Bequant

Bequant TCP Performance and Fairness, compared with Cubic, BBR and BBR2

Contents

 Confidential

1. Introduction

This document studies the performance of the Bequant TCP (v4.0.13) and compares it with three other TCP variants: Linux Cubic TCP, Google BBRv1, and BBRv2. For different access networks and buffer depth at the bottleneck, it analyzes the speed, latency and losses of stand-alone connections, competing connections of the same variant and competing connections of different variants.

This study concentrates on the performance of the Bequant TCP stack, it does not address the additional improvements obtained when running as a TCP proxy, as in the Bequant BQN product.

To provide results as realistic and possible, tests are run using real networks and devices.

2. Test Set-up

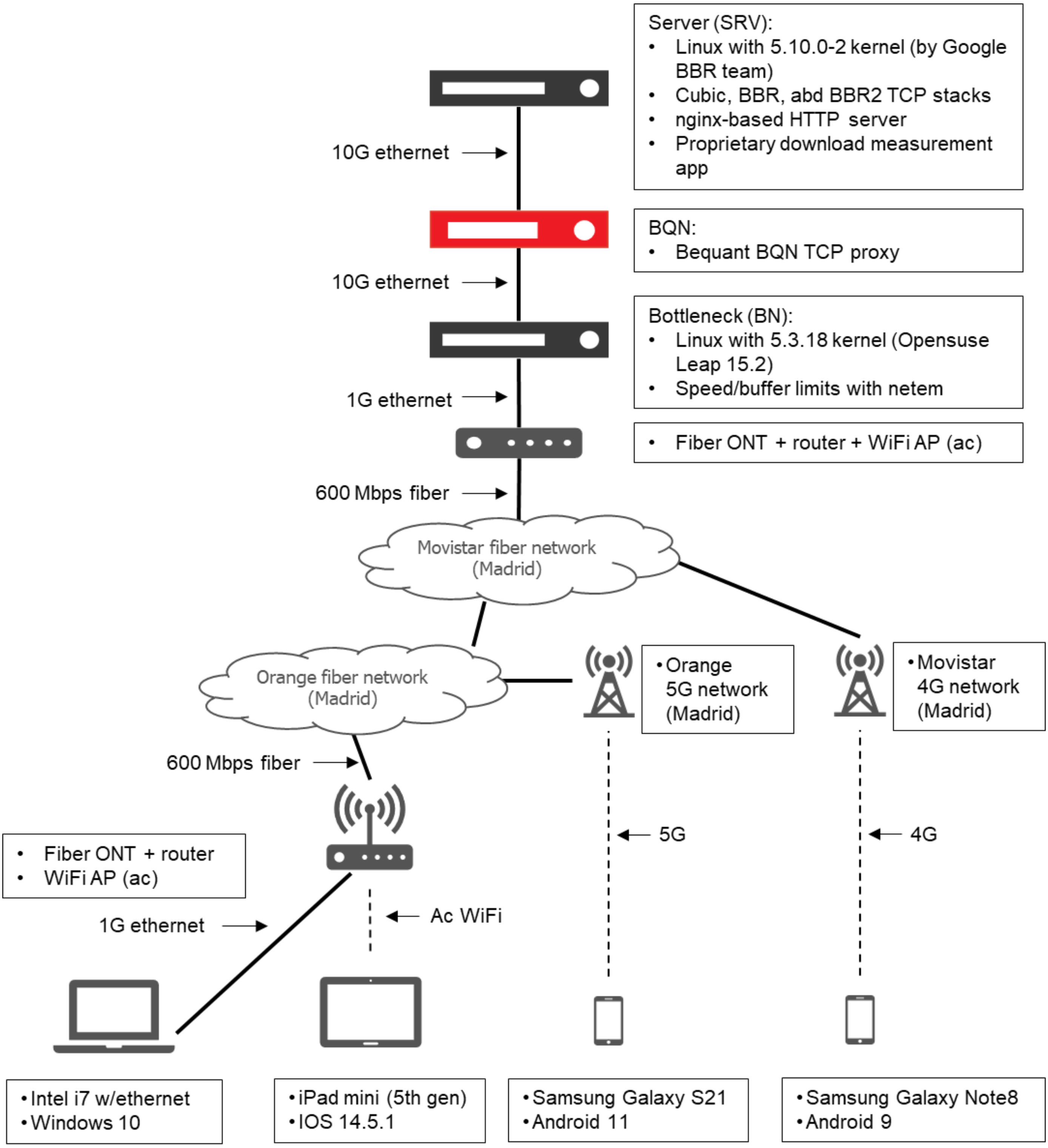

Figure 1. Test set-up

The tests consist in downloading files of known size between a web server and clients over different access networks, and comparing the speed, losses, and delays of the Bequant TCP with three of the most common TCP variants:

- Linux Cubic

- Google BBRv1
- Google BBRv2

The three TCP variants are implemented in Linux kernel 5.10.0-2[1]. This kernel has been installed in a Linux server (tagged as SRV in the Test Set-up diagram).

The SRV server contains an Nginx web server that delivers content to the client devices. The content is random binary and of the exact size needed by the tests. The Bequant TCP implementation is in a proxy server (BQN) connected right in front of the SRV server with a high-capacity network interface (10 Gbps). The BQN proxy server can forward certain TCP connections transparently, and so the TCP clients will experience a TCP connection with the TCP variant active in the SRV server (which can be Cubic, BBR or BBR2). Furthermore, the BQN proxy server can act as a TCP proxy for other connections, in which case the TCP clients will experience a TCP connection with the Bequant TCP.

Four different client devices have been used, including the three main client operating systems in use today:

- An Intel i7-7500U laptop with an Ethernet port and Windows 10 Pro (20H2)
- An iPad mini (5th generation) tablet with iPadOS 14.5.1
- A Samsung Galaxy Note8 mobile cellular phone with Android 9
- A Samsung Galaxy S21 mobile cellular phone with Android 11

Each client is used for a different access network technology:

- A 600 Mbps GPON fiber network from Orange in downtown Madrid, which is connected to a laptop via 1Gbps copper ethernet
- A WiFi (ac) link, coming out of the previous fiber router.
- A 4G network from Telefonica-Movistar in downtown Madrid
- A 5G network from Orange in downtown Madrid

2.1.1. Bottleneck Buffer Depth

To have control over the depth of the bottleneck packet buffer, an additional Linux server (BN) is used to limit the download speed with the *netem* application (https://wiki.linuxfoundation.org/networking/netem). Depending on the test requirements, there will be a deep buffer of 1000 packets (the default value in Linux servers) or a shallow buffer of 10 packets (default value in many network appliances).

[1] This is a build prepared by Google, including the BBRv2 Alpha version, downloaded from https://github.com/google/bbr/releases/tag/v2alpha-2021-02-09 on May 19, 2021.

A speed limit is introduced with the packet buffer to make sure the buffer is the bottleneck of the network path. For example, for fiber access, with or without WiFi, a limit of 500 Mbps is chosen because it is less than the access maximum speed.

For the 4G and 5G connections, the bottleneck is in the network (around 100 Mbps in the 4G network, around 250 Mbps in the 5G network), where there are very large buffers at the xNodeBs (as can be seen in the large latencies, without losses, that can build up during the connection life), so in deep-buffer tests, no speed limiting is set on our lab. For shallow-buffer tests, we limit the speed with *netem*, using a 10-packet buffer: to 50 Mbps in the case of 4G and to 100 Mbps in the case of 5G (in both cases below the maximum speed of the network).

2.2. Testing Tool

A proprietary TCP testing tool is used to download files from the SRV server to the client device. This testing tool is invoked at the client device from a regular browser (Safari in the iPad and Chrome in Windows and Android) and can control the number of iterations, the speed limitations and many other parameters of the path, as can be seen in the following screenshot.

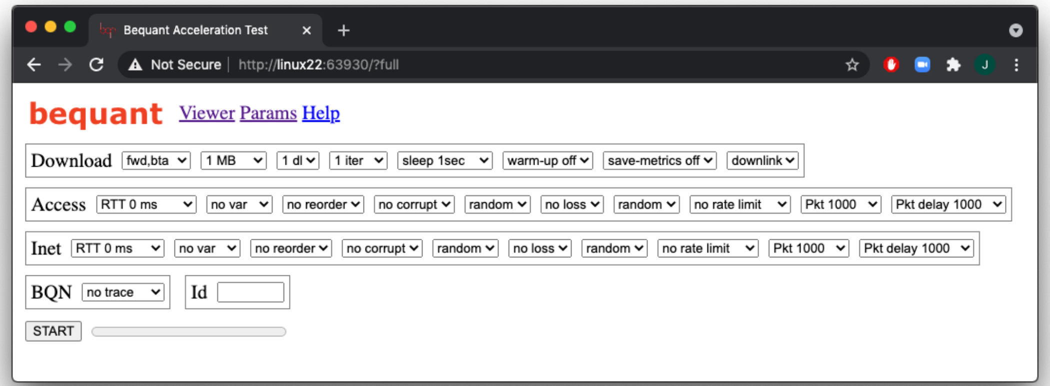

Figure 2. Input parameters for the Bequant testing tool.

We believe these tests, using regular browsers and downloads in the order of a few Mbytes, are more representative of real situations, as opposed to testing very long TCP connections (hundreds of Mbytes) in parallel, since that is a situation rarely seen in normal networks (even the heaviest downloads, such as high-definition video, is delivered as a succession of shorter downloads). In these tests, we have chosen a size of 10 Mbyte for each download.

Additionally, these tests are performed on representative public networks (fiber and 4G/5G) and real devices, which can be quite different from laboratory environments

All downloads go through the BQN node using two different ports alternatively: downloads in one of the ports go through the TCP proxy functionality, so the BQN TCP variant is used, and the downloads to the other port are forwarded transparently, so the variant to test BQN against is used.

We measure the downloads analyzing tcpdump traces captured on the BN server.

3. Test Description

3.1. Test Scenarios

Combining the devices/access technologies and the buffer depth conditions, there are eight scenarios to test:

Access	Client OS	Bottleneck		Latency (1)
		Buffer Depth (in packets)	Maximum speed	
Fiber + Ethernet	Windows 10	Deep (1000)	500 Mbps	3.5 ms
Fiber + Ethernet	Windows 10	Shallow (10)	500 Mbps	3.5 ms
Fiber + WiFi-ac	iPadOS	Deep (1000)	500 Mbps	7 ms
Fiber + WiFi-ac	iPadOS	Shallow (10)	500 Mbps	7 ms
4G	Android	Deep (in network)	No limit	23 ms
4G	Android	Shallow (10)	50 Mbps	23 ms
5G	Android	Deep (in network)	No limit	13 ms
5G	Android	Shallow (10)	100 Mbps	13 ms

(1) Latencies are measured without load (no download traffic) to reflect access network latency without the additional latency added by TCP flow control.

3.2. Tests Runs

The testing tool produces detailed data about the different test runs. The download size is always 10 MBytes. In this study, we shall concentrate on the goodput, latency and packet losses metrics, . To compare Bequant BQN TCP with each of the three variants, we run the following three tests for each of the eight scenarios described above:

- Single downloads with Bequant TCP vs the other variant, to compare the speed obtained when there are no competing flows.
- Six concurrent downloads, with all six using the same TCP variant (first, the Bequant TCP and then all six using the other TCP variant). The goal is to see each TCP variant fairness against itself (how it behaves when in congestion, competing with other flows from the same variant).
- Six concurrent downloads, with three of them using the Bequant TCP and the other three using the other TCP variant. The purpose of this test is to see how each TCP variant behaves when in congestion, competing with flows from the other TCP variant. By using the same number of concurrent flows (6) as the tests with just one TCP variant, we can compare the speed reached by a TCP

variant when the other one is absent, and therefore we can see the fairness towards each other.

3.3. Goodput Measurements

Goodput measurements consist in the total user bytes sent and acknowledged, at the TCP level, divided by the time elapsed from the sending of the first byte to the reception of the acknowledgement of the last byte. Measurements are taken from tcpdump traces at the BN server.

The mean goodput for the concurrent downloads will in fact show how even the distribution is (its level of fairness). When all the bandwidth is used and shared evenly between all the flows, the average goodput will approach the total bandwidth divided by the number of flows. However, when all the bandwidth is used and not evenly shared, the average will be a higher number, up to:

$$\frac{Total\ bandwidth}{n}\left(1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}\right)$$

where n is the number of concurrent connections

This may seem counter-intuitive, but it can be better understood with a simple example: let's take two concurrent downloads of 1 Mbyte size over a 1 Mbps link. In a completely fair situation, the two downloads will proceed at the same pace, and so the total 16 Mbits of the two files (2 files x 1 Mbyte x 8 bits per byte) will take 16 seconds to download, giving a 0.5 Mbps average download speed per dowload (8 Mbits in 16 seconds). However, in a completely unfair download, one of the files will be downloaded first (which will take 8 seconds) and the other file will be downloaded next, taking additional 8 seconds. The total time for the two downloads will still be 16 seconds, but the first file will experience a speed of 1 Mbps (the full bandwidth) while the second file will experience a speed of 0.5 Mbps. The average of both numbers is 0.75 Mbps, 1.5 times the average download speed in the completely fair set-up.

In the case of six concurrent downloads with completely unfair sharing (i.e. with every download happening consecutively), when the full bandwidth is used, the average speed will be 2.45 times the expected average speed with perfectly even sharing (total bandwidth divided by 6, in this case).

3.4. Latency Measurements

Latency is measured from the BQN node to the TCP client, with the time elapsed from sending TCP data packets to the reception of the corresponding Acknowledgement packets. The annex shows the mean value of the maximum, minimum and average latency values measured for each 10 MByte download.

3.5. Packet Loss Measurements

Packet losses are evaluated by the ratio of throughput to goodput, i.e. how many more bytes at IP level are sent, compared to the TCP level acknowledged bytes. Without packet losses there is still an overhead due to TCP headers (this minimum overhead is 2.8%). An increase above that TCP overhead will be due to packet losses being retransmitted.

4. Test Results

4.1. Fiber + Ethernet with Deep Buffer

Access	Fiber 600 Mbps + Ethernet
Client OS	Windows 10
Buffer Depth	Deep (1000 packets)
Maximum Speed	500 Mbps
Latency at Rest	3.5 ms
Download Size	10 MBytes

4.1.1. Goodput

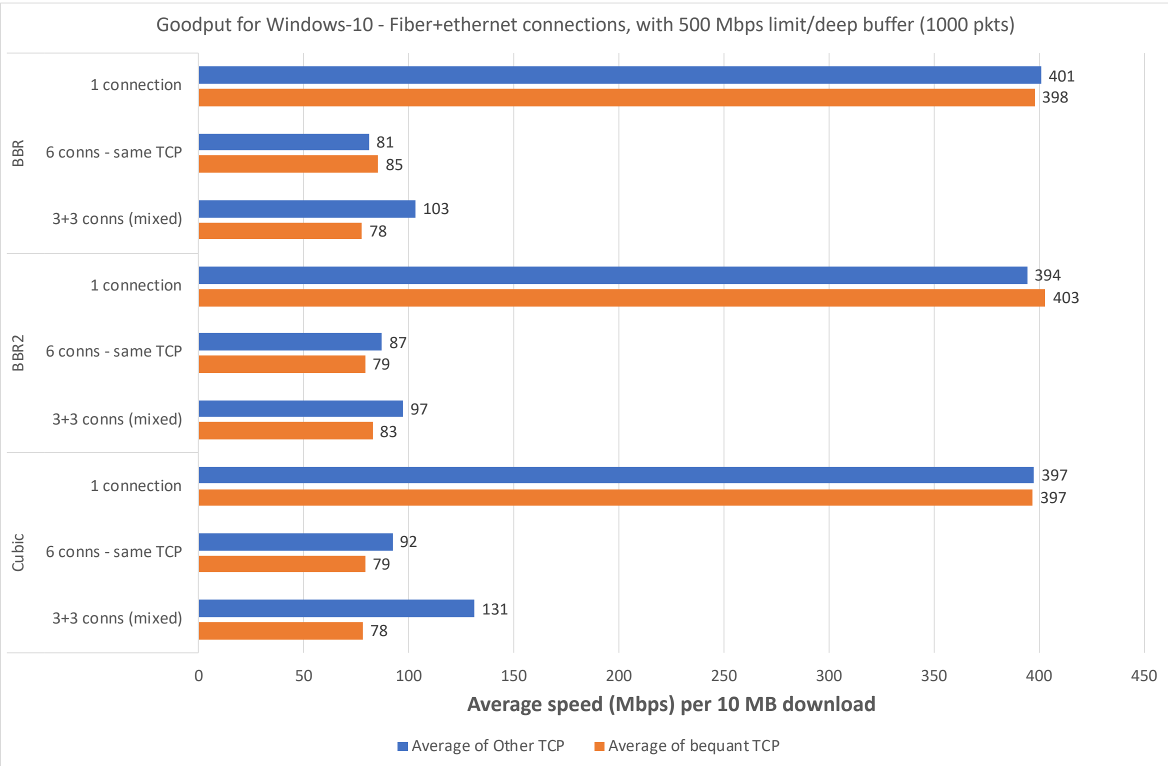

Figure 3. Goodput results for Fiber + Ethernet with deep buffer (higher is better).

Figure 3 shows the goodput results for a Windows-10 client and a 600 Mbps fiber through an Ethernet wire connection, where a 500 Mbps speed limitation is enforced with a deep buffer (1000-packet buffer). The four TCP variants can use all the available bandwidth with single 10 MB connections, reaching an average speed very close to 400 Mbps, with no significant differences, which is what can be expected with a low latency (3.5 ms).

When running six concurrent downloads from the same TCP variant the bandwidth is shared quite evenly between the 6 connections, with an average download speed from 80 to 90 Mbps. When running 6 concurrent downloads, with 3 Bequant TCP flows and 3 flows with a different TCP variant, the fairness is largely maintained, even though in all cases the Bequant TCP is less aggressive than the other variants, with Linux Cubic showing the most aggressive behavior (but still not affecting the speed of Bequant TCP flows very much).

4.1.2. Latency

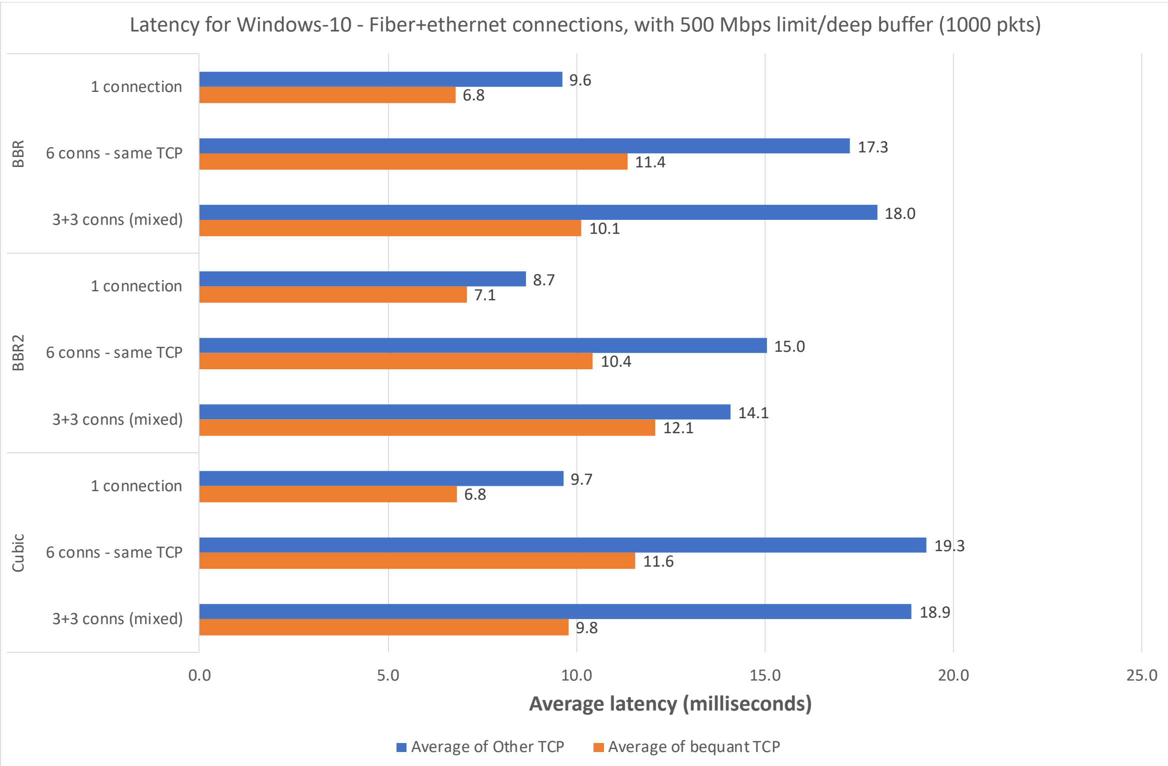

Figure 4. Latency results for Fiber + Ethernet with a deep buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 3.5 milliseconds.

Figure 4 shows the average latency measured during the 10 Mbyte downloads. In almost all the tests, the average latency experienced by the Bequant TCP connections was lower than for the other TCP variants, with values for Cubic and BBR, with shared connections, almost doubling those for the Bequant TCP.

4.1.3. Losses

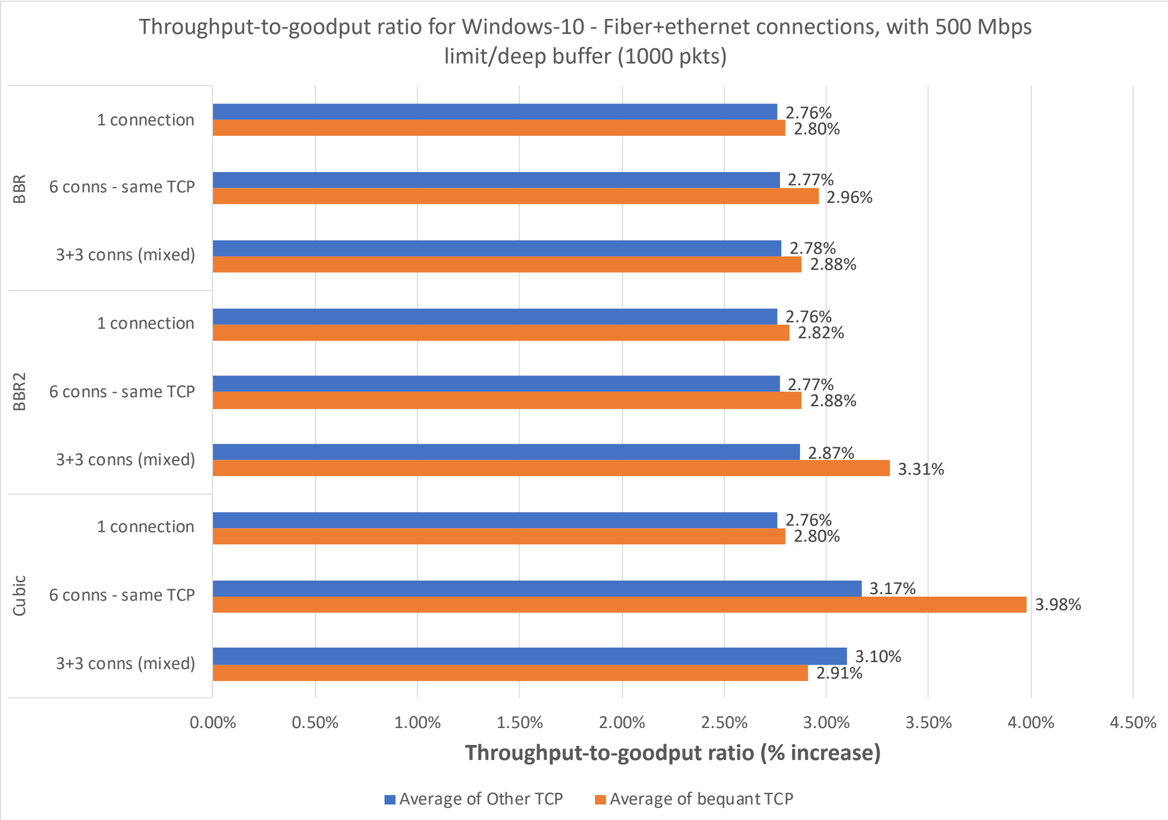

Figure 5. Throughput-to-goodput increase ratio for Fiber + Ethernet with a deep buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is close to 2.8%, due to TCP protocol overhead.

Figure 5 shows that the Bequant TCP uses slightly more IP packets to send 10 MB, mainly due to some pre-emptive retransmissions. In this deep-buffer situation, over fiber, few losses are observed: the highest amount measured are for 6 concurrent downloads with Bequant TCP, when comparing against Cubic downloads, but this can be attributed to variations in the fiber connection, since the conditions in that situation would be the same as when comparing with BBR or BBR2 downloads.

4.2. Fiber + Ethernet with Shallow Buffer

Access	Fiber 600 Mbps + Ethernet
Client OS	Windows 10
Buffer Depth	Shallow (10 packets)
Maximum Speed	500 Mbps
Latency at Rest	3.5 ms
Download Size	10 MBytes

4.2.1. Goodput

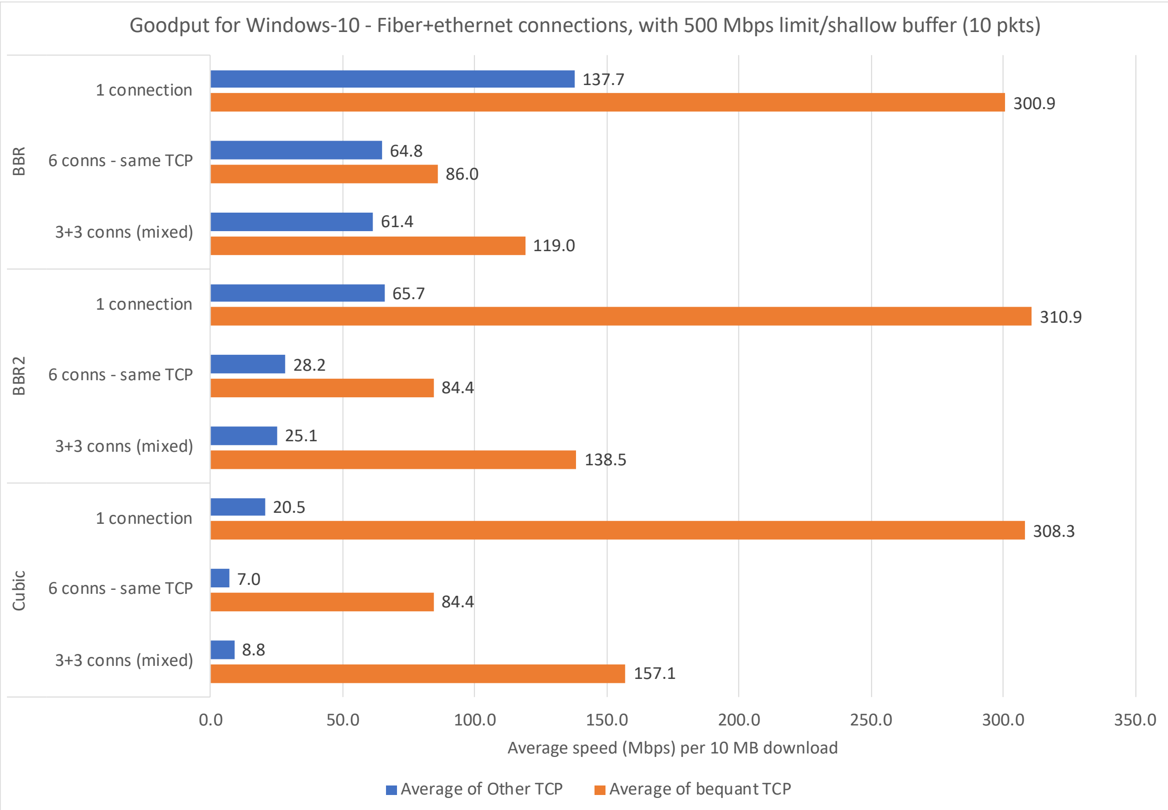

Figure 6. Goodput results Fiber + Ethernet with a shallow buffer (higher is better). The results show the average speed per 10 Mbyte download.

Figure 6 shows how the situation changes significantly when facing shallow buffers. While the Bequant TCP can still manage to get speeds close to 300 Mbps (compared to 400 Mbps with deep buffers), the other TCP variants obtain much lower speeds: 138 Mbps with BBR, 66 Mbps with BBR2, and 20 Mbps with Cubic. This is due to a superior handling by the Bequant TCP of the potentially significant packet losses that can result from high speeds being limited by shallow buffers.

In the case of concurrent connections from the same variant, the Bequant TCP uses all the available 500 Mbps (with average speeds for each of the 6 connections around 85 Mbps), but with some uneven bandwidth sharing, since the 500 Mbps are at IP-level, and an even sharing (with the ~28% losses observed) would point to around 65 Mbps, which is what BBR obtains. BBR2 and Cubic are far from being able to use the whole available bandwidth, reaching only an average of 28 Mbps with BBR2, and 7 Mbps with Cubic.

When running 6 concurrent downloads, with 3 Bequant TCP downloads and 3 downloads with BBR, the average speed of Bequant TCP goes up to 119 Mbps, resulting in a more uneven sharing, but it does not affect the BBR flows very much. Against BBR2 and Cubic, the goodput of Bequant TCP also goes up because the Bequant TCP takes advantage of unused capacity from the other flows (to 139 against BBR2, and 157 Mbps against Cubic), but not because of the Bequant TCP overwhelming the other connections (the other connections get an average speed very similar to what they get when competing against themselves).

4.2.2. Latency

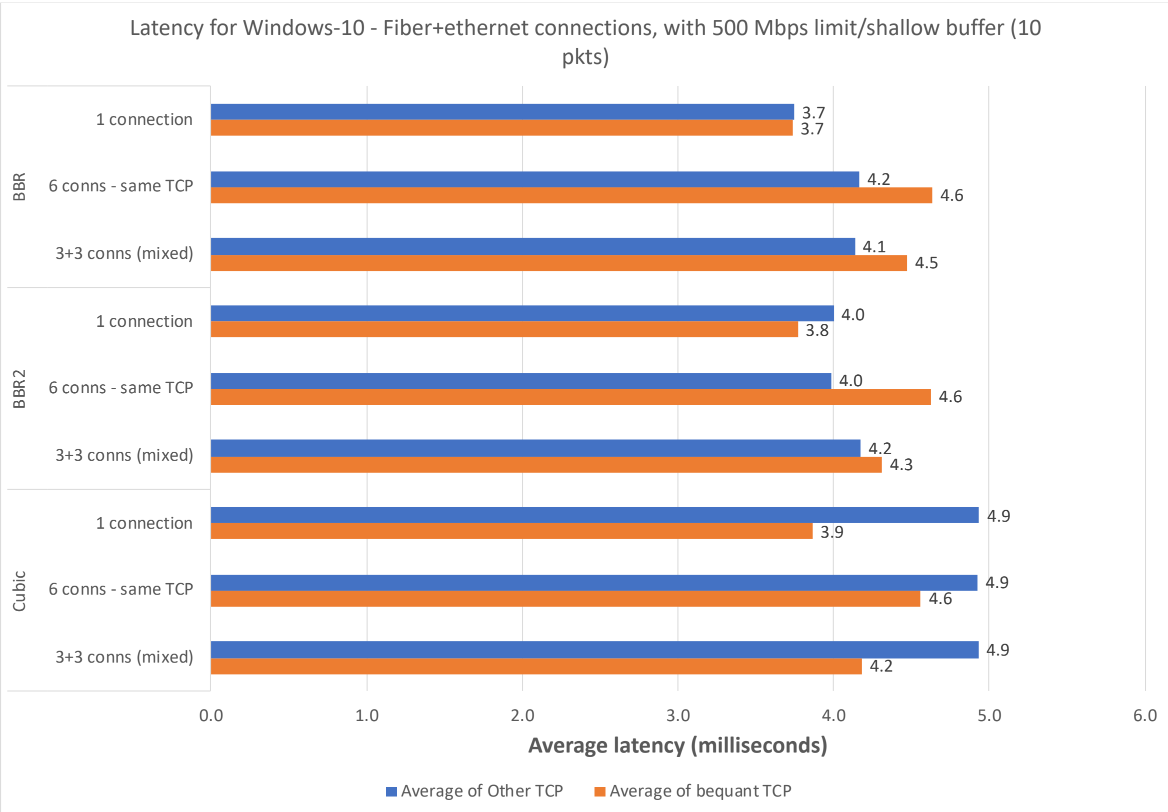

Figure 7. Latency results Fiber + Ethernet with a shallow buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 3.5 milliseconds.

Figure 7 shows little differences in latency in all the scenarios. This is because the buffer, where the latency is formed, is very small and in all cases, it tends to be fully saturated. And at the large speeds involved (~500 Mbps), the delay contribution is rather low.

4.2.3. Losses

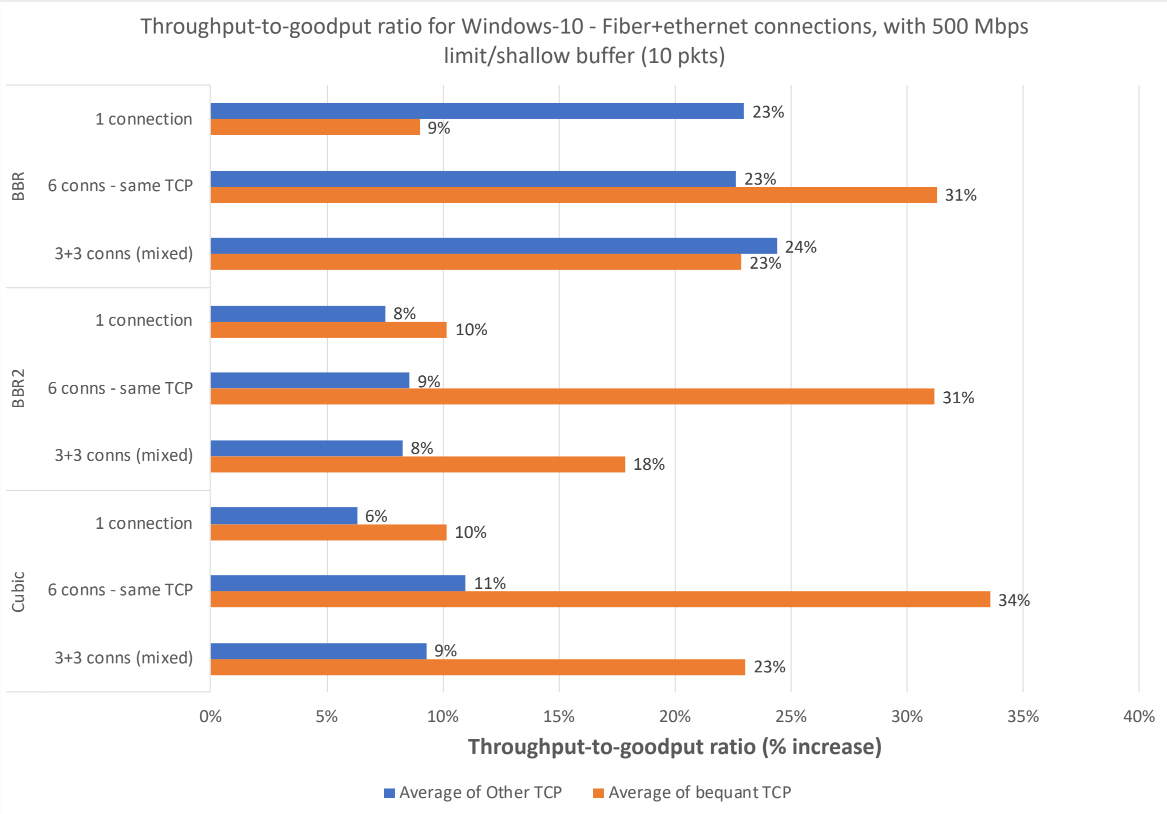

Figure 8. Throughput-to-goodput increase ratio for Fiber + Ethernet with a shallow buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is close to 2.8%, due to TCP protocol overhead.

In the case of single connections, Figure 8 shows higher losses for the Bequant TCP than for BBR2 and Cubic, but this can be explained because it reaches much higher speeds (around 300 Mbps, vs ~20 Mbps for Cubic and ~30 Mbps for BBR2). However, BBR reaches around 140 Mbps, which is less than half of the 300 Mbps of the Bequant TCP) at the expense of more than double the losses (around 20% losses vs less than 10% in the case of Bequant TCP).

In the case of concurrent connections from the same variant, the Bequant TCP displays high losses (around 28%), with a more aggressive behavior than BBR, with losses around 20%. In the case of BBR2 and Cubic, the fact that they incur lower losses (around 10%) is not so relevant, because they use only a fraction of the available bandwidth (7 Mbps per connection for Cubic and 28 Mbps per connection for BBR2).

While the Bequant TCP therefore looks more aggressive when competing against itself, it limits its aggressiveness when competing against other TCP variants. Against BBR, for example, the losses stay at around 20% for both the BBR and the Bequant TCP connections. Still, BBR displays greater evenness with average speeds around 60 Mbps, while the Bequant TCP connections gets double that speed. The comparison against BBR2 and Cubic shows that the Bequant TCP also limits its aggressiveness and does not impact the other TCP variants, which have about the same losses as when competing against themselves, but the speeds obtained by the BBR2 and Cubic connections are only a small fraction of what the available bandwidth would allow.

4.3. Fiber + WiFi with Deep Buffer

Access	Fiber 600 Mbps + WiFi-ac
Client OS	iPadOS
Buffer Depth	Deep (1000 packets)
Maximum Speed	500 Mbps
Latency at Rest	7 ms
Download Size	10 MBytes

4.3.1. Goodput

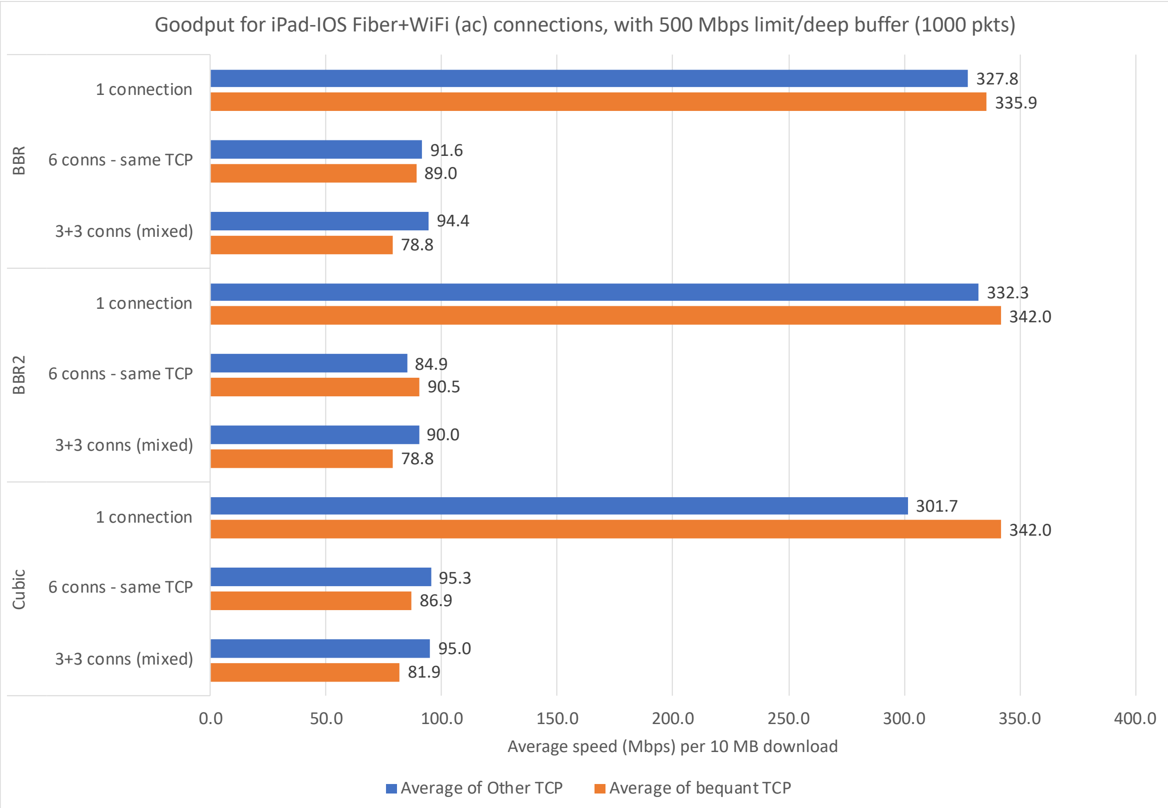

Figure 9. Goodput results for Fiber with WiFi (ac) with a deep buffer (higher is better). The results show the average speed per 10 Mbyte download.

Figure 9 shows the test results for an iPadOS-based iPad mini client and a 600 Mbps fiber through fast WiFi (ac) connection, where a 500 Mbps speed limitation is enforced with a deep buffer (1000-packet buffer). It can be seen that all the TCP variants, except Cubic, can use all the available bandwidth with a single 10 MB connection, reaching an average speed close to 340 Mbps, with no significant differences, which is what can be expected with a 7 ms latency; Cubic falls a little behind (reaching 302 Mbps).

When running six concurrent connections from the same TCP variant, the bandwidth is shared quite evenly between the 6 connections, with an average download speed from 85 to 95 Mbps, which is a little larger than what was observed for the ethernet+fiber connections and can be attributed to the larger variability of WiFi connections, but still denotes a fairly even sharing.

When running 6 concurrent downloads, with 3 Bequant TCP downloads and 3 downloads with a different TCP, the fairness is largely maintained, even though in all cases the Bequant TCP is less aggressive than the other variants, but still, that does not affect the speed of Bequant TCP flows in a major way.

4.3.2. Latency

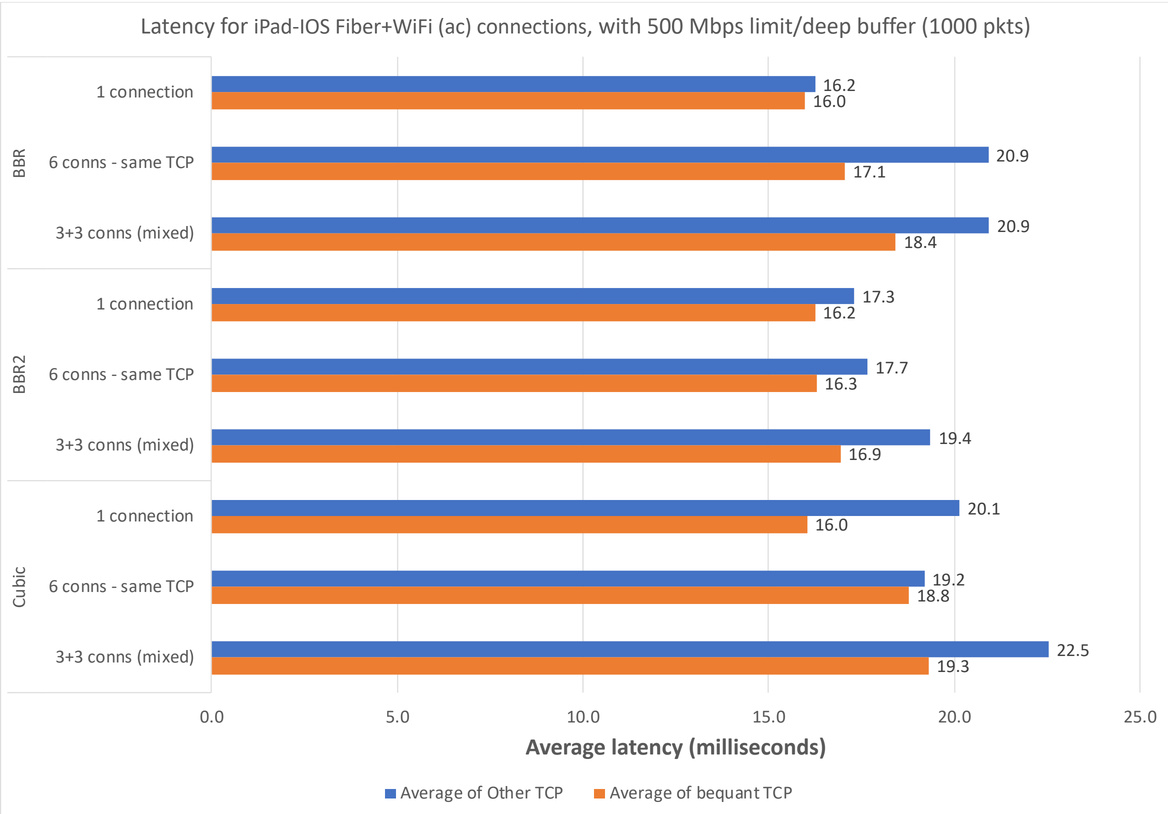

Figure 10. Latency results for Fiber + WiFi (ac) with a deep buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 7 milliseconds.

Figure 10 presents no very large differences between TCP variants and scenarios. The Bequant TCP attains lower latencies in general, but only significant in a few cases: single and mixed Cubic TCP connections, and concurrent BBR connections (both with same and mixed variants).

4.3.3. Losses

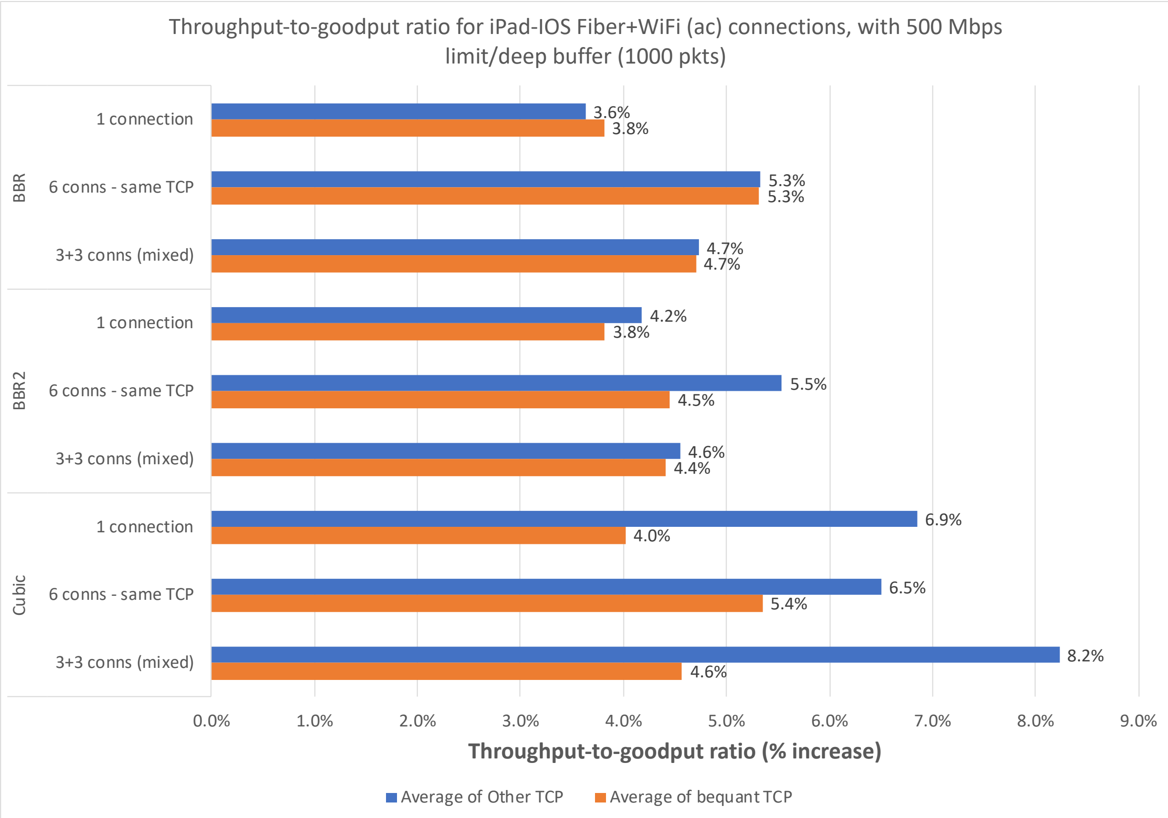

Figure 11. Throughput-to-goodput increase ratio for Fiber + WiFi with a deep buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is around 2.8%, due to TCP protocol overhead.

Figure 11 only shows significantly higher packet losses for Cubic TCP, with both single and concurrent connections.

4.4. Fiber + WiFi with Shallow Buffer

Access	Fiber 600 Mbps + WiFi-ac
Client OS	iPadOS
Buffer Depth	Shallow (10 packets)
Maximum Speed	500 Mbps
Latency at Rest	7 ms
Download Size	10 MBytes

4.4.1. Goodput

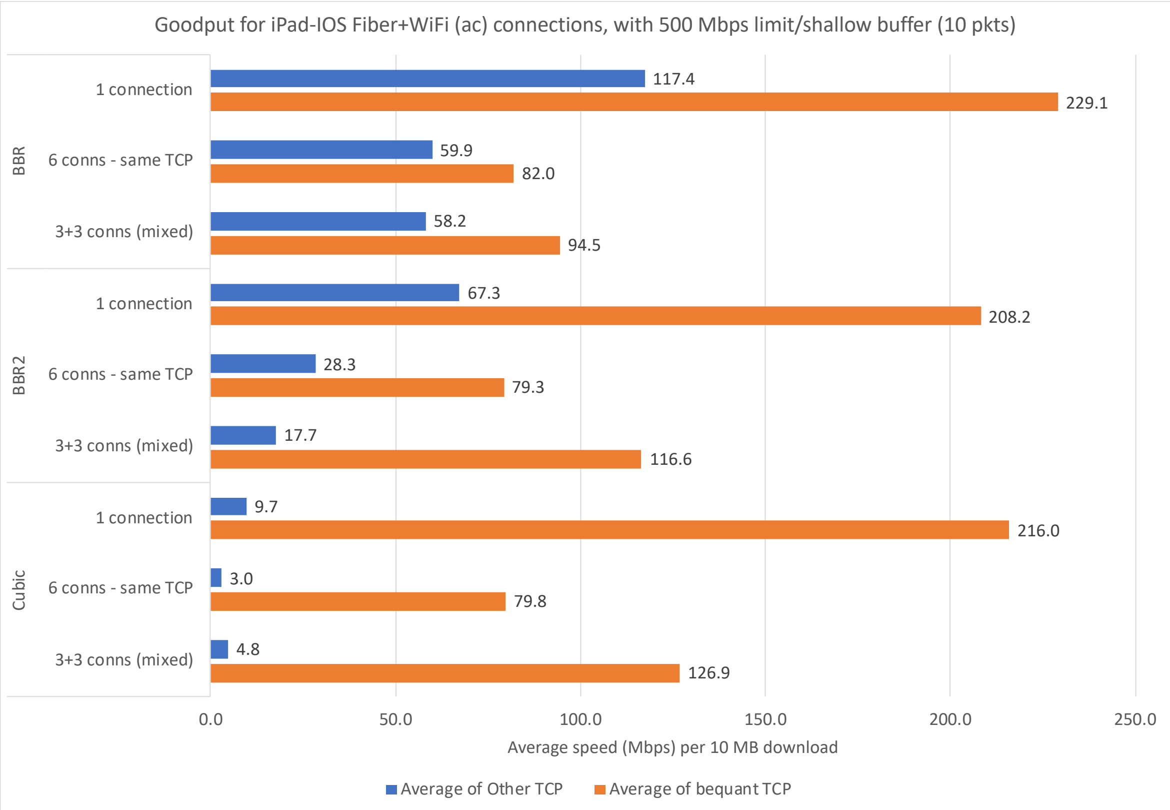

Figure 12. Goodput results for Fiber + WiFi with a shallow buffer (higher is better). The results show the average speed per 10 Mbyte download.

As in the case of ethernet+fiber, Figure 12 shows that the situation changes significantly with shallow buffers. While the Bequant TCP can still manage to get speeds around 210 Mbps (compared to 340 Mbps with deep buffers), the other TCP variants obtain much lower speeds: 117 Mbps with BBR, 67 Mbps with BBR2, and 10 Mbps with Cubic. This is again due to a superior handling by the Bequant TCP of the potentially significant packet losses that can result from high speeds being limited by shallow buffers, even in the face of the variability introduced by WiFi networks.

When running six concurrent downloads from the same TCP variant, the Bequant TCP can make use of all the available bandwidth in a quite fair way (with an average speed of 80 Mbps, as in the case of ethernet+fiber with deep buffers). However, it is evident that the other TCP variants are far from being able to use the whole available bandwidth, reaching only an average of 60 Mbps in the case of BBR, 28 Mbps with BBR2, and 3 Mbps with Cubic.

This is again reflected when running 6 concurrent downloads, with 3 Bequant TCP downloads and 3 downloads with a different TCP. The average speed of Bequant TCP goes up significantly, because there is unused capacity from the other flows (to 94 Mbps against BBR, 117 against BBR2, and 127 Mbps against Cubic), but not because of the Bequant TCP overwhelming the other connections. In fact, the other connections get an average speed very similar to what they get when competing against themselves, except in the case of BBR2, with a larger drop (from 28 to 18 Mbps).

4.4.2. Latency

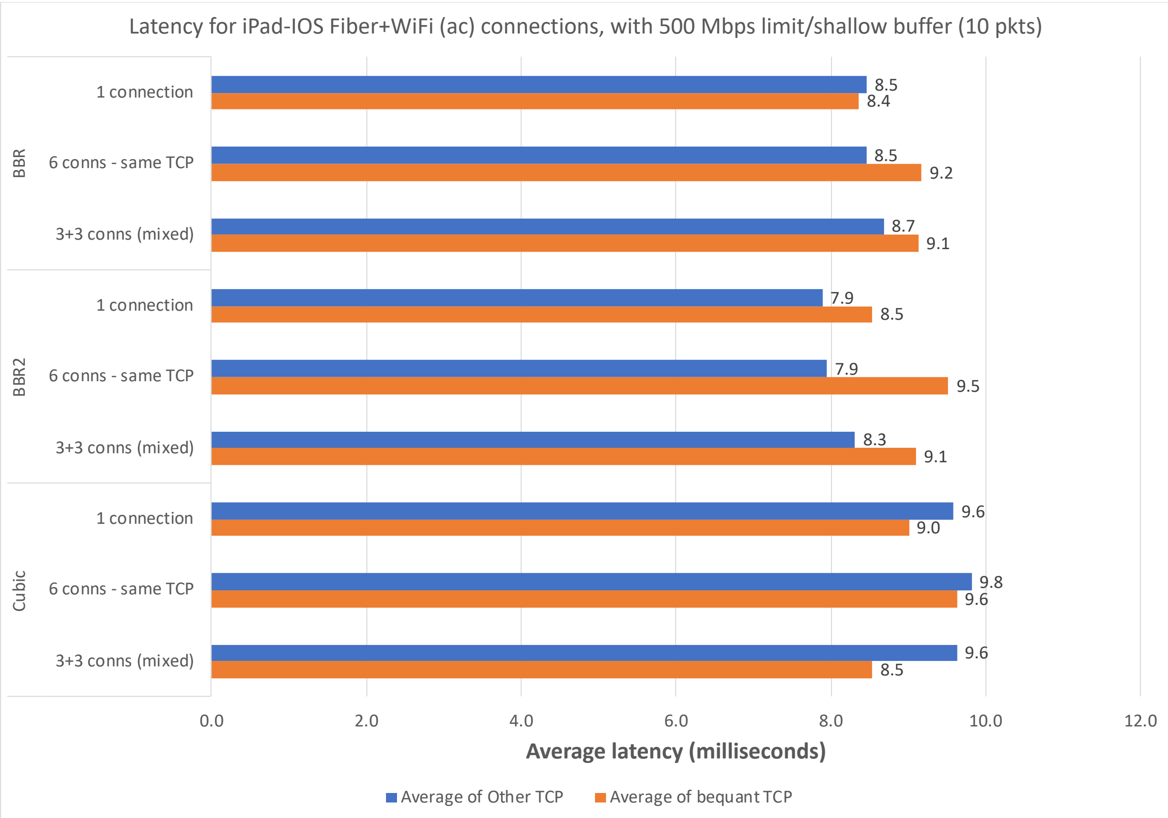

Figure 13. Latency results for Fiber + WiFi with a shallow buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 7 milliseconds.

Figure 13 shows little differences in latency in all the scenarios. This is because the buffer, where the latency is formed, is very small and in all cases, it tends to be fully saturated. And at the large speeds involved (~500 Mbps), the delay contribution is rather low.

4.4.3. Losses

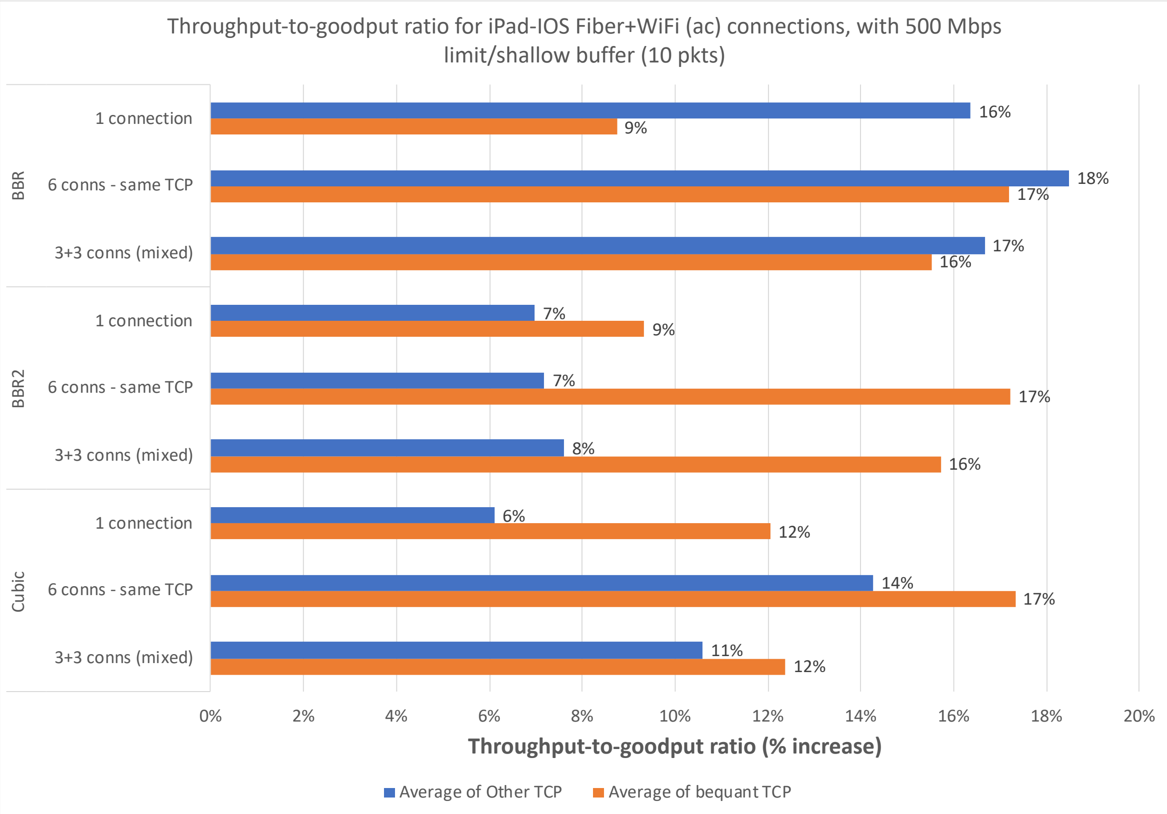

Figure 14. Throughput-to-goodput increase ratio for Fiber + WiFi with a deep buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is around 2.8%, due to TCP protocol overhead.

In the case of single connections, Figure 14 shows slightly losses for the Bequant TCP than for BBR2 and Cubic, but this can be explained because it reaches much higher speeds (over 200 Mbps, vs ~10 Mbps for Cubic and ~70 Mbps for BBR2). However, BBR reaches around 130 Mbps, which close to half of the over 200 Mbps of the Bequant TCP) at the expense of around double the losses (~13% losses vs ~6 in the case of Bequant TCP%, when discounting ~3% from protocol overhead).

In the case of concurrent connections from the same variant, the Bequant TCP displays losses around 15%, using all the available 500 Mbps (with average speeds for each of the 6 connections around 80 Mbps), but with some uneven bandwidth sharing, since the 500 Mbps are at IP-level, and an even sharing would point to around 70 Mbps with losses at 15%, which is a little above what BBR obtains (60 Mbps). In the case of BBR2 and Cubic, the fact that they incur lower losses (~4% for BBR2 and ~10% for Cubic) is not so relevant, because they use only a fraction of the available bandwidth (3 Mbps per connection for Cubic and 28 Mbps per connection for BBR2).

While the Bequant TCP therefore looks more aggressive when competing against itself, it limits its aggressiveness when competing against other TCP variants. Against BBR, for example, the losses stay at around 14% for both the BBR and the Bequant TCP connections. Still, BBR displays greater evenness with average speeds around 60 Mbps, while the Bequant TCP connections gets 80-95 Mbps. The comparison against

BBR2 and Cubic shows that the Bequant TCP also limits its aggressiveness and does not impact the other TCP variants, which have about the same losses as when competing against themselves, but the speeds obtained by the BBR2 and Cubic connections are only a small fraction of what the available bandwidth would allow.

4.5. 4G with Deep Buffer

Access	4G
Client OS	Android
Buffer Depth	Deep (by 4G network)
Maximum Speed	By 4G network
Latency at Rest	23 ms
Download Size	10 MBytes

4.5.1. Goodput

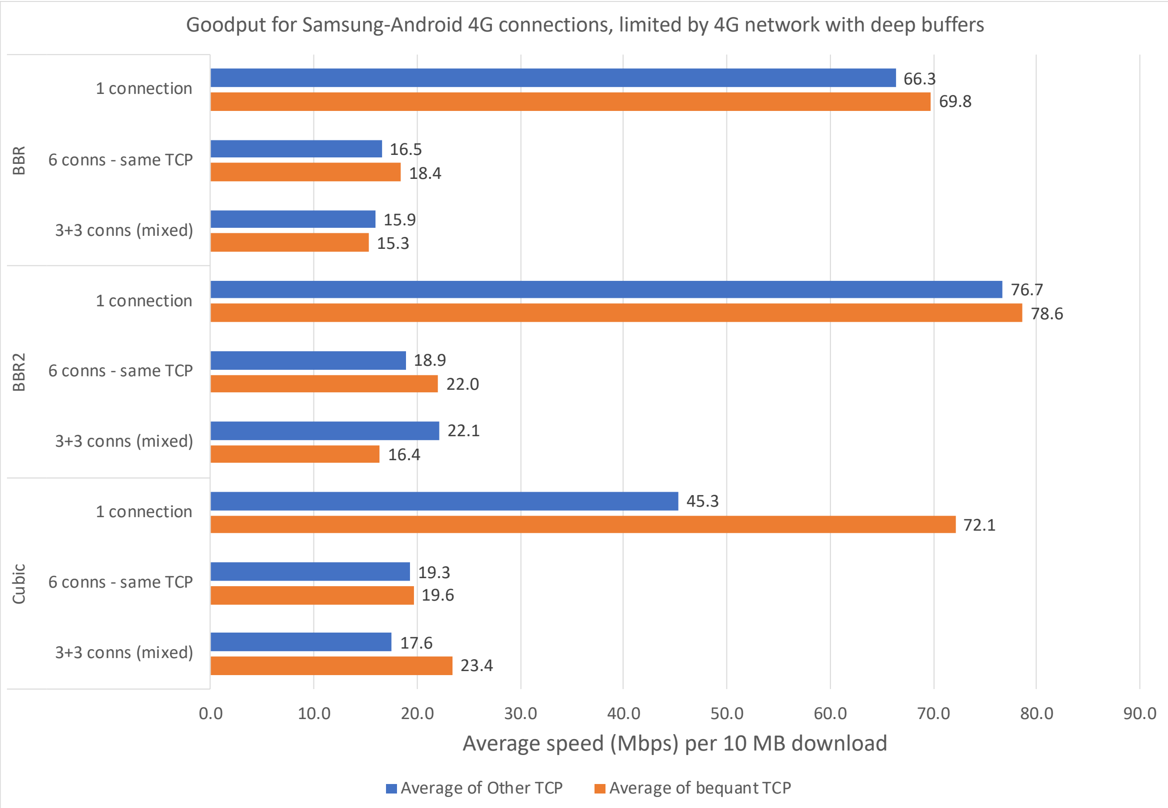

Figure 15. Goodput results for 4G with deep buffer (higher is better). The results show the average speed per 10 Mbyte download.

Figure 15 shows the test results for a Samsung Galaxy Note8 Android device with a 4G connection in downtown Madrid to the Telefonica-Movistar network. The connection speed is just limited by the 4G network, which uses large deep buffers at the eNodeBs. All the TCP variants, except Cubic, get from 66 Mbps to 80 Mbps, which is around what can be expected from a 10 Mbyte download with a 100 Mbps bandwidth and 23 ms of at-rest latency. Cubic just manages to get to 45 Mbps.

When running six concurrent flows from the same TCP variant, the bandwidth is shared quite evenly between the 6 connections, with an average download speed from 16 to 22 Mbps.

When running 6 concurrent downloads, with 3 Bequant TCP downloads and 3 downloads with a different TCP, fairness is largely maintained, even though BBR2 seems to behave a little more aggressively that Bequant TCP, while the latter seems to be more aggressive than Cubic.

4.5.2. Latency

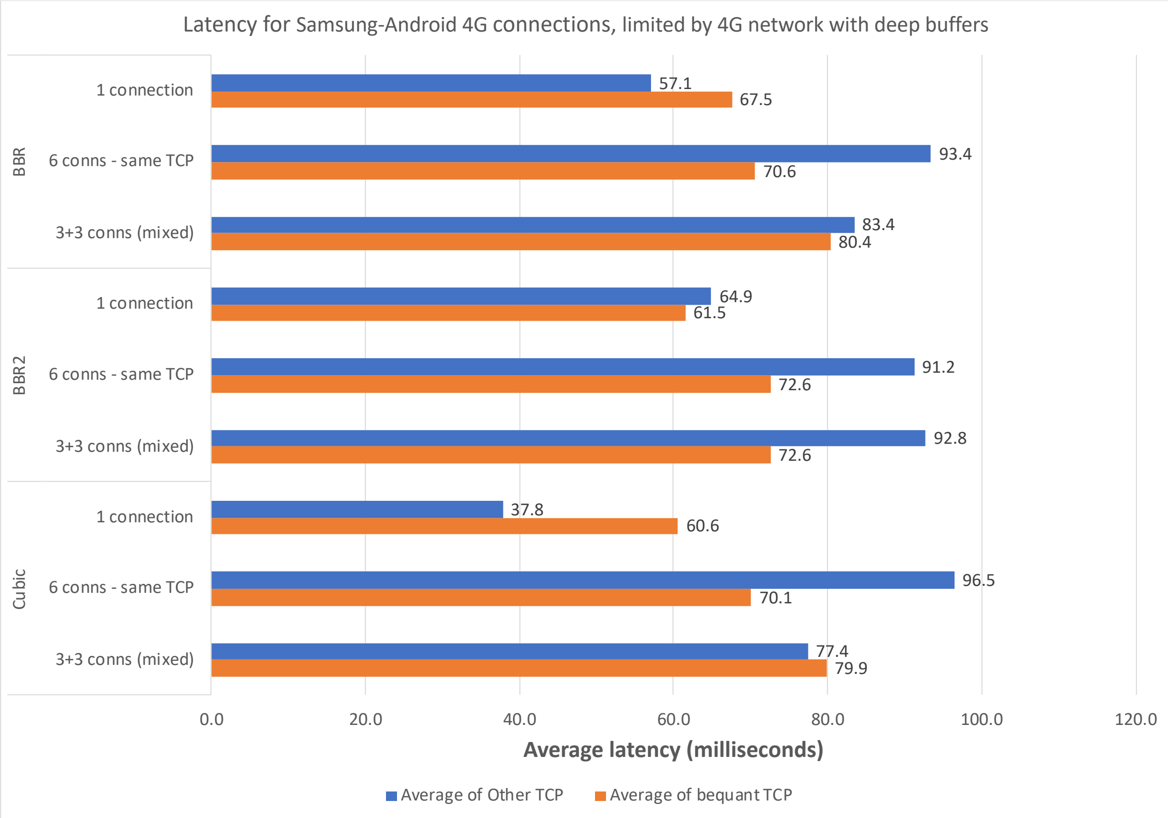

Figure 16. Latency results for 4G with deep buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 25 milliseconds.

The latency results in Figure 15 for single connections mainly show a lower latency for Cubic connections, which is behind the significantly lower speed of Cubic connections. For concurrent connections of the same TCP variant, the Bequant TCP connections display around 20ms reduction in latency, but when it competes against the other variants, that difference only appears against BBR2.

4.5.3. Losses

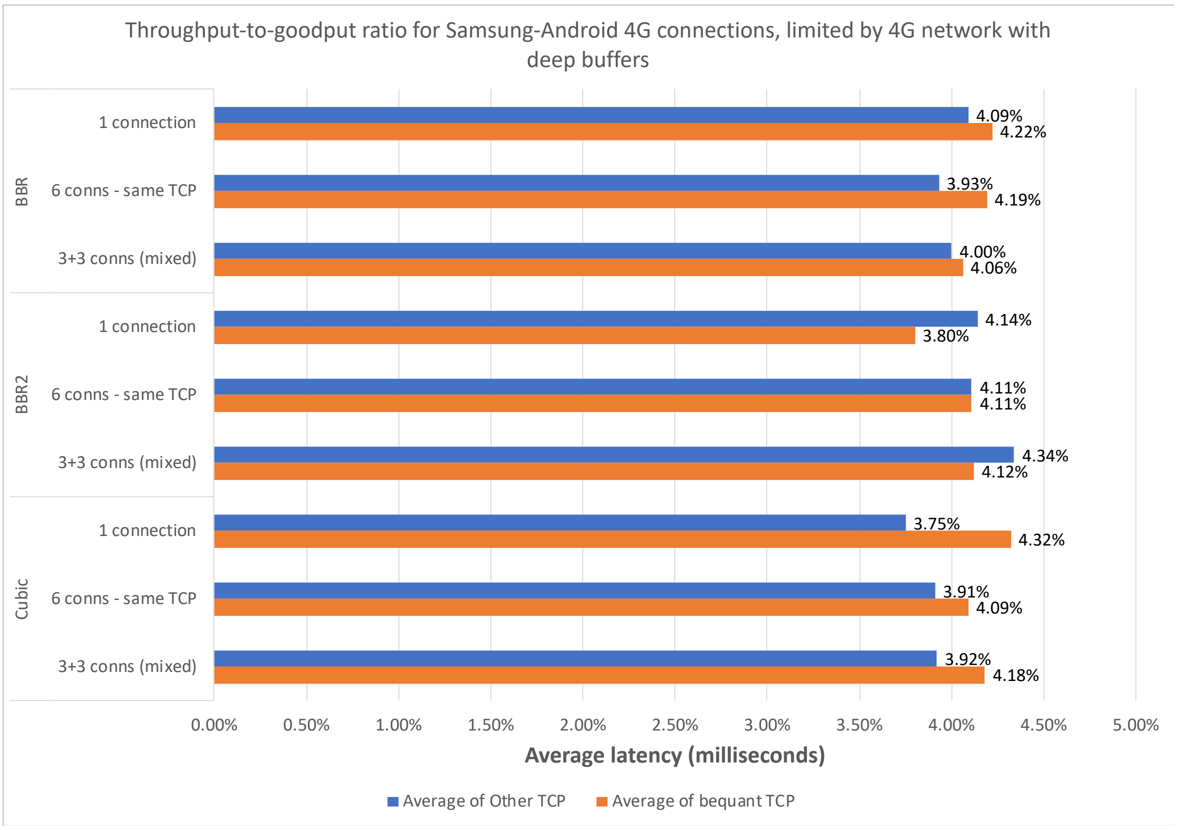

Figure 17. Throughput-to-goodput increase ratio for 4G with deep buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is around 2.8%, due to TCP protocol overhead.

Figure 17 show no relevant differences in packet losses, which can be expected from the deep buffers present in 4G networks and their relatively low speeds.

4.6. 4G with Shallow Buffer

Access	4G
Client OS	Android
Buffer Depth	Shallow (10 packets)
Maximum Speed	50 Mbps
Latency at Rest	23 ms
Download Size	10 MBytes

4.6.1. Goodput

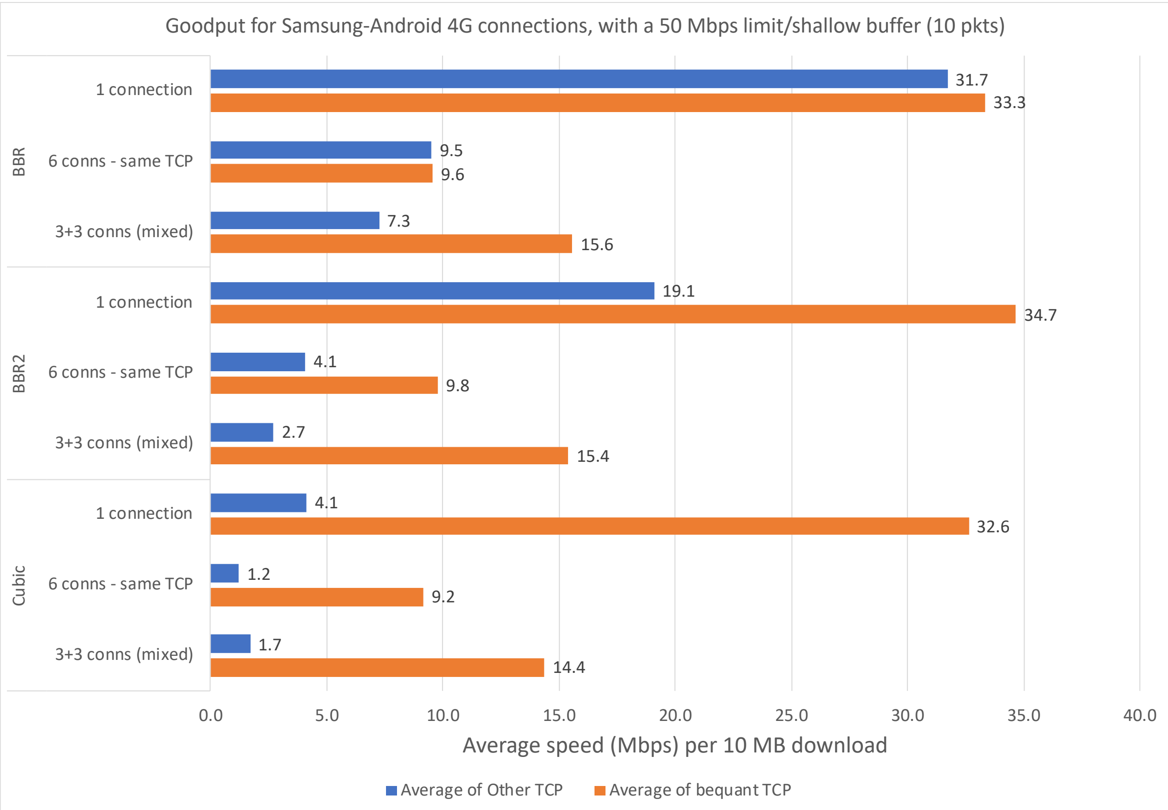

Figure 18. Goodput results for 4G with shallow buffer (higher is better). The results show the average speed per 10 Mbyte download.

Figure 18 shows that the situation changes significantly with shallow buffers and a speed limitation of 50 Mbps. While the Bequant TCP and BBR can still manage to get speeds close to 34 Mbps (compared to 70 to 60 Mbps with deep buffers), the other TCP variants obtain much lower speeds: 19 Mbps with BBR2, and 4 Mbps with Cubic.

When running six concurrent downloads from the same TCP variant, the Bequant TCP and BBR can make use of all the available bandwidth (50 Mbps) in a quite fair way (with an average speed of 9 Mbps, as in the case of ethernet+fiber with deep buffers). However, it is evident that BBR2 and Cubic are far from being able to use the whole available bandwidth, reaching only an average of 4 Mbps.

When running 6 concurrent downloads, with 3 Bequant TCP downloads and 3 downloads with a different TCP, the Bequant TCP seems to be somewhat more aggressive than the other variants and increases its average speed to 15 Mbps. BBR and BBR2 downloads go down, from 9 Mbps to 7 Mbps in the case of BBR, and from 4 Mbps to 3 Mbps in the case of BBR2: significant but still not very much. In the case of Cubic, the download speed, which is very low to start with and leaves bandwidth unused (as with BBR2), goes up by a little: from 1.2 Mbps to 1.5 Mbps.

4.6.2. Latency

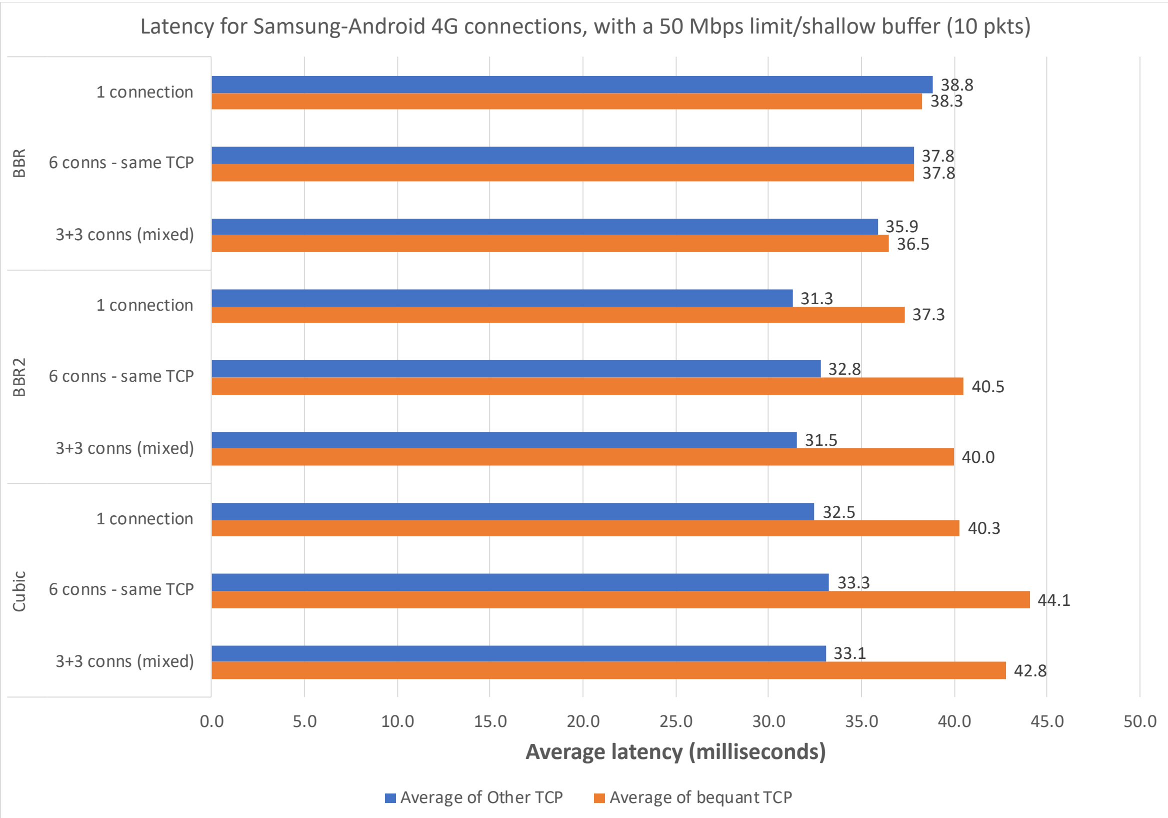

Figure 19. Latency results for 4G with shallow buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 25 milliseconds.

The latency results in Figure 19 only show a significant decrease in latency for BBR2 and Cubic connections, in all scenarios, which may be behind the much lower speed results obtained by those two variants. Against BBR, which is the only variant with speeds comparable to the Bequant TCP, there are no significant differences, as could be expected from the small buffer at the bottleneck.

4.6.3. Losses

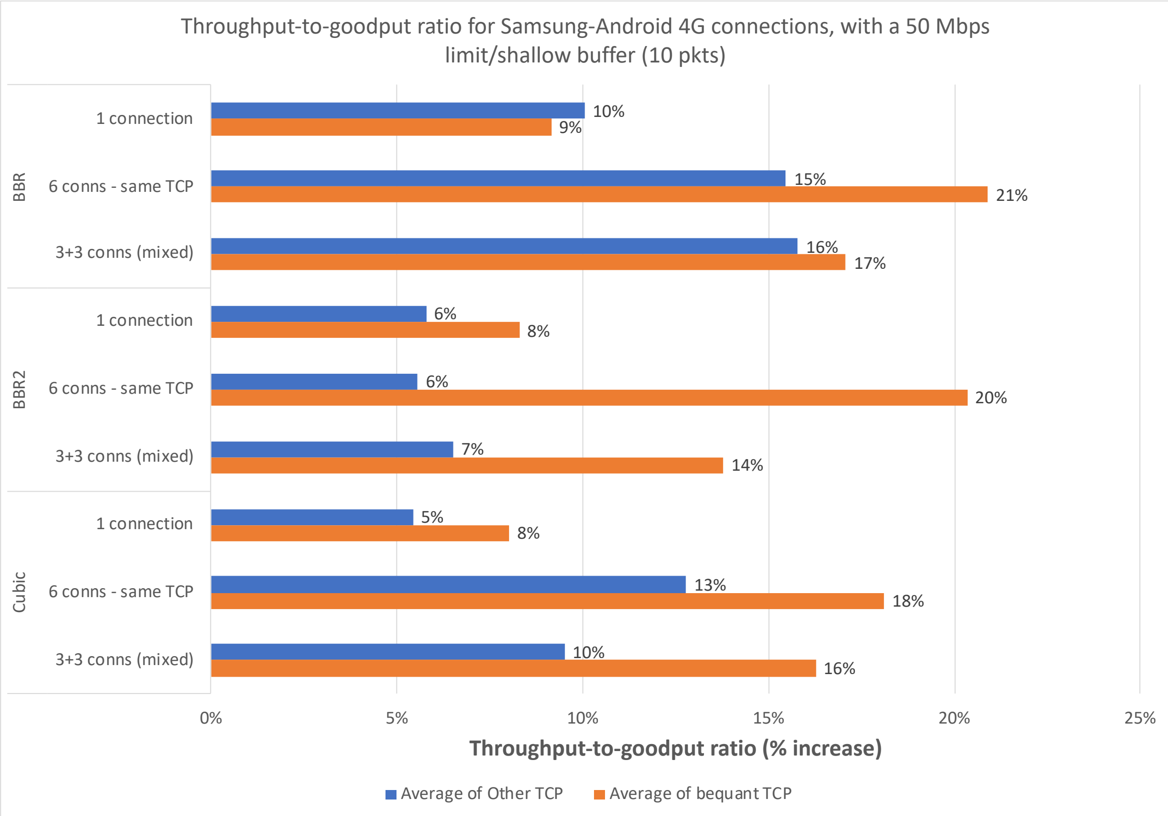

Figure 20. Throughput-to-goodput increase ratio for 4G with shallow buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is around 2.8%, due to TCP protocol overhead.

Again, the significantly lower losses in Figure 20 for BBR2 and Cubic TCP are not so relevant, because of the very low speeds they attain. Comparing with BBR, the Bequant TCP produces more losses when competing against itself, but those packet losses are limited when competing against BBR, where they are about the same. Still, the goodput numbers do point to somewhat more aggressiveness, and less even sharing, in Bequant TCP than in BBR, but without severely affecting the BBR connections when competing against them.

4.7. 5G with Deep Buffers

Access	5G
Client OS	Android
Buffer Depth	Deep (by 5G network)
Maximum Speed	By 5G network
Latency at Rest	13 ms
Download Size	10 MBytes

4.7.1. Goodput

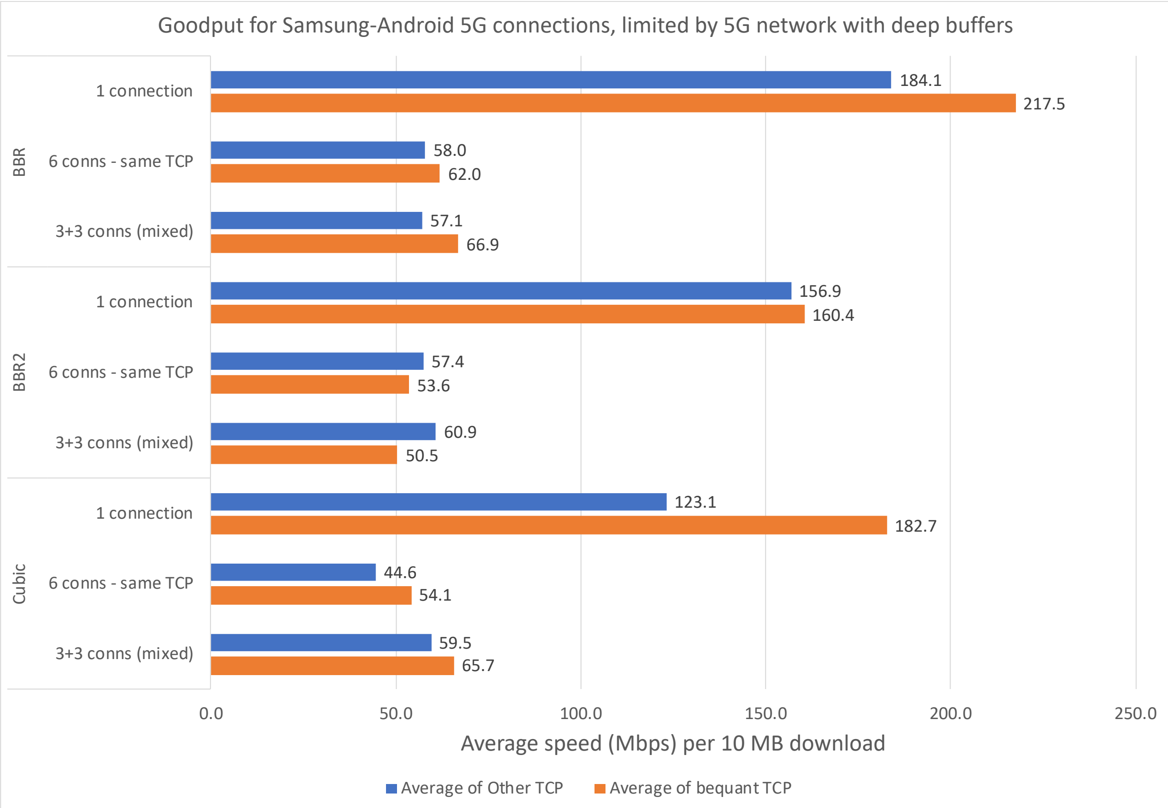

Figure 21. Goodput results for 5G with deep buffer (higher is better). The results show the average speed per 10 Mbyte download.

The goodput results in Figure 21 show some variability in the network speed for different runs, which can be deduced from the results for single connections with Bequant TCP, which should be the same when competing against the different TCP variants, but which oscillates between 160 Mbps (against BBR2) and 217 Mbps (against BBR). So, while it does look like the Bequant TCP can attain higher speeds than BBR (because the runs are alternated), the largest difference is against Cubic.

However, in the case of competing connections, no significant differences are observed between TCP variants.

4.7.2. Latency

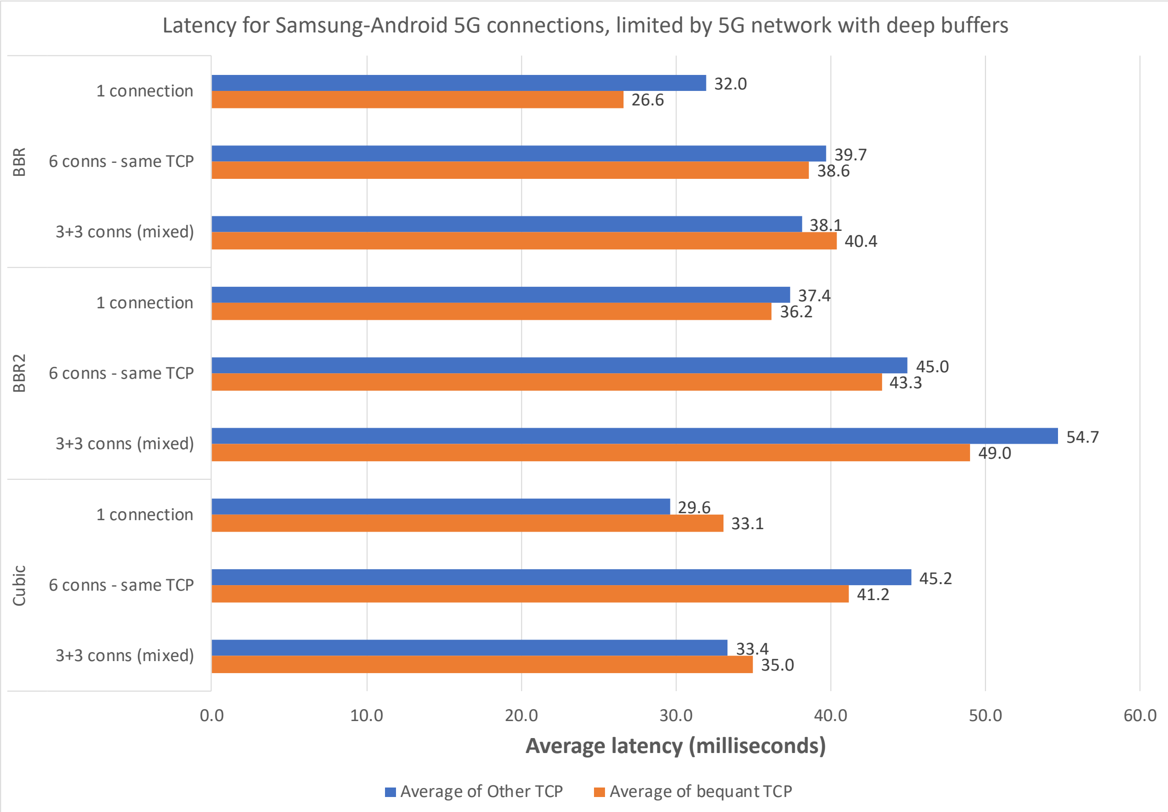

Figure 22. Latency results for 5G with deep buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 15 milliseconds.

No major latency differences can be observed between different TCP variants in Figure 22. It is interesting to observe, that the ultra-low latencies of 5G networks do not happen in reality, at least in this 5G network, and these latencies are on the order of twice those obtained in WiFi+fiber.

4.7.3. Losses

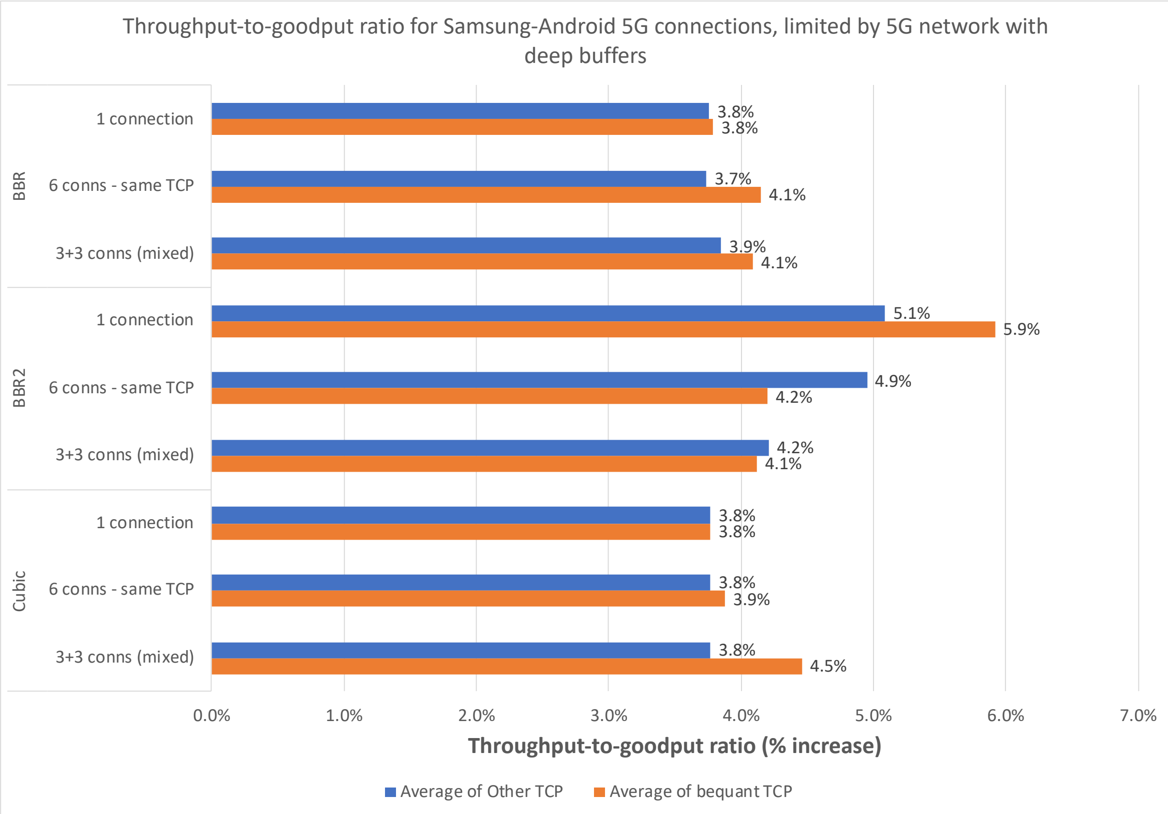

Figure 23. Throughput-to-goodput increase ratio for 5G with deep buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is around 2.8%, due to TCP protocol overhead.

The packet loss results in Figure 23, as in the case of goodput in Figure 21 and also, to some extent, in latency in Figure 22, show some variability in the network conditions for different runs, which can be deduced from the results for single connections with Bequant TCP, which should be the same when competing against the different TCP variants, but which show higher losses against BBR2. In any case, no major differences are displayed between different TCP variants, which is what could be expected from the deep buffers used by these networks.

4.8. 5G with Shallow Buffers

Access	5G
Client OS	Android
Buffer Depth	Shallow (10 packets)
Maximum Speed	50 Mbps
Latency at Rest	13 ms
Download Size	10 MBytes

4.8.1. Goodput

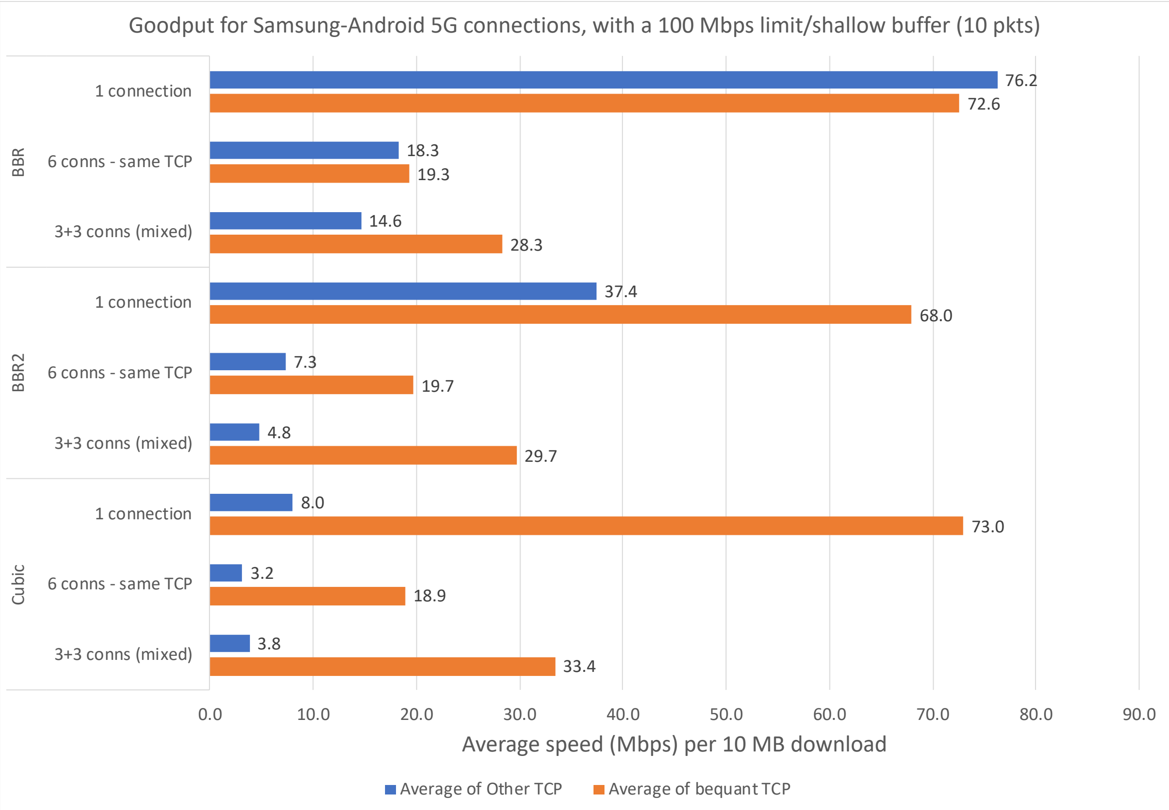

Figure 24. Goodput results for 5G with shallow buffer (higher is better). The results show the average speed per 10 Mbyte download.

As in 4G with shallow buffers, Figure 24 shows that BBR2 and Cubic display a very large drop in goodput with shallow buffers, and only BBR offers a goodput similar to the Bequant TCP. In fact, when competing against Bequant TCP, BBR connections share the available bandwidth in a more even way.

4.8.2. Latency

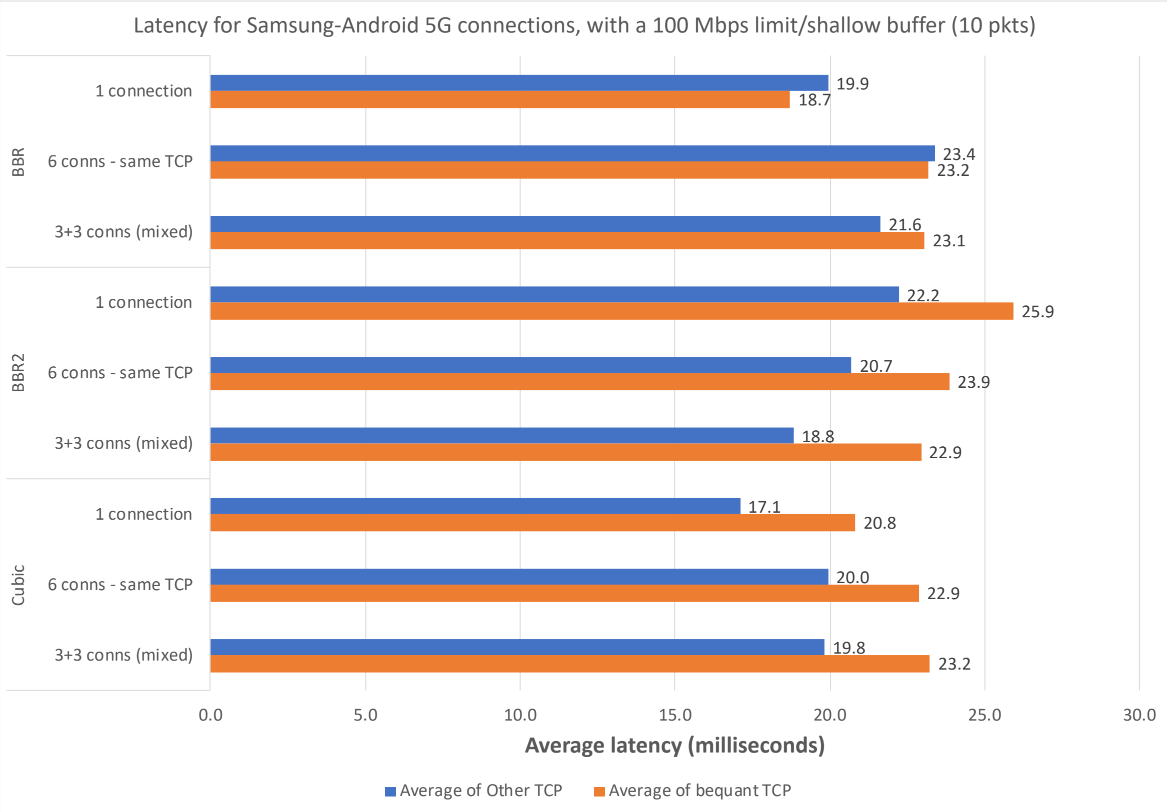

Figure 25. Latency results for 5G with shallow buffer (lower is better). The results show the average latency measured during a 10 Mbyte download. The latency measured at rest was around 15 milliseconds.

No significant differences in latency can be observed between the Bequant TCP and BBR in Figure 25, while the small differences with BBR2 and Cubic are not relevant, because of the large drop in goodput.

4.8.3. Losses

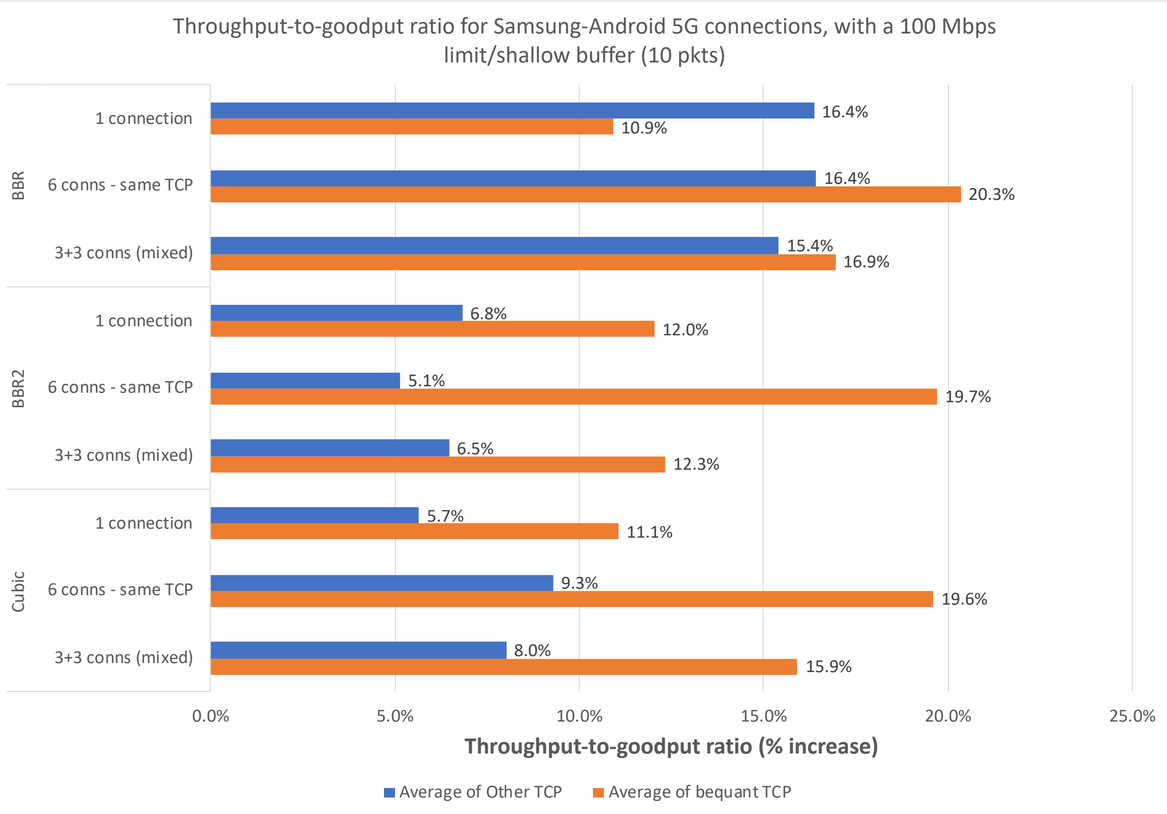

Figure 26. Throughput-to-goodput increase ratio for 5G with shallow buffer (lower is better). The results show how much larger is the throughput compared with the measured goodput, on average, for 10 Mbyte TCP downloads. Without any losses, this ratio is around 2.8%, due to TCP protocol overhead.

Figure 26 shows that BBR suffers about the same amount of packet losses (~13%) in single connection as when competing against Bequant TCP, but the latter goes from lower losses in single connections (~8%) to higher losses when competing against other Bequant TCP flows (~17%). The comparison with BBR2 and Cubic is not relevant, because of the dramatic decrease in performance they entail.

5. Main Conclusions

- The tests presented have been performed under conditions representative of current fast commercial networks, equipment and services. Shallow buffers are very common in network appliances and bandwidth shapers and should be taken into account.

- In the deep buffer scenarios:

    - Without competing connections, the Bequant TCP delivers similar download speeds to BBR and BBR2 in all deep-buffer scenarios, and is significantly faster than Cubic in wireless networks (WiFi , 4G, and 5G).
    - There seem to be no large differences in fairness in congestion situations with competing connections of all TCP variants considered.
    - Latency is generally lower with Bequant TCP, especially in fiber connections, and not so much in 5G.
    - There are no major differences in packet losses, except for the higher losses of Cubic in the ipadOs+WiFi scenario.

- In shallow buffer scenarios:

    - Without competing connections, the Bequant TCP is significantly faster than all the other variants and in all conditions, except for BBR in 4G and 5G.
    - With competing connections in congestion, the Bequant TCP is also significantly faster, but without overwhelming the other connections (the Bequant TCP mainly takes advantage of the capacity left unused by the connections from the other TCP variants).
    - In the case of competition with BBR flows in 4G and 5G, the Bequant TCP is somewhat more aggressive, but still does not affect the BBR flows very much.

- This study has not considered the additional improvements that could be delivered by the BQN as a TCP proxy, since the content server was placed right next to the BQN node, with no latency in-between.

A. Appendix: Full Results

						Average Speed (Mbps)			Average IP volume (MB)			Throughput/Gooput (%)		
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Windows-10	fiber+eth	500 Mbps	deep (1000)	Cubic	1 connection	396.851	397.489	-0.16	10.28	10.276	0.037	2.8%	2.8%	1.4%
Windows-10	fiber+eth	500 Mbps	deep (1000)	Cubic	6 conns - same TCP	79.186	92.086	-14.009	10.398	10.317	0.785	4.0%	3.2%	25.6%
Windows-10	fiber+eth	500 Mbps	deep (1000)	Cubic	3+3 conns (mixed)	77.952	131.124	-40.551	10.291	10.31	-0.185	2.9%	3.1%	-6.1%
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR	1 connection	397.708	400.892	-0.794	10.28	10.276	0.041	2.8%	2.8%	1.4%
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR	6 conns - same TCP	85.359	80.961	5.432	10.296	10.277	0.192	3.0%	2.8%	6.9%
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR	3+3 conns (mixed)	77.551	103.057	-24.749	10.288	10.278	0.09	2.9%	2.8%	3.6%
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR2	1 connection	402.991	394.171	2.238	10.282	10.276	0.057	2.8%	2.8%	2.2%
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR2	6 conns - same TCP	79.068	87.049	-9.169	10.288	10.277	0.108	2.9%	2.8%	4.0%
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR2	3+3 conns (mixed)	82.57	97.179	-15.033	10.331	10.287	0.425	3.3%	2.9%	15.3%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Windows-10	fiber+eth	500 Mbps	shallow (10)	Cubic	1 connection	308.284	20.496	1404.1	11.016	10.63	3.631	10.2%	6.3%	61.3%
Windows-10	fiber+eth	500 Mbps	shallow (10)	Cubic	6 conns - same TCP	84.383	6.959	1112.5	13.357	11.099	20.348	33.6%	11.0%	205.5%
Windows-10	fiber+eth	500 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	157.085	8.837	1677.6	12.304	10.929	12.577	23.0%	9.3%	148.0%
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR	1 connection	300.874	137.664	118.56	10.9	12.298	-11.367	9.0%	23.0%	-60.8%
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR	6 conns - same TCP	85.986	64.824	32.645	13.127	12.263	7.045	31.3%	22.6%	38.2%
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR	3+3 conns (mixed)	118.978	61.388	93.814	12.286	12.438	-1.22	22.9%	24.4%	-6.2%
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR2	1 connection	310.871	65.722	373.01	11.019	10.752	2.476	10.2%	7.5%	35.5%
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR2	6 conns - same TCP	84.392	28.191	199.36	13.117	10.857	20.814	31.2%	8.6%	263.7%
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	138.503	25.054	452.82	11.784	10.827	8.84	17.8%	8.3%	115.7%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	Cubic	1 connection	342.029	301.663	13.381	10.402	10.685	-2.65	4.0%	6.9%	-41.3%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	Cubic	6 conns - same TCP	86.944	95.274	-8.743	10.535	10.65	-1.077	5.4%	6.5%	-17.7%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	Cubic	3+3 conns (mixed)	81.867	95.005	-13.828	10.457	10.823	-3.378	4.6%	8.2%	-44.5%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR	1 connection	335.875	327.755	2.477	10.381	10.363	0.174	3.8%	3.6%	5.0%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR	6 conns - same TCP	88.984	91.569	-2.823	10.532	10.533	-0.008	5.3%	5.3%	-0.2%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR	3+3 conns (mixed)	78.801	94.368	-16.496	10.47	10.473	-0.03	4.7%	4.7%	-0.6%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR2	1 connection	341.997	332.339	2.906	10.382	10.418	-0.348	3.8%	4.2%	-8.6%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR2	6 conns - same TCP	90.456	84.908	6.533	10.445	10.554	-1.04	4.5%	5.5%	-19.7%
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR2	3+3 conns (mixed)	78.846	90.029	-12.422	10.441	10.455	-0.135	4.4%	4.6%	-3.1%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	Cubic	1 connection	215.963	9.728	2120.1	11.205	10.611	5.597	12.1%	6.1%	97.2%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	Cubic	6 conns - same TCP	79.821	2.954	2602	11.732	11.427	2.668	17.3%	14.3%	21.4%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	126.889	4.789	2549.4	11.238	11.06	1.604	12.4%	10.6%	16.8%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR	1 connection	229.078	117.366	95.183	10.874	11.637	-6.558	8.7%	16.4%	-46.6%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR	6 conns - same TCP	82.028	59.877	36.995	11.719	11.848	-1.084	17.2%	18.5%	-7.0%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR	3+3 conns (mixed)	94.487	58.153	62.48	11.552	11.666	-0.978	15.5%	16.7%	-6.8%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR2	1 connection	208.192	67.348	209.13	10.934	10.698	2.209	9.3%	7.0%	33.8%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR2	6 conns - same TCP	79.255	28.294	180.11	11.722	10.719	9.355	17.2%	7.2%	139.5%
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	116.564	17.703	558.46	11.572	10.761	7.534	15.7%	7.6%	106.6%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	4G	Unlimited	deep (1000)	Cubic	1 connection	72.111	45.312	59.143	10.432	10.375	0.546	4.3%	3.8%	15.2%
Android	4G	Unlimited	deep (1000)	Cubic	6 conns - same TCP	19.635	19.282	1.834	10.409	10.391	0.17	4.1%	3.9%	4.6%
Android	4G	Unlimited	deep (1000)	Cubic	3+3 conns (mixed)	23.417	17.56	33.353	10.418	10.392	0.251	4.2%	3.9%	6.6%
Android	4G	Unlimited	deep (1000)	BBR	1 connection	69.768	66.331	5.183	10.422	10.409	0.121	4.2%	4.1%	3.2%
Android	4G	Unlimited	deep (1000)	BBR	6 conns - same TCP	18.422	16.547	11.334	10.419	10.393	0.25	4.2%	3.9%	6.6%
Android	4G	Unlimited	deep (1000)	BBR	3+3 conns (mixed)	15.312	15.949	-3.996	10.406	10.4	0.059	4.1%	4.0%	1.5%
Android	4G	Unlimited	deep (1000)	BBR2	1 connection	78.642	76.716	2.51	10.38	10.414	-0.323	3.8%	4.1%	-8.2%
Android	4G	Unlimited	deep (1000)	BBR2	6 conns - same TCP	22.047	18.874	16.814	10.411	10.411	-0.004	4.1%	4.1%	0.0%
Android	4G	Unlimited	deep (1000)	BBR2	3+3 conns (mixed)	16.376	22.144	-26.047	10.412	10.434	-0.219	4.1%	4.3%	-5.1%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	4G	50 Mbps	shallow (10)	Cubic	1 connection	32.631	4.139	688.44	10.802	10.543	2.452	8.0%	5.4%	47.7%
Android	4G	50 Mbps	shallow (10)	Cubic	6 conns - same TCP	9.17	1.226	647.97	11.811	11.278	4.731	18.1%	12.8%	41.7%
Android	4G	50 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	14.377	1.701	744.96	11.626	10.951	6.163	16.3%	9.5%	71.0%
Android	4G	50 Mbps	shallow (10)	BBR	1 connection	33.348	31.715	5.148	10.915	11.005	-0.817	9.1%	10.1%	-9.0%
Android	4G	50 Mbps	shallow (10)	BBR	6 conns - same TCP	9.571	9.518	0.556	12.09	11.547	4.706	20.9%	15.5%	35.1%
Android	4G	50 Mbps	shallow (10)	BBR	3+3 conns (mixed)	15.582	7.25	114.93	11.708	11.579	1.115	17.1%	15.8%	8.2%
Android	4G	50 Mbps	shallow (10)	BBR2	1 connection	34.677	19.098	81.569	10.829	10.578	2.373	8.3%	5.8%	43.4%
Android	4G	50 Mbps	shallow (10)	BBR2	6 conns - same TCP	9.787	4.085	139.61	12.034	10.556	14.007	20.3%	5.6%	265.8%
Android	4G	50 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	15.408	2.717	467.09	11.378	10.651	6.828	13.8%	6.5%	111.7%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	5G	Unlimited	deep (1000)	Cubic	1 connection	182.694	123.135	48.368	10.377	10.377	0.003	3.8%	3.8%	0.0%
Android	5G	Unlimited	deep (1000)	Cubic	6 conns - same TCP	54.091	44.587	21.316	10.388	10.377	0.1	3.9%	3.8%	2.9%
Android	5G	Unlimited	deep (1000)	Cubic	3+3 conns (mixed)	65.655	59.494	10.355	10.446	10.377	0.668	4.5%	3.8%	18.3%
Android	5G	Unlimited	deep (1000)	BBR	1 connection	217.483	184.065	18.156	10.379	10.376	0.033	3.8%	3.8%	0.8%
Android	5G	Unlimited	deep (1000)	BBR	6 conns - same TCP	61.975	58.01	6.835	10.415	10.374	0.404	4.1%	3.7%	11.0%
Android	5G	Unlimited	deep (1000)	BBR	3+3 conns (mixed)	66.907	57.103	17.169	10.409	10.385	0.224	4.1%	3.9%	6.2%
Android	5G	Unlimited	deep (1000)	BBR2	1 connection	160.394	156.878	2.241	10.592	10.509	0.785	5.9%	5.1%	16.3%
Android	5G	Unlimited	deep (1000)	BBR2	6 conns - same TCP	53.607	57.445	-6.68	10.42	10.495	-0.711	4.2%	4.9%	-15.2%
Android	5G	Unlimited	deep (1000)	BBR2	3+3 conns (mixed)	50.478	60.861	-17.06	10.412	10.421	-0.081	4.1%	4.2%	-2.1%
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	5G	100 Mbps	shallow (10)	Cubic	1 connection	72.978	8.033	808.48	11.106	10.565	5.127	11.1%	5.7%	95.8%
Android	5G	100 Mbps	shallow (10)	Cubic	6 conns - same TCP	18.916	3.155	499.56	11.96	10.929	9.441	19.6%	9.3%	111.0%
Android	5G	100 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	33.421	3.814	776.23	11.592	10.801	7.316	15.9%	8.0%	98.8%
Android	5G	100 Mbps	shallow (10)	BBR	1 connection	72.591	76.228	-4.771	11.091	11.636	-4.678	10.9%	16.4%	-33.3%
Android	5G	100 Mbps	shallow (10)	BBR	6 conns - same TCP	19.294	18.26	5.659	12.034	11.642	3.369	20.3%	16.4%	23.9%
Android	5G	100 Mbps	shallow (10)	BBR	3+3 conns (mixed)	28.296	14.617	93.58	11.694	11.541	1.331	16.9%	15.4%	9.9%
Android	5G	100 Mbps	shallow (10)	BBR2	1 connection	67.967	37.445	81.513	11.203	10.682	4.877	12.0%	6.8%	76.4%
Android	5G	100 Mbps	shallow (10)	BBR2	6 conns - same TCP	19.687	7.328	168.66	11.97	10.514	13.843	19.7%	5.1%	283.3%
Android	5G	100 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	29.679	4.797	518.67	11.233	10.646	5.51	12.3%	6.5%	90.9%

						Average maximum RTT (ms)			Average mean RTT (ms)			Average minimum RTT (ms)		
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Windows-10	fiber+eth	500 Mbps	deep (1000)	Cubic	1 connection	14.06	17.505	-19.676	6.824	9.65	-29.286	3.729	4.403	-15.318
Windows-10	fiber+eth	500 Mbps	deep (1000)	Cubic	6 conns - same TCP	33.523	29.761	12.642	11.566	19.292	-40.047	4.424	6.022	-26.53
Windows-10	fiber+eth	500 Mbps	deep (1000)	Cubic	3+3 conns (mixed)	29.618	30.763	-3.72	9.787	18.902	-48.222	3.424	6.307	-45.708
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR	1 connection	13.806	20.369	-32.223	6.785	9.612	-29.413	3.748	3.685	1.703
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR	6 conns - same TCP	28.197	31.158	-9.503	11.35	17.262	-34.249	4.46	4.621	-3.483
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR	3+3 conns (mixed)	31.15	36.008	-13.492	10.141	17.975	-43.586	3.694	5.223	-29.271
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR2	1 connection	13.924	16.923	-17.724	7.083	8.65	-18.118	3.811	3.783	0.74
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR2	6 conns - same TCP	26.244	29.156	-9.989	10.443	15.043	-30.58	4.138	4.486	-7.745
Windows-10	fiber+eth	500 Mbps	deep (1000)	BBR2	3+3 conns (mixed)	29.237	29.492	-0.867	12.096	14.088	-14.144	4.492	4.72	-4.831
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Windows-10	fiber+eth	500 Mbps	shallow (10)	Cubic	1 connection	7.898	6.381	23.784	3.864	4.935	-21.708	3.208	4.289	-25.191
Windows-10	fiber+eth	500 Mbps	shallow (10)	Cubic	6 conns - same TCP	13.016	7.613	70.964	4.56	4.924	-7.387	3.224	4.141	-22.14
Windows-10	fiber+eth	500 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	9.159	8.342	9.795	4.181	4.929	-15.178	3.175	3.796	-16.364
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR	1 connection	7.28	7.415	-1.818	3.736	3.746	-0.272	3.154	3.338	-5.526
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR	6 conns - same TCP	11.055	12.275	-9.938	4.634	4.165	11.265	3.227	3.421	-5.678
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR	3+3 conns (mixed)	10.774	11.24	-4.146	4.472	4.135	8.164	3.251	3.33	-2.37
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR2	1 connection	6.943	7.005	-0.881	3.768	3.998	-5.752	3.143	3.459	-9.141
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR2	6 conns - same TCP	12.196	9.572	27.414	4.625	3.984	16.083	3.229	3.461	-6.705
Windows-10	fiber+eth	500 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	9.97	10.396	-4.097	4.31	4.173	3.288	3.216	3.569	-9.892
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	Cubic	1 connection	31.342	36.917	-15.104	16.029	20.138	-20.404	7.477	9.745	-23.271
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	Cubic	6 conns - same TCP	37.689	34.385	9.61	18.772	19.185	-2.152	7.73	9.228	-16.234
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	Cubic	3+3 conns (mixed)	36.825	36.661	0.449	19.308	22.537	-14.329	7.742	9.431	-17.91
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR	1 connection	29.912	31.068	-3.719	15.972	16.242	-1.663	7.396	7.681	-3.715
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR	6 conns - same TCP	37.753	37.721	0.084	17.055	20.907	-18.425	7.513	7.85	-4.305
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR	3+3 conns (mixed)	34.465	34.702	-0.683	18.426	20.931	-11.969	7.329	8.264	-11.312
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR2	1 connection	28.768	31.294	-8.075	16.243	17.3	-6.11	7.454	7.505	-0.678
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR2	6 conns - same TCP	35.175	39.63	-11.243	16.287	17.656	-7.752	7.491	7.606	-1.517
iPad-IOS	fiber+WiFi	500 Mbps	deep (1000)	BBR2	3+3 conns (mixed)	35.381	34.834	1.57	16.93	19.358	-12.539	7.115	8.122	-12.398
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	Cubic	1 connection	24.321	11.137	118.38	9.006	9.585	-6.038	6.813	8.758	-22.204
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	Cubic	6 conns - same TCP	24.9	13.568	83.521	9.629	9.818	-1.922	6.996	8.731	-19.87
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	17.336	15.788	9.802	8.537	9.628	-11.331	6.769	7.801	-13.233
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR	1 connection	21.622	22.28	-2.956	8.363	8.462	-1.165	6.629	6.899	-3.916
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR	6 conns - same TCP	18.105	16.9	7.132	9.175	8.458	8.473	7.088	7.033	0.788
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR	3+3 conns (mixed)	17.967	18.622	-3.52	9.132	8.683	5.164	7.251	6.854	5.801
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR2	1 connection	25.82	15.901	62.375	8.537	7.89	8.203	6.647	6.891	-3.543
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR2	6 conns - same TCP	25.48	14.179	79.698	9.521	7.949	19.78	7.046	6.856	2.768
iPad-IOS	fiber+WiFi	500 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	21.737	19.988	8.748	9.101	8.314	9.466	7.039	6.902	1.979
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	4G	Unlimited	deep (1000)	Cubic	1 connection	123.031	59.56	106.57	60.607	37.809	60.298	24.891	26.471	-5.971
Android	4G	Unlimited	deep (1000)	Cubic	6 conns - same TCP	246.377	174.854	40.904	70.088	96.465	-27.344	28.207	33.151	-14.914
Android	4G	Unlimited	deep (1000)	Cubic	3+3 conns (mixed)	171.759	162.682	5.579	79.873	77.414	3.177	28.876	29.09	-0.736
Android	4G	Unlimited	deep (1000)	BBR	1 connection	135.251	130.73	3.458	67.525	57.065	18.33	24.55	25.872	-5.109
Android	4G	Unlimited	deep (1000)	BBR	6 conns - same TCP	175.571	231.989	-24.319	70.595	93.428	-24.439	27.475	28.896	-4.917
Android	4G	Unlimited	deep (1000)	BBR	3+3 conns (mixed)	186.127	184.746	0.748	80.382	83.436	-3.66	28.665	26.938	6.411
Android	4G	Unlimited	deep (1000)	BBR2	1 connection	125.31	124.7	0.489	61.522	64.927	-5.245	24.338	25.919	-6.098
Android	4G	Unlimited	deep (1000)	BBR2	6 conns - same TCP	170.651	215.544	-20.828	72.558	91.246	-20.481	28.671	29.25	-1.979
Android	4G	Unlimited	deep (1000)	BBR2	3+3 conns (mixed)	205.562	209.723	-1.984	72.561	92.752	-21.769	28.409	29.387	-3.327
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	4G	50 Mbps	shallow (10)	Cubic	1 connection	105.091	71.187	47.627	40.301	32.473	24.105	24.755	25.189	-1.722
Android	4G	50 Mbps	shallow (10)	Cubic	6 conns - same TCP	125.91	56.104	124.42	44.06	33.27	32.433	25.631	25.543	0.343
Android	4G	50 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	121.535	58.269	108.57	42.779	33.06	29.398	25.424	25.271	0.606
Android	4G	50 Mbps	shallow (10)	BBR	1 connection	118.705	117.36	1.146	38.273	38.833	-1.441	23.985	24.228	-1.003
Android	4G	50 Mbps	shallow (10)	BBR	6 conns - same TCP	102.35	93.453	9.521	37.847	37.802	0.121	25.442	25.854	-1.593
Android	4G	50 Mbps	shallow (10)	BBR	3+3 conns (mixed)	86.108	86.611	-0.581	36.47	35.861	1.698	25.929	24.867	4.269
Android	4G	50 Mbps	shallow (10)	BBR2	1 connection	94.229	59.52	58.314	37.308	31.287	19.244	24.534	23.321	5.198
Android	4G	50 Mbps	shallow (10)	BBR2	6 conns - same TCP	119.013	78.11	52.365	40.511	32.772	23.614	25.622	24.733	3.593
Android	4G	50 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	122.326	82.234	48.753	39.972	31.515	26.835	25.151	23.24	8.221
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	5G	Unlimited	deep (1000)	Cubic	1 connection	73.973	85.398	-13.379	33.056	29.605	11.656	16.226	16.665	-2.63
Android	5G	Unlimited	deep (1000)	Cubic	6 conns - same TCP	128.462	127.007	1.146	41.184	45.224	-8.934	17.023	17.112	-0.519
Android	5G	Unlimited	deep (1000)	Cubic	3+3 conns (mixed)	95.851	92.807	3.279	34.991	33.369	4.861	15.972	16.605	-3.812
Android	5G	Unlimited	deep (1000)	BBR	1 connection	49.525	80.19	-38.241	26.599	31.953	-16.756	13.434	13.646	-1.55
Android	5G	Unlimited	deep (1000)	BBR	6 conns - same TCP	99.486	114.816	-13.352	38.611	39.718	-2.787	15.147	14.996	1.004
Android	5G	Unlimited	deep (1000)	BBR	3+3 conns (mixed)	102.242	105.931	-3.483	40.393	38.126	5.947	16.468	16.415	0.326
Android	5G	Unlimited	deep (1000)	BBR2	1 connection	84.938	88.312	-3.821	36.216	37.38	-3.114	17.592	18.989	-7.357
Android	5G	Unlimited	deep (1000)	BBR2	6 conns - same TCP	119.505	123.383	-3.143	43.289	45.006	-3.815	17.274	16.79	2.883
Android	5G	Unlimited	deep (1000)	BBR2	3+3 conns (mixed)	135.361	135.957	-0.439	49.032	54.719	-10.393	19.544	20.101	-2.771
Terminal	**Network**	**Limit**	**Buffer (pkts)**	**TCP**	**Connections**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**	**bequant TCP**	**Other TCP**	**Change**
Android	5G	100 Mbps	shallow (10)	Cubic	1 connection	35.195	33.157	6.148	20.826	17.101	21.783	13.649	13.161	3.713
Android	5G	100 Mbps	shallow (10)	Cubic	6 conns - same TCP	36.973	30.062	22.988	22.88	19.951	14.678	15.266	14.848	2.815
Android	5G	100 Mbps	shallow (10)	Cubic	3+3 conns (mixed)	40.384	33.382	20.977	23.244	19.833	17.201	15.725	14.913	5.444
Android	5G	100 Mbps	shallow (10)	BBR	1 connection	39.662	34.356	15.444	18.726	19.949	-6.131	12.647	13.118	-3.59
Android	5G	100 Mbps	shallow (10)	BBR	6 conns - same TCP	39.64	37.189	6.591	23.2	23.389	-0.81	16.083	15.96	0.766
Android	5G	100 Mbps	shallow (10)	BBR	3+3 conns (mixed)	37.986	35.775	6.18	23.065	21.638	6.596	15.147	15.021	0.839
Android	5G	100 Mbps	shallow (10)	BBR2	1 connection	48.699	34.198	42.4	25.932	22.231	16.649	17.596	17.233	2.106
Android	5G	100 Mbps	shallow (10)	BBR2	6 conns - same TCP	47.968	33.71	42.295	23.865	20.706	15.252	15.34	15.324	0.103
Android	5G	100 Mbps	shallow (10)	BBR2	3+3 conns (mixed)	44.673	39.35	13.529	22.944	18.83	21.849	14.636	13.785	6.175