIC for some time to come.

ACKNOWLEDGEMENTS

We would like to thank the anonymous reviewers, Ali Razeen, and our shepherd, Srikanth Sundaresan, for providing their valuable feedback. This work was supported by the Singapore Ministry of Education grant T1 251RES1917.

REFERENCES

[1] Aditya Akella, Srinivasan Seshan, Richard Karp, Scott Shenker, and Christos Papadimitriou. 2002. Selfish behavior and stability of the Internet: A game-theoretic analysis of TCP. *ACM SIGCOMM CCR* 32, 4 (2002).
[2] Guido Appenzeller, Isaac Keslassy, and Nick McKeown. 2004. Sizing Router Buffers. In *Proceedings of SIGCOMM.*
[3] Venkat Arun and Hari Balakrishnan. 2018. Copa: Practical Delay-Based Congestion Control for the Internet. In *Proceedings of NSDI.*
[4] Neal Cardwell, Yuchung Cheng, C Stephen Gunn, Soheil Hassas Yeganeh, and Van Jacobson. 2017. BBR: congestion-based congestion control. *Commun. ACM* 60, 2 (2017), 58–66.
[5] Neal Cardwell, Yuchung Cheng, C. Stephen Gunn, Soheil Hassas Yeganeh, Van Jacobson, and Amin Vahdat. 2017. *TCP BBR congestion control comes to GCP – your Internet just got faster.* https://tinyurl.com/yc7bd9jk
[6] Neal Cardwell, Yuchung Cheng, Soheil Hassas Yeganeh, Ian Swett, Victor Vasiliev, Priyaranjan Jha, Yousuk Seung, Matt Mathis, and Van Jacobson. 2019. *BBR v2 A Model-based Congestion Control.* https://datatracker.ietf.org/meeting/104/materials/slides-104-iccrg-an-update-on-bbr-00
[7] Erik Carlsson and Eirini Kakogianni. 2018. *Smoother Streaming with BBR.* https://tinyurl.com/yyt5tbhd
[8] Steve Chien and Alistair Sinclair. 2007. Convergence to Approximate Nash Equilibria in Congestion Games. In *Proceedings of SODA.*
[9] Dah-Ming Chiu and Raj Jain. 1989. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN systems* 17, 1 (1989), 1–14.
[10] Mo Dong, Tong Meng, Doron Zarchy, Engin Arslan, Yossi Gilad, Brighten Godfrey, and Michael Schapira. 2018. PCC Vivace: Online-Learning Congestion Control. In *Proceedings of NSDI.*
[11] Mihaela Enachescu, Yashar Ganjali, Ashish Goel, Nick McKeown, and Tim Roughgarden. 2006. Routers with Very Small Buffers. In *Proceedings of INFOCOM.*
[12] Jim Gettys. 2011. Bufferbloat: Dark buffers in the internet. *IEEE Internet Computing* 15, 3 (2011), 96–96.
[13] Sangtae Ha, Injong Rhee, and Lisong Xu. 2008. CUBIC: a new TCP-friendly high-speed TCP variant. *ACM SIGOPS Operating Systems Review* 42, 5 (2008), 64–74.
[14] Mario Hock, Roland Bless, and Martina Zitterbart. 2017. Experimental Evaluation of BBR Congestion Control. In *Proceedings of ICNP.*
[15] Mario Hock, Roland Bless, and Martina Zitterbart. 2017. Experimental Evaluation of BBR Congestion Control. In *Proceedings of ICNP.*
[16] Alexey Ivanov. 2019. *Evaluating BBRv2 on the Dropbox Edge Network.* https://tinyurl.com/yyrs68pp
[17] Van Jacobson. 1988. Congestion avoidance and control. *SIGCOMM CCR* 18, 4 (1988), 314–329.
[18] Tomoki Kozu, Yuria Akiyama, and Saneyasu Yamaguchi. 2013. Improving RTT fairness on cubic TCP. In *2013 First International Symposium on Computing and Networking*. IEEE, 162–167.
[19] Nick McKeown, Guido Appenzeller, and Isaac Keslassy. 2019. Sizing Router Buffers (Redux). *SIGCOMM CCR* 49, 5 (2019), 69–74.
[20] Ayush Mishra, Xiangpeng Sun, Atishya Jain, Sameer Pande, Raj Joshi, and Ben Leong. 2019. The Great Internet TCP Congestion Control Census. In *Proceedings of SIGMETRICS.*
[21] Ayush Mishra, Jingzhi Zhang, Melodies Sims, Sean Ng, Raj Joshi, and Ben Leong. 2021. Conjecture: Existence of Nash Equilibria in Modern Internet Congestion Control. In *Proceedings of APNet.*
[22] Jitendra Padhye and Sally Floyd. 2001. On Inferring TCP Behavior. In *Proceedings of SIGCOMM.*
[23] Vern Paxson and Mark Allman. 2009. TCP Congestion Control. RFC 5681.
[24] Canada Sandvine Inc. Waterloo, ON. 2018. *The 2018 Global Internet Phenomena Report.* https://www.sandvine.com/phenomena
[25] D. Scholz, B. Jaeger, L. Schwaighofer, D. Raumer, F. Geyer, and G. Carle. 2018. Towards a Deeper Understanding of TCP BBR Congestion Control. In *Proceedings of IFIP.*
[26] Satadal Sengupta, Hyojoon Kim, and Jennifer Rexford. 2022. Continuous in-network round-trip time monitoring. In *Proceedings of SIGCOMM.*
[27] Bruce Spang, Serhat Arslan, and Nick McKeown. 2021. Updating the theory of buffer sizing. *Performance Evaluation* 151 (2021).
[28] Tuan Anh Trinh and Sándor Molnár. 2004. A Game-Theoretic Analysis of TCP Vegas. In *Quality of Service in the Emerging Networking Panorama.*
[29] Belma Turkovic, Fernando A. Kuipers, and Steve Uhlig. 2019. Fifty Shades of Congestion Control: A Performance and Interactions Evaluation. *arXiv preprint arXiv:1903.03852* (2019).
[30] Ranysha Ware, Matthew K. Mukerjee, Srinivasan Seshan, and Justine Sherry. 2019. Modeling BBR's Interactions with Loss-Based Congestion Control. In *Proceedings of IMC.*
[31] Peng Yang, Juan Shao, Wen Luo, Lisong Xu, Jitendra Deogun, and Ying Lu. 2011. TCP Congestion Avoidance Algorithm Identification. *IEEE/ACM Transactions on Networking* 22, 4 (2011), 1311–1324.

A ETHICS

This work does not raise any ethical issues.