Characterizing the Impact of Active Queue Management on Speed Test Measurements

Anonymous

No Institute Given

Abstract. Present day speed test tools measure peak throughput, but often fail to capture the user-perceived responsiveness of a network connection under load. Recently, platforms such as Ookla's Speedtest.net and Cloudflare have introduced metrics such as "latency under load" or "working latency" to fill this gap. Yet, the sensitivity of these metrics to basic network configurations such as Active Queue Management (AQM) remains poorly understood. In this work, we conduct an empirical study of the impact of AQM on speed test measurements in a laboratory setting. Using controlled experiments, we compare the distribution of throughput, latency, and latency under load measurements across different AQM schemes, including CoDel, FQ-CoDel and Stochastic Fair Queuing (SFQ). On comparing the results with a standard drop-tail baseline, we find that [TS: add the main punchline here.] These results highlight the critical role of AQM in shaping how emerging latency metrics should be interpreted, and underscore the need for careful calibration of speed test platforms before their results are used to guide policy or regulatory outcomes.

Keywords: Speed Test · Active Queue Management · Responsiveness · Bufferbloat

1 Introduction

Internet performance has historically been summarized using a single number: "speed" [1, 9, 3]. Despite the widespread utility, the user-perceived Quality of Experience (QoE) for many applications (e.g., video-conferencing, gaming, cloud collaboration) is governed less by peak bandwidth and more by latency under load. To address this, measurement providers have recently begun introducing "latency under load" (LUL) or "responsiveness" metrics, which attempt to capture how queuing delays increase during download and upload activity.

However, the interpretation and use of these metrics has not been standardized. For instance, Ookla defines "working latency" as the increase in round-trip time (RTT) under load compared to the unloaded RTT, measured during a speed test [2]. Apple uses a different metric, called round trips per minute (RPM) under load, which counts the number of round trips completed during a fixed time interval while the connection is saturated [11]. Further, these tests have been known to discard outliers that often correspond to glitches that users typically notice during real-time applications such as video conferencing and streaming [2]. As a result, users and regulators are left with incomplete pictures of what causes an Internet connection to be unresponsive, and how it can be mitigated. A central, unanswered question is how traditional metrics such as throughput and latency, and new metrics such as LUL and RPM, behave in the presence or absence of active queue management (AQM) algorithms such as FQ-CoDel, which were explicitly designed to maintain low latency under load [6].

In this paper, we investigate how the empirical distribution of modern speed test measurement results shifts when an AQM is deployed. Rather than reporting only typical throughput (e.g., mean) and latency values (e.g., 90^{th} percentile, median), we analyze full distributions: the tails, the spikes, and metrics similar to "glitches per minute" [2] that are most relevant to real-time applications. Our goal is to empirically characterize the difference between unmanaged queues and AQM-enabled network paths, and to highlight how this difference is (or is not) reflected in widely deployed measurement platforms. By doing so, we aim to inform both test designers and network operators of the gaps between the status quo of Internet measurement and the actual experience of end-users.

[TS: Methods summary and contributions go here.]

[SR: We also need some stronger justifications for why we are doing this study. Maybe something like: "While AQM has been widely studied in the context of TCP performance, its impact on speedtest measurements remains underexplored. Secondly, AQM deployment is steadily increasing (some numbers from Jason/Comcast article/any other source) Understanding this relationship is crucial for both network operators and end-users to accurately assess and improve their internet experience."]

2 Background

2.1 Active Queue Management (AQM)

AQM techniques have been an active area of research and deployment for the past few decades to reduce latency and bufferbloat [5] and to ensure fairness and coexistence among TCP flows on the shared network links. Traditional AQM algorithms have been built to run in conjunction with TCP congestion control algorithms, which rely only on packet loss as a signal for congestion.

To handle the bursty nature of TCP, these AQM techniques are equipped with large data buffers to prevent excessive packet drops due to these bursts. However, bufferbloat arises when queue length grows unbounded, specifically if the buffers are increasingly large, the packets end up being queued in much deeper queues, leading to excessive queuing latency. latency can often build up as a result of individual bufferbloats at multiple routers on the network path.

AQM technique such as Random Early Detection [4], CoDel [10] and FQ-CoDel [7] have been primarily designed to reduce latency and bufferbloat, by actively managing the queue lengths and dropping packets before the queue becomes full. Other techniques such as Stochastic Fair Queuing (SFQ) [8] and

Deficit Round Robin (DRR) [14] aim to provide fairness among competing flows by ensuring equal bandwidth allocation. More recently, AQM techniques such as Proportional Integral controller Enhanced (PIE) [12] and Low Latency, Low Loss, Scalable Throughput (L4S) [13] have been proposed to provide low latency and high throughput for modern applications such as video streaming and online gaming.

[SR: Should mention something about L4S here? It is positioned as a new AQM which tries to universally solve bufferbloat and fairness but we don't evaluate it.]

2.2 Speedtest Measurement Tools

[SR: Add details for each of these tools.]

- M-Lab NDT: The Measurement Lab's Network Diagnostic Tool (NDT) is a widely used tool for measuring network performance, including bandwidth, latency, and packet loss.
- Ookla Speedtest: Ookla's Speedtest is a popular web-based tool that provides users with real-time information about their internet connection speed, including download and upload speeds.
- Apple Speedtest: Apple's Speedtest app is designed for iOS devices and provides users with a simple way to test their internet connection speed.

3 Motivation (Current speedtest tools are not AQM-aware punchline)

References

1. Bauer, S., Clark, D.D., Lehr, W.: Understanding broadband speed measurements. TPRC (2010)
2. Dave Täht: The new features and flaws of speedtest.net, ookla and cloudflare. https://blog.cerowrt.org/post/speedtests/, accessed: 2025-10-01
3. Feamster, N., Livingood, J.: Internet speed measurement: Current challenges and future recommendations (2019), https://arxiv.org/abs/1905.02334
4. Floyd, S., Jacobson, V.: Random early detection gateways for congestion avoidance. IEEE/ACM Transactions on Networking **1**(4), 397–413 (1993). https://doi.org/10.1109/90.251892
5. Gettys, J., Nichols, K.: Bufferbloat: Dark buffers in the internet: Networks without effective aqm may again be vulnerable to congestion collapse. Queue **9**(11), 40–54 (nov 2011). https://doi.org/10.1145/2063166.2071893, https://doi.org/10.1145/2063166.2071893
6. Høiland-Jørgensen, T.: Bufferbloat and beyond
7. Høiland-Jørgensen, T., McKenney, P., dave.taht@gmail.com, Gettys, J., Dumazet, E.: The Flow Queue CoDel Packet Scheduler and Active Queue Management Algorithm. RFC 8290 (Jan 2018). https://doi.org/10.17487/RFC8290, https://www.rfc-editor.org/info/rfc8290

8. McKenney, P.E.: Stochastic fairness queuing. In: IEEE INFOCOM'90 Proceedings. pp. 733–740. The Institute of Electrical and Electronics Engineers, Inc., San Francisco (June 1990)
9. Midoglu, C., Wimmer, L., Lutu, A., Alay, O., Griwodz, C.: Monroe-nettest: A configurable tool for dissecting speed measurements in mobile broadband networks (2018), https://arxiv.org/abs/1710.07805
10. Nichols, K., Jacobson, V.: Controlling queue delay. Commun. ACM **55**(7), 42–50 (jul 2012). https://doi.org/10.1145/2209249.2209264, https://doi.org/10.1145/2209249.2209264
11. Paasch, C., Meyer, R., Cheshire, S., Hawkins, W.: Responsiveness under Working Conditions. Internet-Draft draft-ietf-ippm-responsiveness-07, Internet Engineering Task Force (Jul 2025), https://datatracker.ietf.org/doc/draft-ietf-ippm-responsiveness/07/, work in Progress
12. Pan, R., Natarajan, P., Piglione, C., Prabhu, M.S., Subramanian, V., Baker, F., VerSteeg, B.: Pie: A lightweight control scheme to address the bufferbloat problem. In: 2013 IEEE 14th International Conference on High Performance Switching and Routing (HPSR). pp. 148–155 (2013). https://doi.org/10.1109/HPSR.2013.6602305
13. Schepper, K.D., Albisser, O., Tilmans, O., Briscoe, B.: Dual queue coupled aqm: Deployable very low queuing delay for all (2022), https://arxiv.org/abs/2209.01078
14. Shreedhar, M., Varghese, G.: Efficient fair queueing using deficit round robin. In: Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication. p. 231–242. SIGCOMM '95, Association for Computing Machinery, New York, NY, USA (1995). https://doi.org/10.1145/217382.217453, https://doi.org/10.1145/217382.217453