QUIC compared to TCP+TLSv1.2

Miguel Gordo García, Nikolaos Tatsis

KTH Royal Institute of Technology

October 27, 2015

Contents

Contents

What is QUIC?

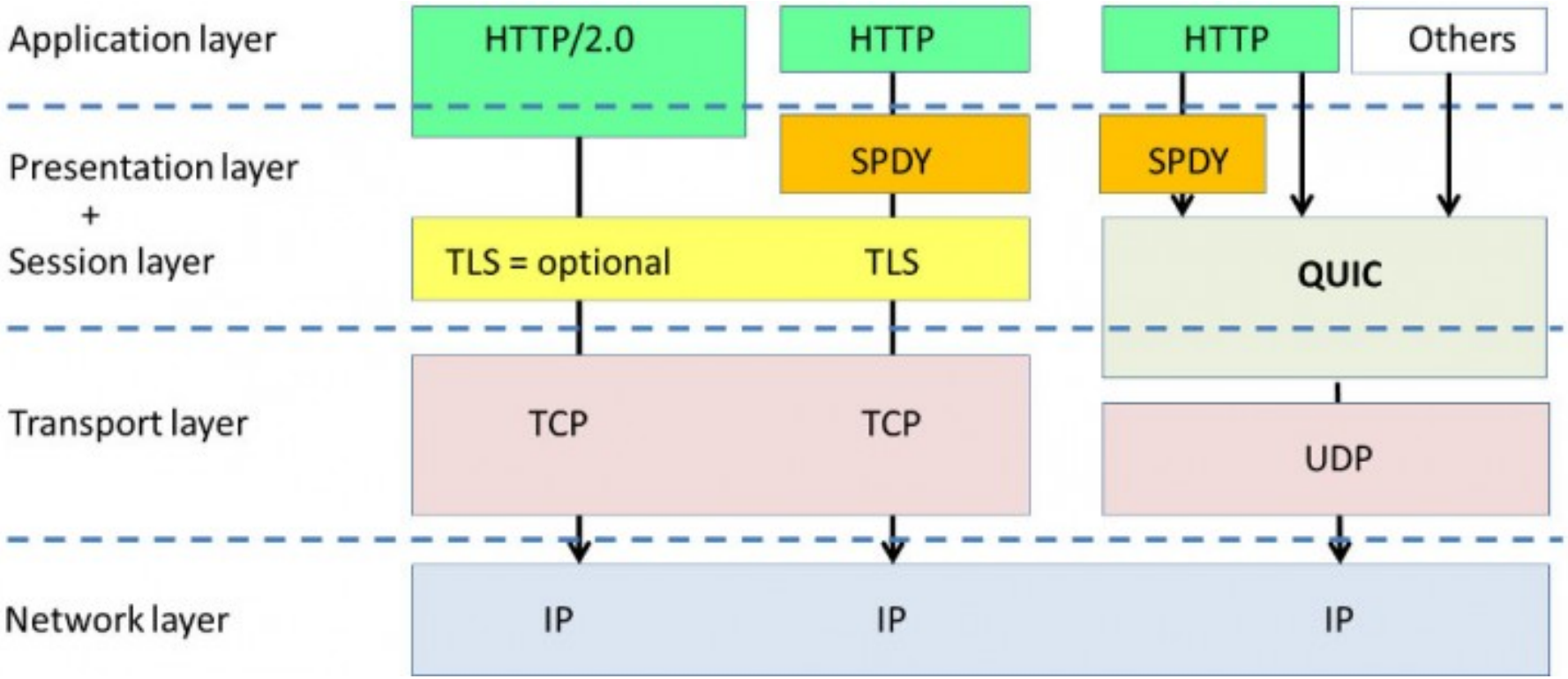

Figure 1: Network Stack

What is QUIC?

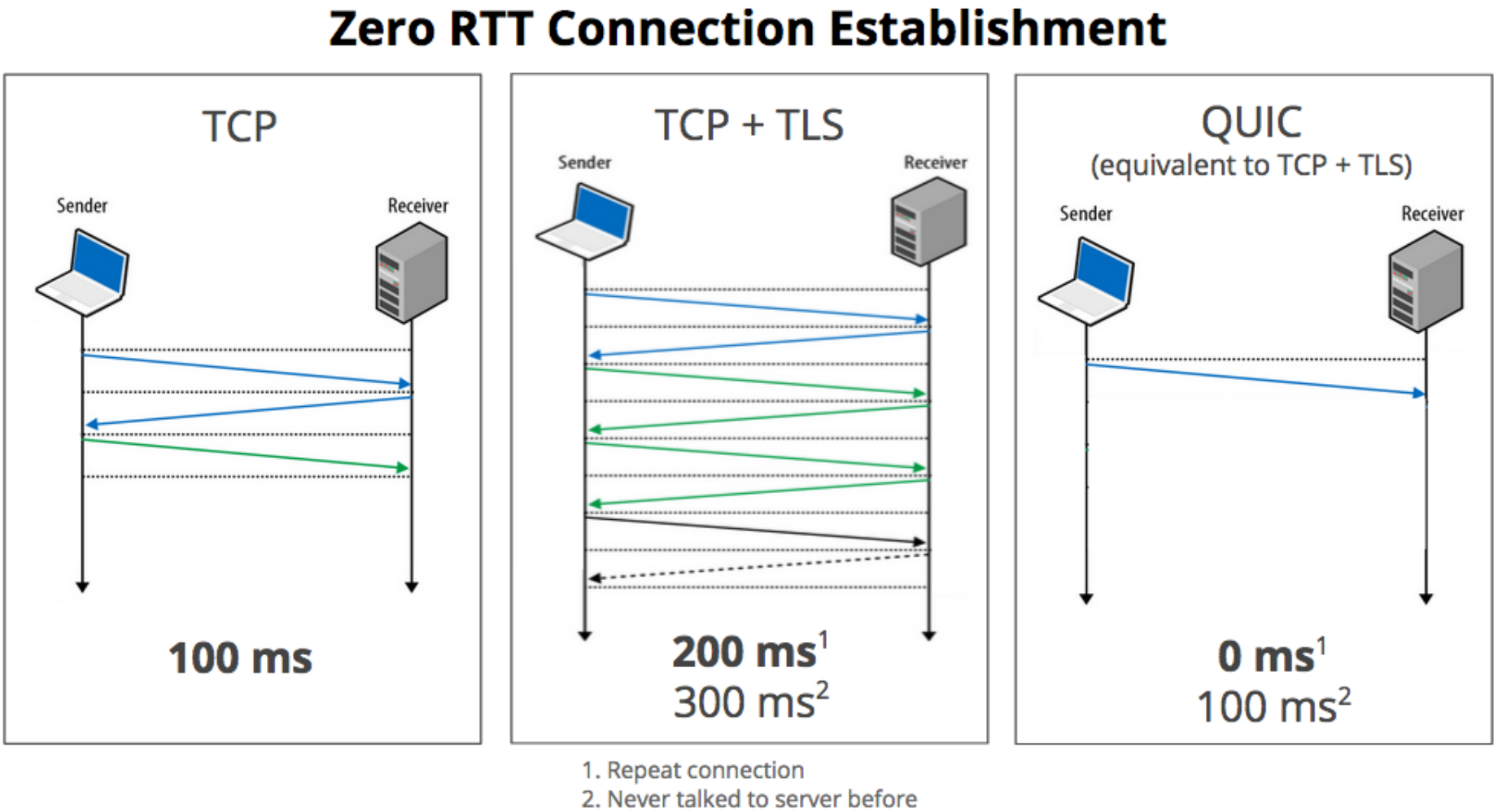

Figure 2: RTT Times between TCP, TCP+TLS and QUIC

Goals

Analyze performance of TCP and QUIC in terms of:

- **Total transfer time**
- **Average Bandwidth used**
- **Overhead in bytes**

Contents

Experimental Setup

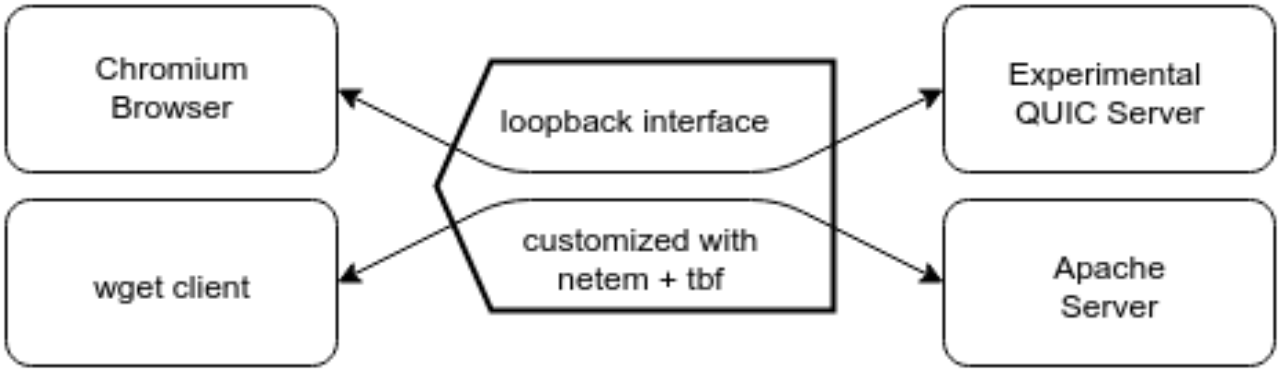

Figure 3: experimental setup used

- Chromium Project experimental QUIC Server - Chromium as client.
- Apache2 Webserver configured for TLS1.2 - wget as client.
- tc netem + tbf configuration of local loopback interface.
- Data captured with tcpdump, packet information stored.
- Packet data analyzed with Python library matplotlib.

Test Variables

- All possible combinations of Network Delay, Packet Loss and Available Bandwidth performed.
- All Network Jitter values tested with a set Delay of 60 ms and a Bandwidth of 100 Mbps.
- Sudden Bandwidth decrease tests to 10% of Available Bandwidth, for all Network Delay values with a bandwidth of 40Mbps.
- All tests performed five times, sudden bandwidth decrease tests performed ten times.

Table 1: Test variables.

Network Delay (ms)	10, 20, 40, 60, 80, 100, and 120
Network Jitter (ms)	0, 10, 20, 40 and 50
Packet loss (%)	0, 2.5 and 5
Available Bandwidth (Mbps)	1, 40 and 100

Contents

Results

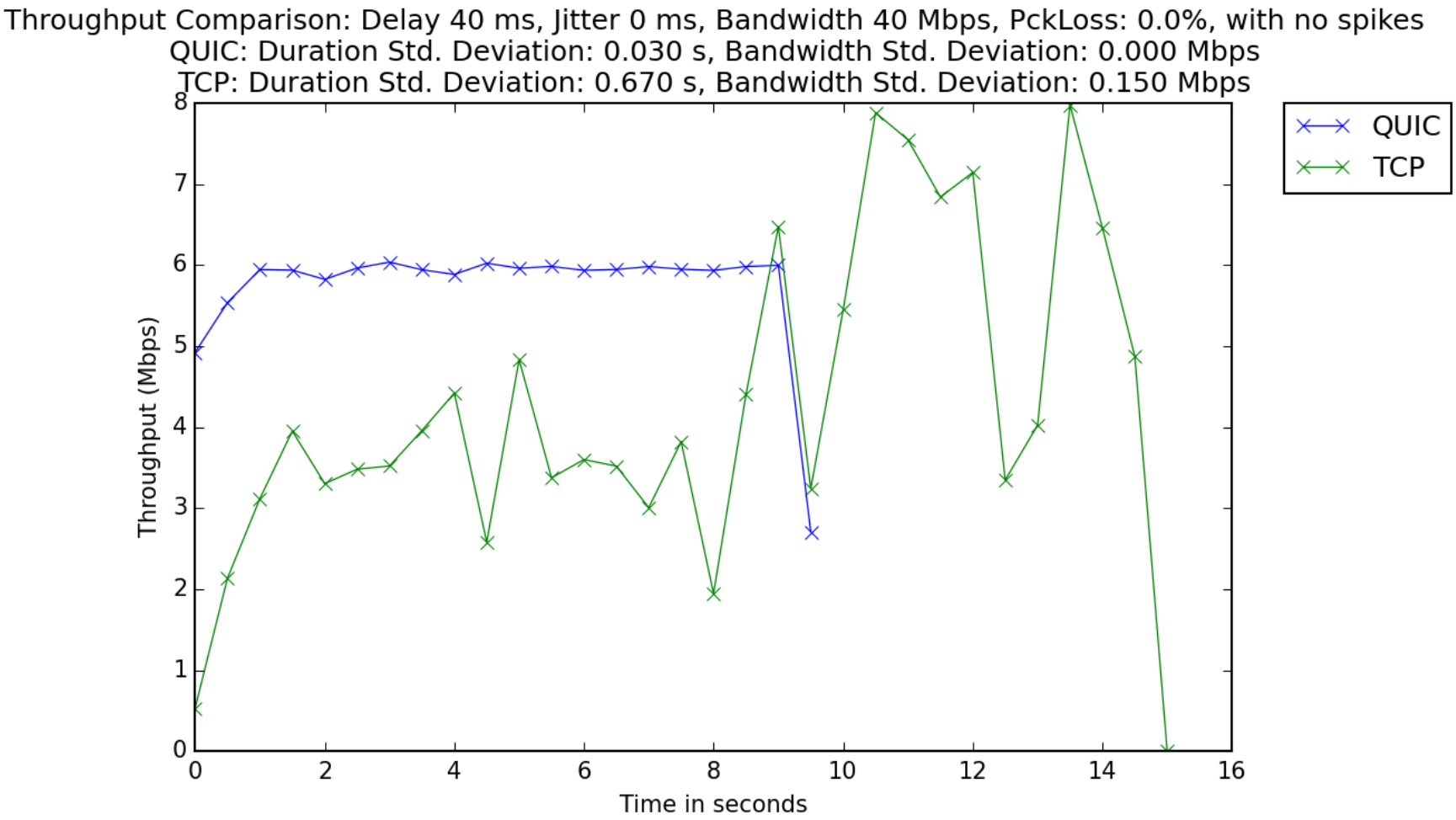

Figure 4: Time Series under favorable conditions

Results

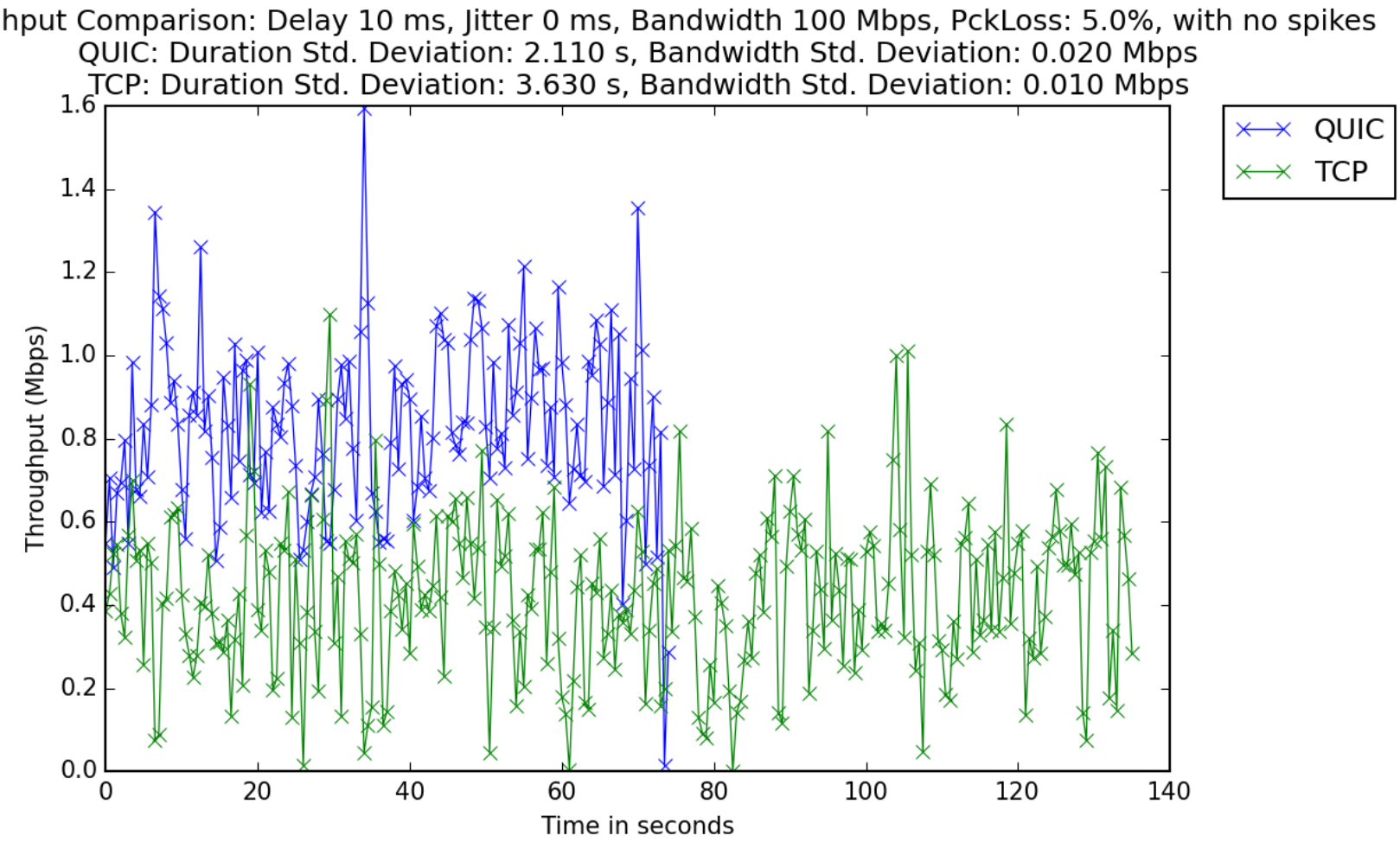

Figure 5: Time Series under heavy packet loss

Results

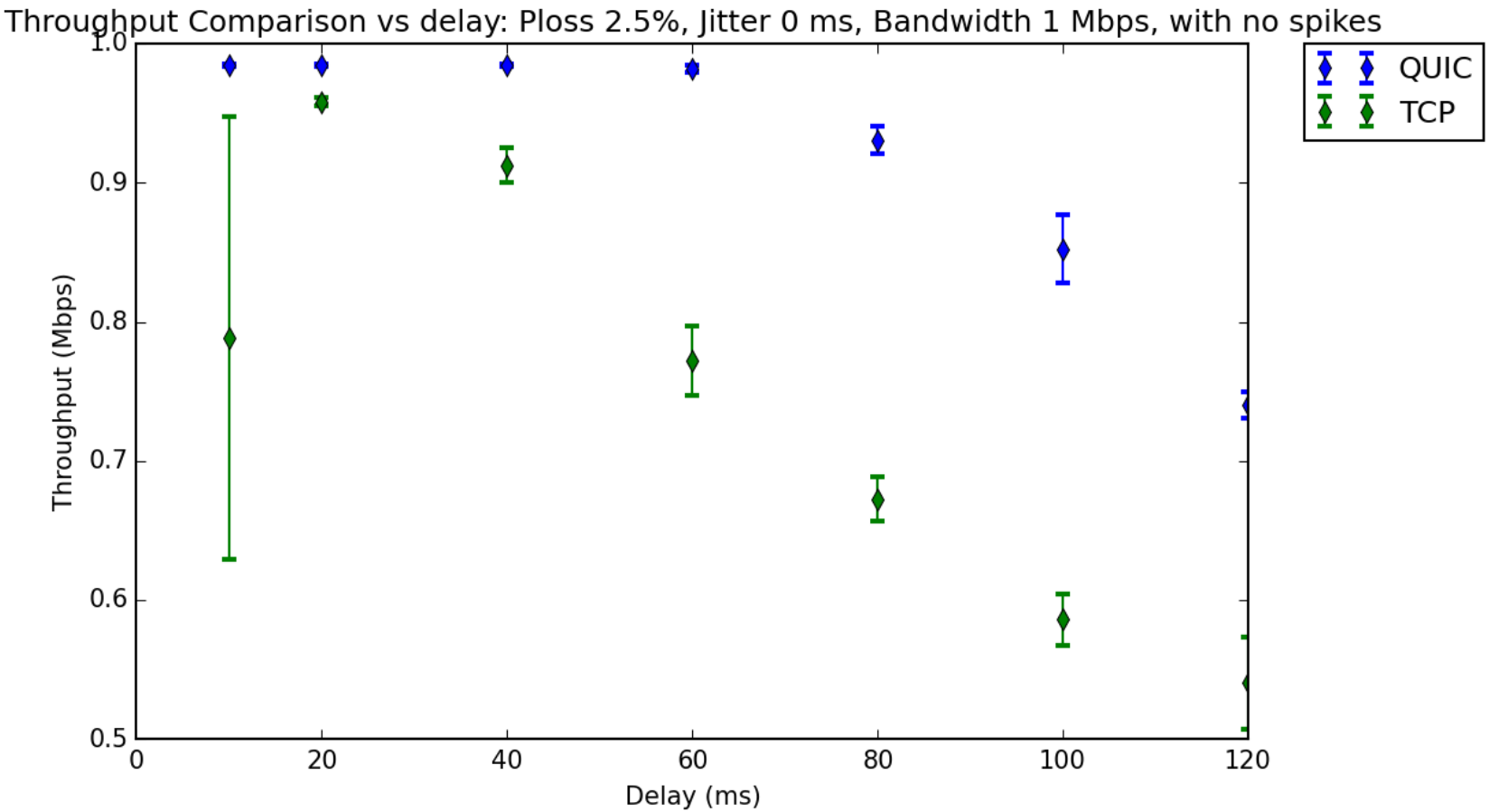

Figure 6: Throughput against Delay

Results

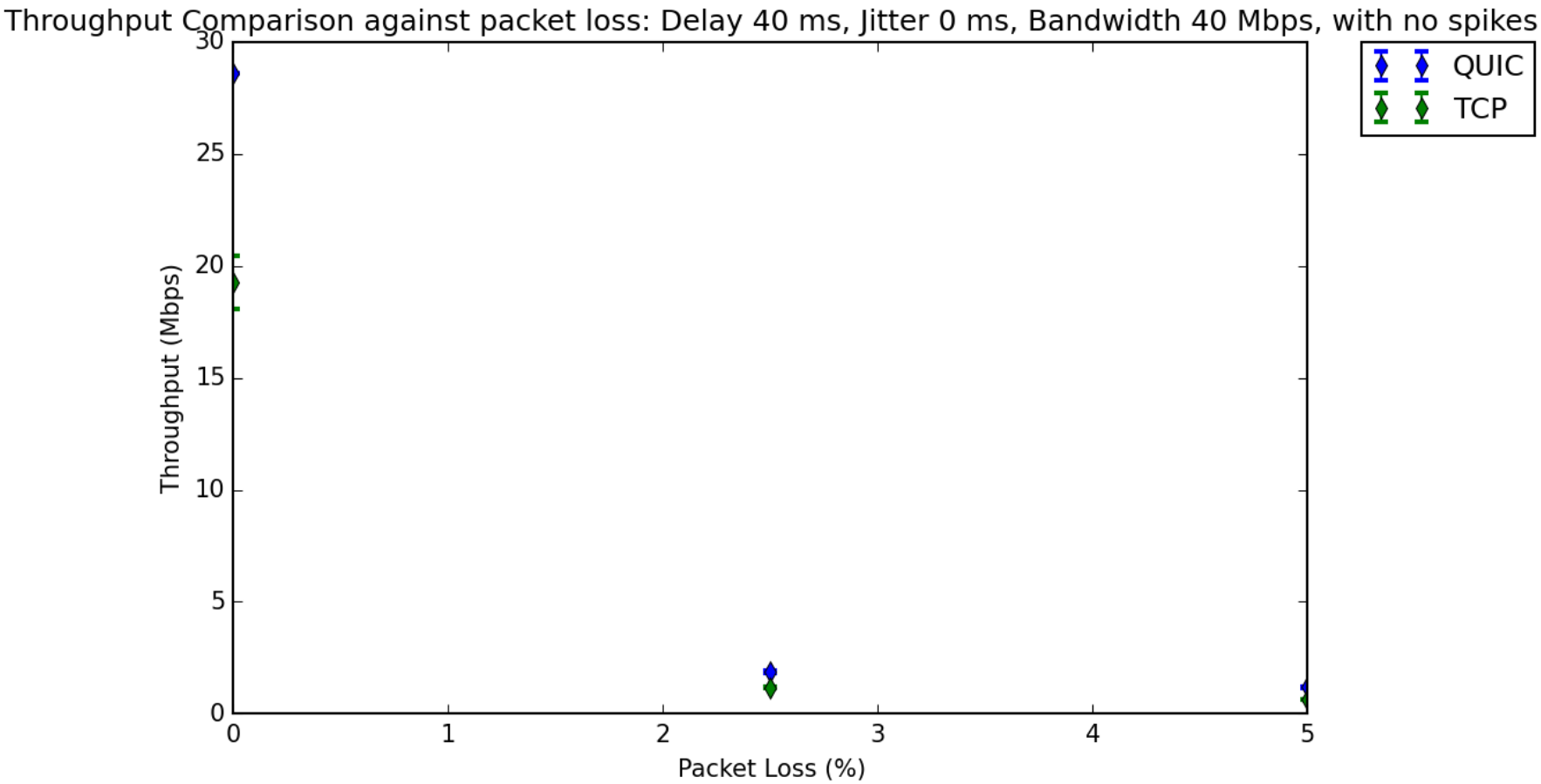

Figure 7: Throughput against Packet Loss

Results

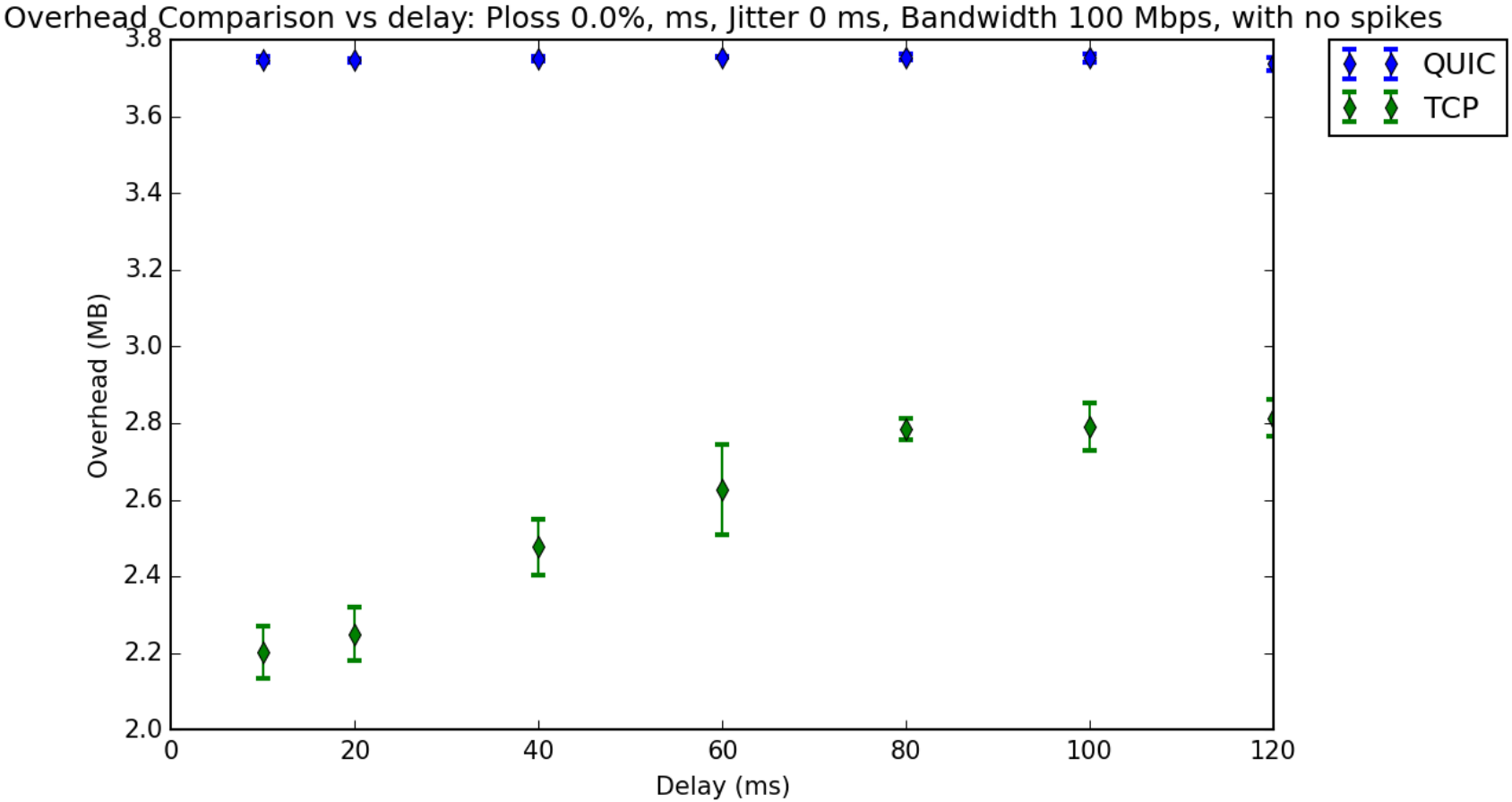

Figure 8: Overhead (in bytes) compared against Delay

Results

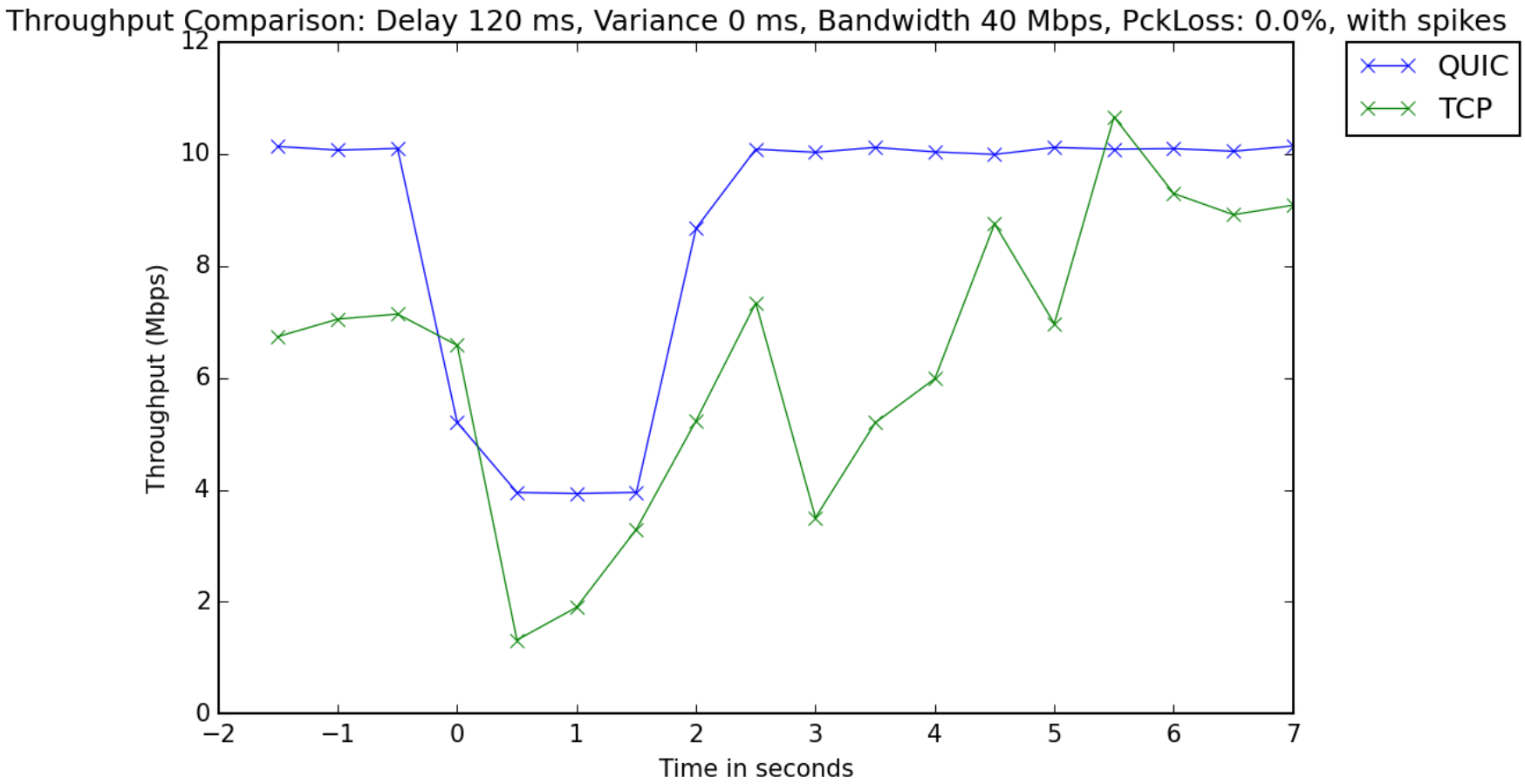

Figure 9: Bandwidth against a decrease in bandwidth

Results

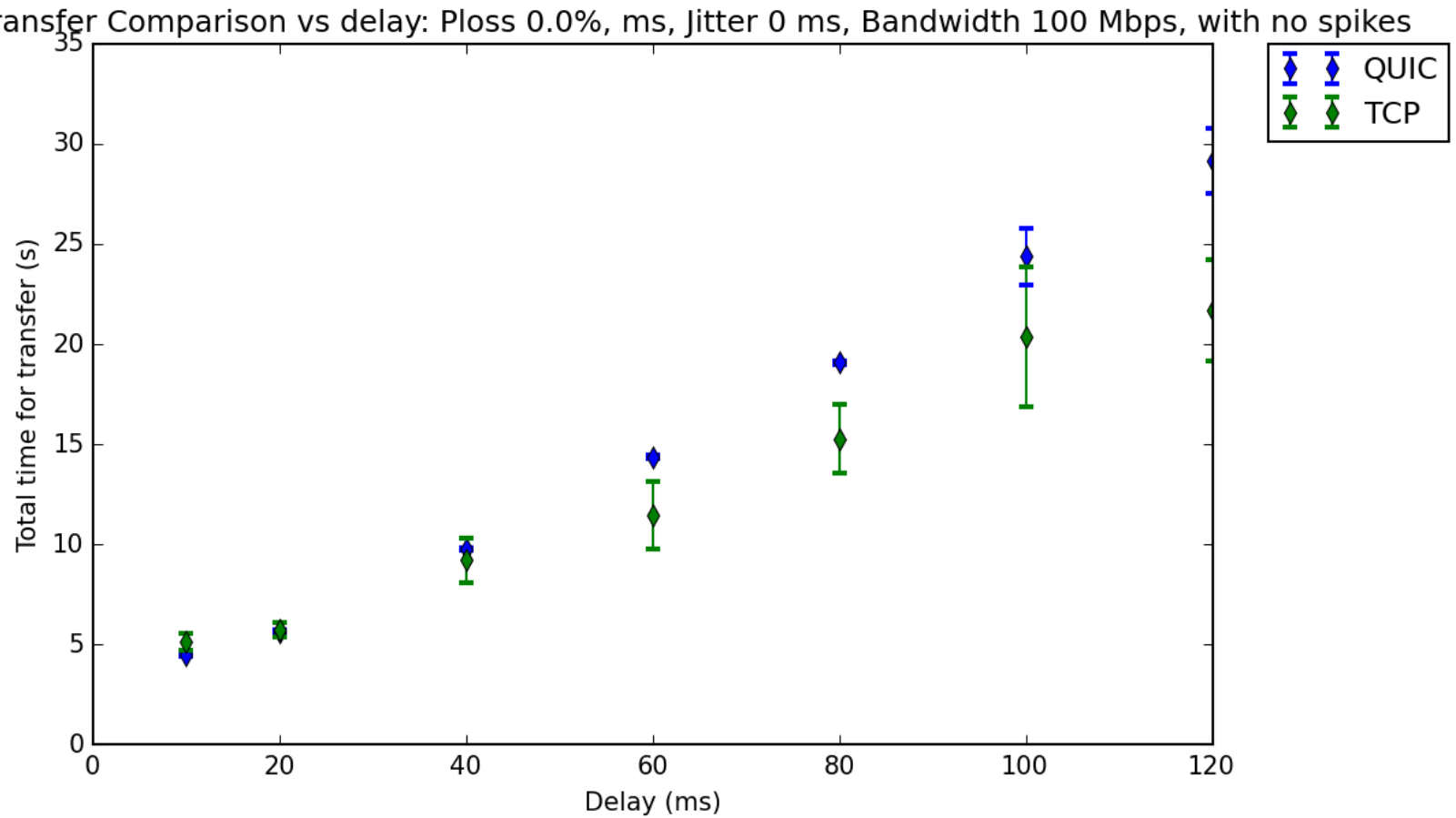

Figure 10: Throughput against Delay

Results

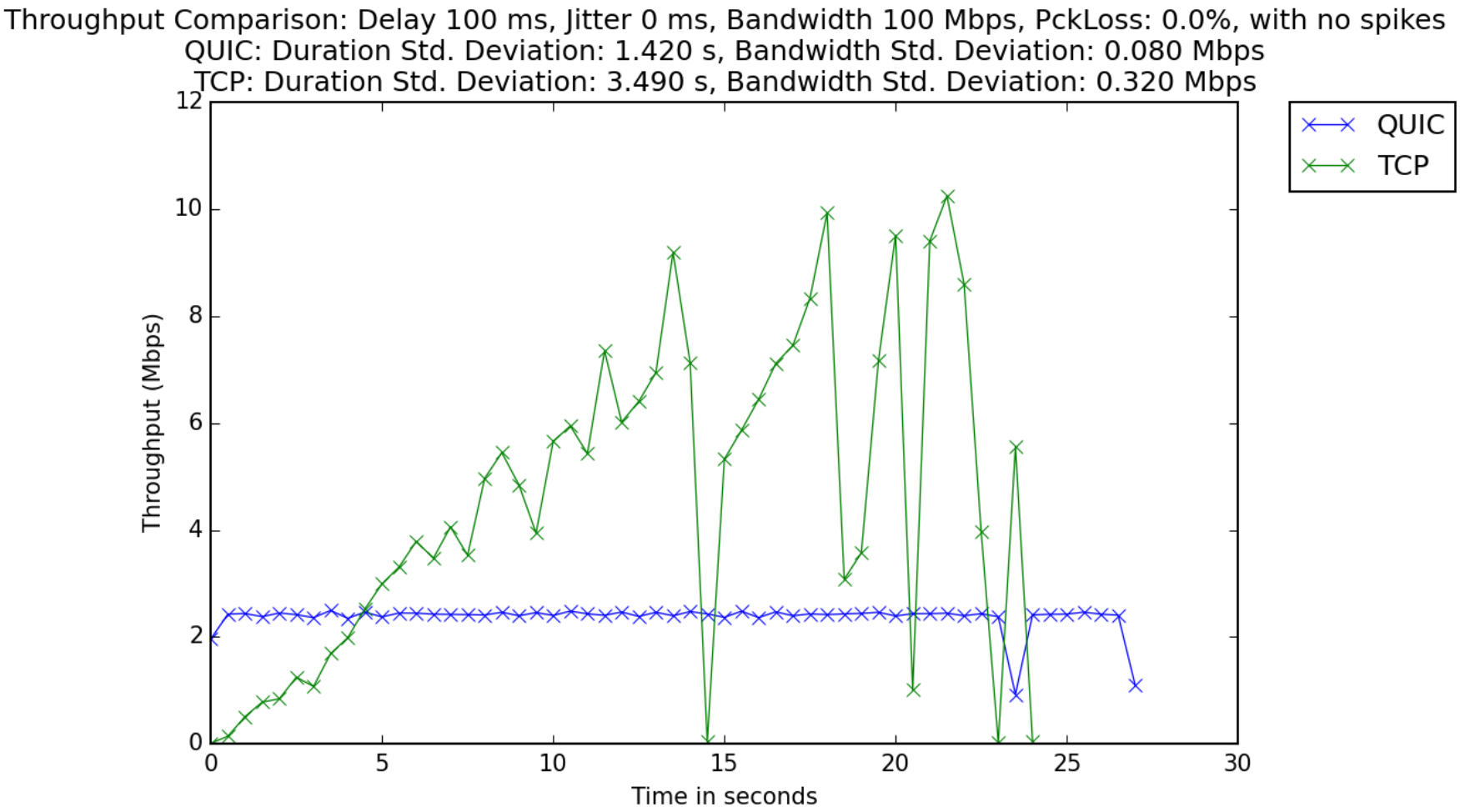

Figure 11: QUIC hits bandwidth threshold

Results

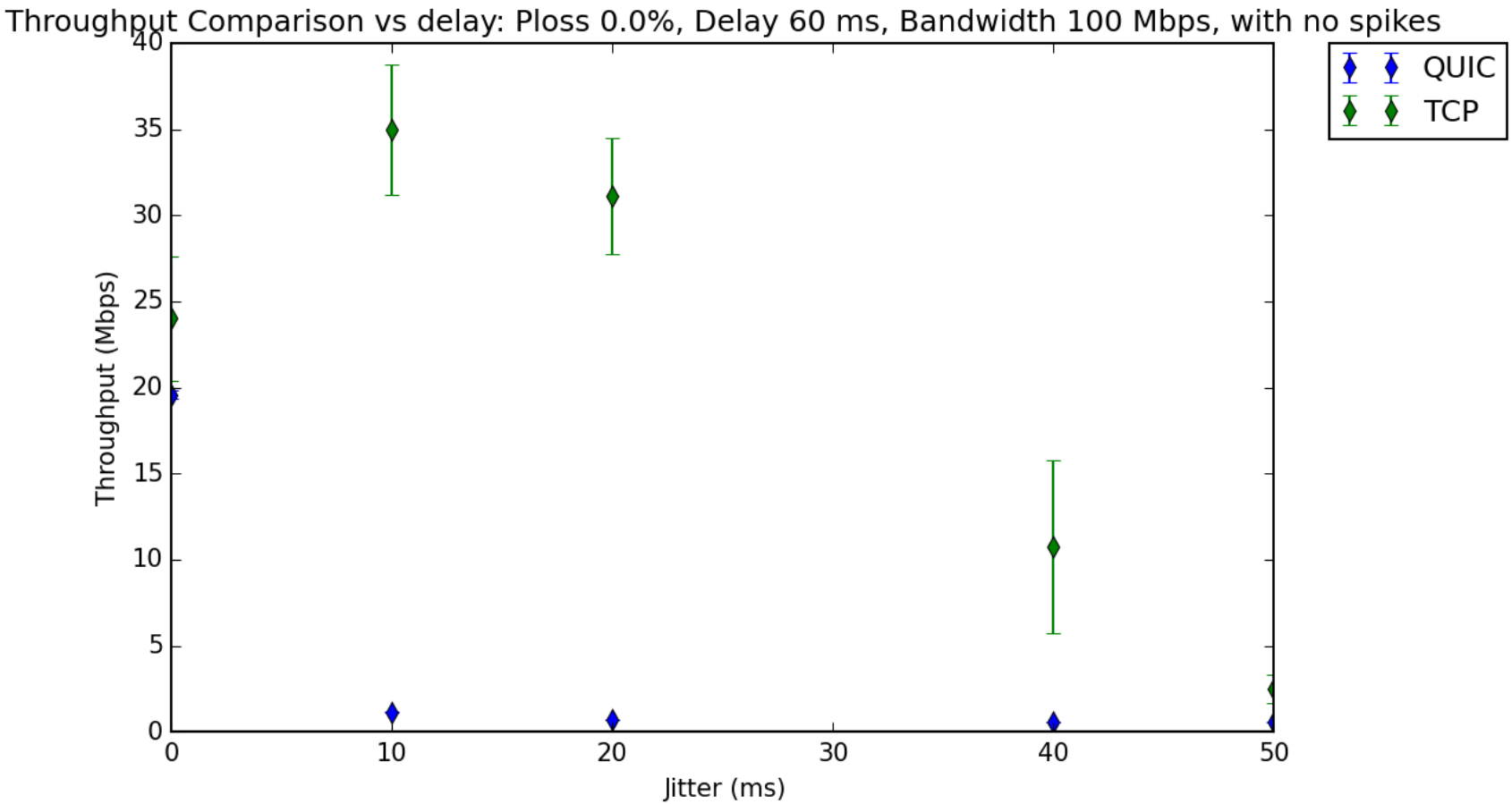

Figure 12: Throughput against Jitter

Contents

Conclusions

- At the cost of higher overhead, QUIC outperforms TCP under favorable conditions in terms of time for transfer and average bandwidth used.
- Under packet loss, QUIC also surpasses TCP
- QUIC recovers more quickly from drops in bandwidth
- With high delays and jitter, QUIC hits a bandwidth threshold and TCP performs better. Causes might lie in current release and not in the protocol itself.

Future Work

Future work centered in:

- Repeating this test suite with future and final versions of QUIC to compare performance upgrades
- Designing new tests to measure fairness when sharing bandwidth with other QUIC/TCP flows
- Stream Multiplexing in QUIC: Evaluate advantages over loading HTTP pages, for example.
- FEC (Forward Error Correction): Benefits of using FEC in packet loss scenarios.

Contents

References

J. Iyengar and I. Swett, 'QUIC: A UDP-Based Secure and Reliable Transport for HTTP/2', Internet Engineering Task Force: Active Internet Drafts, Jul. 2015. Available: https://tools.ietf.org/html/draft-tsvwg-quic-protocol-01

I. Swett and J. Iyengar, 'QUIC Loss Recovery And Congestion Control', Internet Engineering Task Force: Active Internet Drafts, Jun. 2015. Available: https://tools.ietf.org/html/draft-tsvwg-quic-loss-recovery-00

A. Langley, W. Chang, 'QUIC Crypto', The Chromium Projects, Jul. 2015. Available: http://www.rfc-editor.org/rfc/rfc5246.txt

J. Postel, 'Transmission Control Protocol', Internet Request for Comments, vol. RFC 793, Sep. 1981. Available: https://docs.google.com/document/d/1g5nIXAIkN_Y-7XJW5K45IblHd_L2f5LTaDUDwvZ5L6g/edit

T. Dierks, E. Rescorla, 'The Transport Layer Security (TLS) Protocol Version 1.2', Internet Request for Comments, vol. RFC 5246, Aug. 2008. Available: http://www.rfc-editor.org/rfc/rfc5246.txt

M. Gordo Garcia, N. Tatsis, 'QUIC compared to TCP+TLSv1.2', Oct. 2015, GitHub repository, Available: https://www.github.com/mgordo/quic-vs-tcp-tls

Questions?