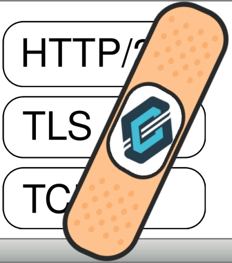
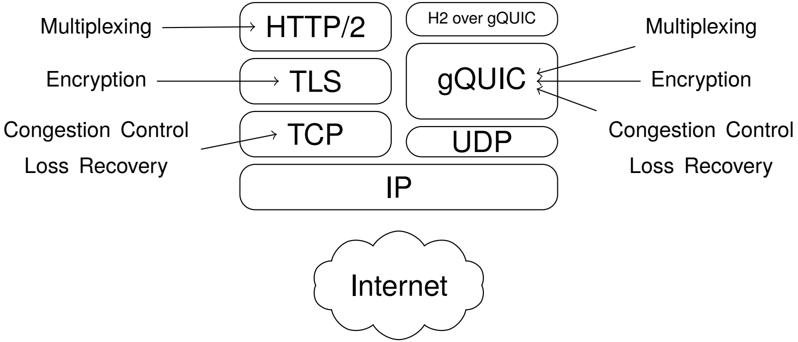
A Performance Perspective on Web Optimized Protocol Stacks: TCP+TLS+HTTP/2 vs. QUIC



Konrad Wolsing, Jan Rüth, Klaus Wehrle, Oliver Hohlfeld

Evolution of Web-stacks



- **Evolvability over time (no ossification)**
- **No head-of-line blocking**
- **0-RTT connection establishment**

Related Work (QUIC vs. TCP)

HTTP over UDP: an Experimental Investigation of QUIC

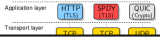
Gaetano Carlucci
Politecnico di Bari &
Quavlive, Italy
gaetano.carlucci@poliba.it

Luca De Cicco
Politecnico di Bari &
Quavlive, Italy
l.decicco@poliba.it

Saverio Mascolo
Politecnico di Bari &
Quavlive, Italy
mascolo@poliba.it

ABSTRACT

This paper investigates "Quick UDP Internet Connections" (QUIC), which was proposed by Google in 2012 as a reliable protocol on top of UDP in order to reduce Web Page re- trieval time. We first check, through experiment can be safely deployed in the Internet and the ate the Web page load time in comparison with HTTP. We have found that QUIC reduces the retrieval time with respect to HTTP in case of



QUIC: Better For What And For Whom?

Sarah Cook, Bertrand Mathieu, Patrick Truong and Isabelle Hamchaoui
Orange Labs
Lannion, France
Email: {sarah.cook,bertrand2.mathieu,patrick.truong,isabelle.hamchaoui}@orange.com

Abstract—**Many applications nowadays use HTTP. HTTP/2, standardised in February 2015, is an improvement of HTTP/1.1.** ... Chromium codebase to evaluate the CUBIC congestion control algorithm implemented in QUIC with the one implemented

Does QUIC make the Web faster ?

Prasenjeet Biswal
Department of Computer Science
University of Houston

Omprakash Gnawali
Department of Computer Science
University of Houston

Abstract—**Increase in size and complexity of web pages has challenged the efficiency of HTTP. Recent developments to** ... the latency due to round trips due to client request, and (5) request prioritization.

QUIC and TCP: A Performance Evaluation

Késsia Nepomuceno
Centro de Informática
Uni. Federal de Pernambuco
Recife, Brazil
ktcn@cin.ufpe.br

Igor Nogueira de Oliveira
Centro de Informática
Uni. Federal de Pernambuco
Recife, Brazil
ino@cin.ufpe.br

Rafael Roque Aschoff
Dept. de Ensino
Inst. Federal de Pernambuco
Palmares, Brazil
rafael.roque@palmares.ifpe.edu.br

Daniel Bezerra
Centro de Informática
Uni. Federal de Pernambuco
Recife, Brazil
db@cin.ufpe.br

Maria Silvia Ito
Centro de Informática
Uni. Federal de Pernambuco
Recife, Brazil
msi@cin.ufpe.br

Wesley Melo
Centro de Informática
Uni. Federal de Pernambuco
Recife, Brazil
wdbm@cin.ufpe.br

Djamel Sadok
Centro de Informática
Uni. Federal de Pernambuco
Recife, Brazil
jamel@gprt.ufpe.br

Géza Szabó
Traffic Laboratory
Ericsson Research
Budapest, Hungary
geza.szabo@ericsson.com

(PLT) impacts on the way a user navigates and influences the time it spends at a site. Users who are frustrated by a slow-loading site, tend to visit the site once, leave to never return [2]. Moreover, PLT seems to have an impact on how well users recommend websites to others [3]. In fact, some

Taking a Long Look at QUIC

An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols

Arash Molavi Kakhki
Northeastern University
arash@ccs.neu.edu

Samuel Jero
Purdue University
sjero@purdue.edu

David Choffnes
Northeastern University
choffnes@ccs.neu.edu

Cristina Nita-Rotaru
Northeastern University
c.nitarotaru@neu.edu

Alan Mislove
Northeastern University
amislove@ccs.neu.edu

ABSTRACT

Google's QUIC protocol, which implements TCP-like properties at the application layer atop a UDP transport, is now used by the vast

1 INTRODUCTION

Transport-layer congestion control is one of the most important elements for enabling both fair and high utilization of Internet

How quick is QUIC?

Péter Megyesi, Zsolt Krämer, Sándor Molnár
High Speed Networks Lab., Dept. of Telecomm. and Media Informatics,
Budapest University of Technology and Economics, Budapest, Hungary
E-mail: {megyesi, kramer, molnar}@tmit.bme.hu

Abstract—**HTTP has been the protocol for transferring web traffic over the Internet since the 90s. However, over the past** ... to develop a new protocol called QUIC (Quick UDP Internet Connections) [3] in 2013 which uses UDP (User Datagram

...en QUIC Meets TCP: an Experimental Study

Yajun Yu[*,†], Mingwei Xu[*,†], Yuan Yang[*,†]
[*]Department of Computer Science and Technology, Tsinghua University, China
[†]Tsinghua National Laboratory for Information Science and Technology
Email: yu-yj16@mails.tsinghua.edu.cn; xmw@cernet.edu.cn; yyang@csnet1.cs.tsinghua.edu.cn;

Abstract—**Recently, QUIC proposed by Google has drawn great attentions due to several attractive features which improve** ... ence model in Fig. 1. In particular, the UDP-based connection management and the congestion control algorithms belong to

Related Work (QUIC vs. TCP)

- **Unoptimized TCP stacks**
  - QUIC is optimized for web performance
  - TCP can be tuned too
  - Related work does not tune TCP
- **Connection establishment**
  - QUIC requires 0–1 RTT
  - TCP+TLS usually 2 RTT
- **User-centered metrics**
  - research focuses on PLT
  - PLT is not suited for user perception

Our Goal

Reproducible, user-centered performance evaluation on eye-to-eye level between TCP+TLS+HTTP/2 and QUIC.

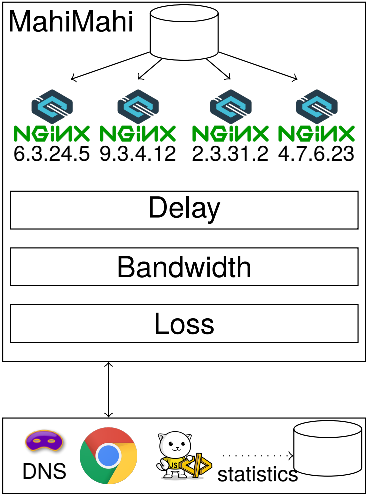
Achieving comparability

- **TCP + TLS + HTTP**
  - ✔ Initial window 32
  - ✔ Packet pacing
  - ★ 1-RTT handshake
  - ★ Use TLS1.3 without 0-RTT
  - ✔ Use only HTTP/2
  - ✖ Head-of-line blocking
- **QUIC**
  - ✔ Initial window 32
  - ✔ Packet pacing
  - ★ no 0-RTT connection setup
  - ✔ No encryption overhead
  - ✔ HTTP/2
  - ✔ No head-of-line blocking

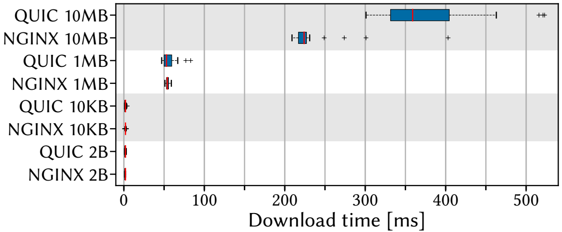
- **Protocol settings**

TCP	**Stock TCP (Linux 4.18): IW10, Cubic**
TCP+	**IW 32, Pacing, tuned network buffers, no slow start after idle, Cubic**
TCP+BBR	**TCP+, but with BBR as congestion control**
QUIC	**Google QUIC Version 43: IW32, Pacing, Cubic**
QUIC+BBR	**QUIC, but with BBR as congestion control**

Testbed



- **Server speed test**

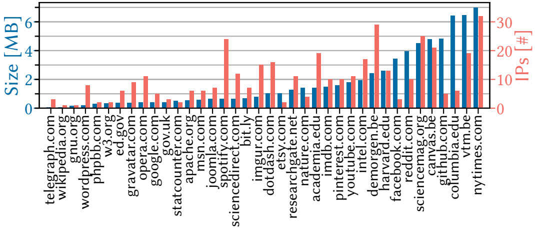


- **Network configuration**

	DSL[2]	LTE[2]	3G[1]	DA2GC[7]	MSS[7]
Uplink	5 Mbps	2.8 Mbps	3.54 Mbps	.468Mbps	1.89Mbps
Downlink	25 Mbps	10.5 Mbps	3.54 Mbps	.468Mbps	1.89Mbps
RTT	24ms	74ms	94ms	262ms	761ms
Loss	0.0%	0.0%	.048%	3.3%	6.0%
Queue size	12ms	200ms	200ms	200ms	200ms

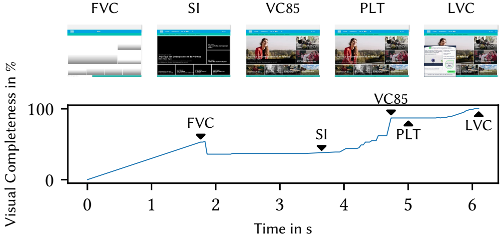
Measurement

- **Replayed Websites**



  - ▶ 38 websites [8] from Alexa and Moz lists

- **Visual Metrics**

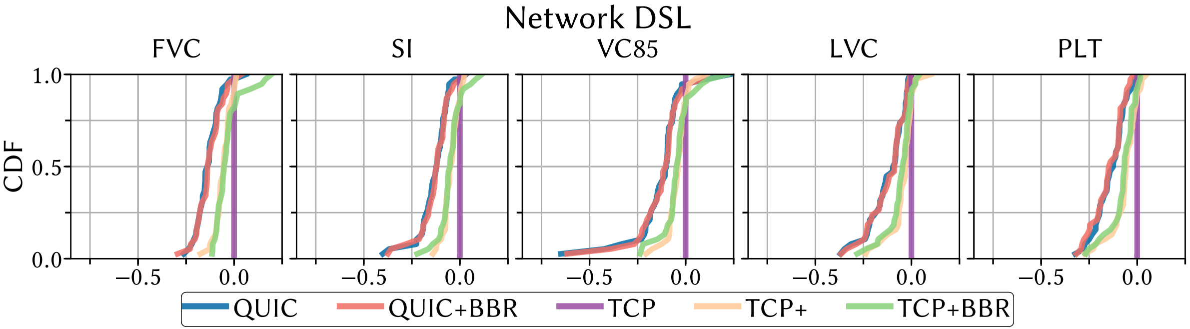


  - ▶ Only PLT not above-the-fold

$$performance\ gain_{QUIC}^{TCP} = \frac{\overline{X}_{QUIC} - \overline{X}_{TCP}}{\overline{X}_{TCP}}$$

$$\frac{0.5s_{QUIC} - 0.75s_{TCP}}{0.75s_{TCP}} = \text{-}0.333$$

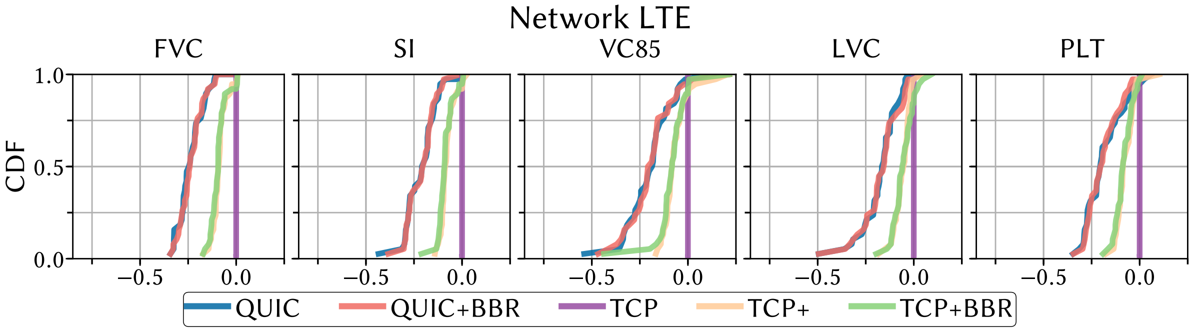
Performance gain DSL



(Downlink 25Mbps, Uplink 5 Mbps, RTT 24ms, Loss 0.0%, [2])

- **Tuning $\Rightarrow$ increased performance**
- **QUIC outperforms even tuned TCP**
- **Steep curves: website size/structure has low impact**

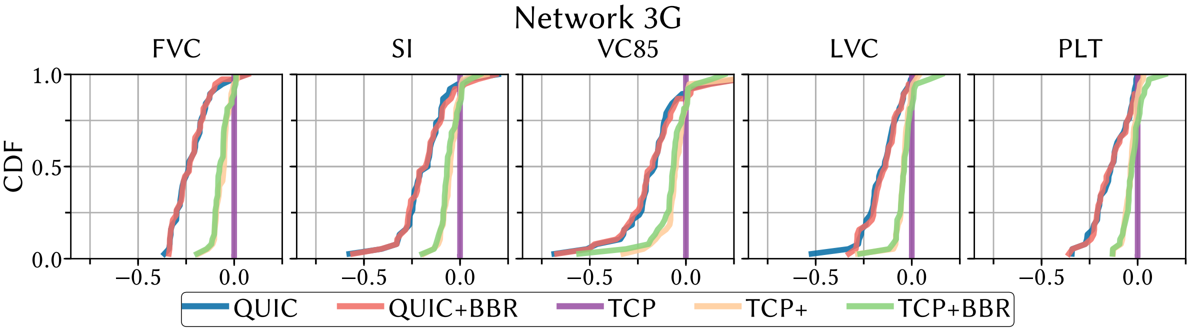
Performance gain LTE



(Downlink 10.5Mbps, Uplink 2.8 Mbps, RTT 74ms, Loss 0.0%, [2])

- **Results similar to DSL**
- **But differences are slightly larger**

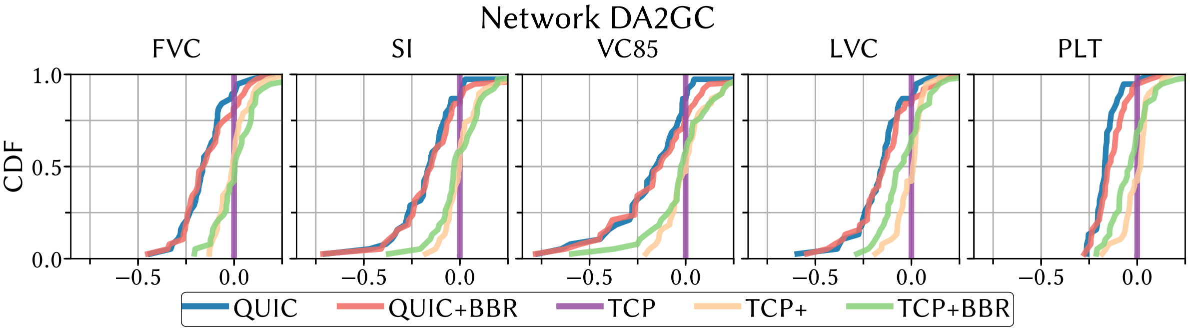
Performance gain 3G



(Downlink 3.54Mbps, Uplink 3.54 Mbps, RTT 94ms, Loss $.048\%$, [1])

- **Tuning impact reduced**
- **Variability rises**

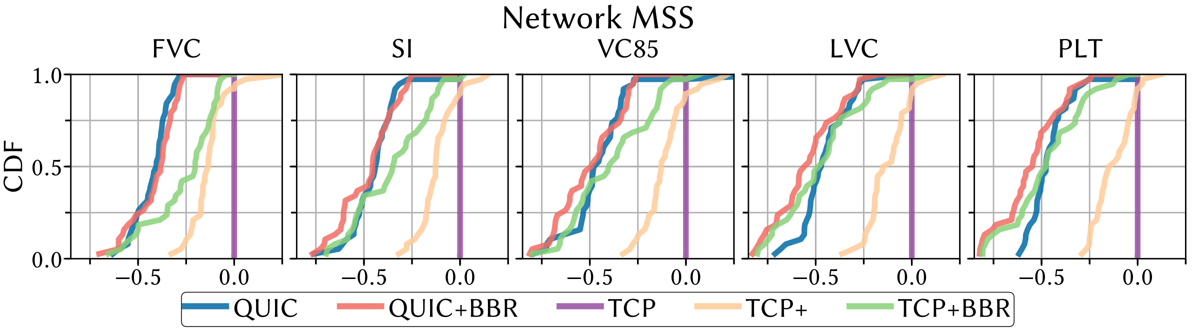
Performance gain DA2GC



(Downlink .468Mbps, Uplink .468 Mbps, RTT 262ms, Loss 3.3%, [7])

- **Tuning becomes a coin toss**
- **QUIC performs still good**
  - No head-of-line blocking
  - Larger SACK ranges

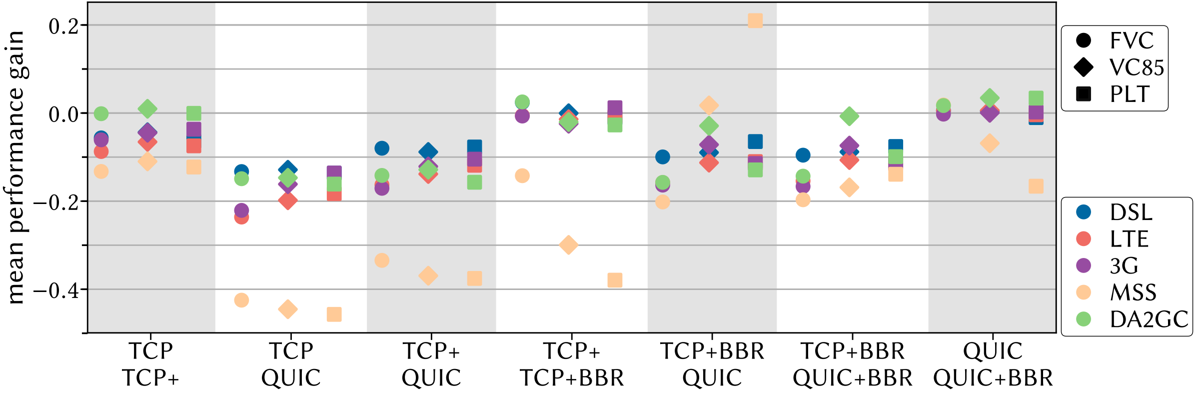
Performance gain MSS

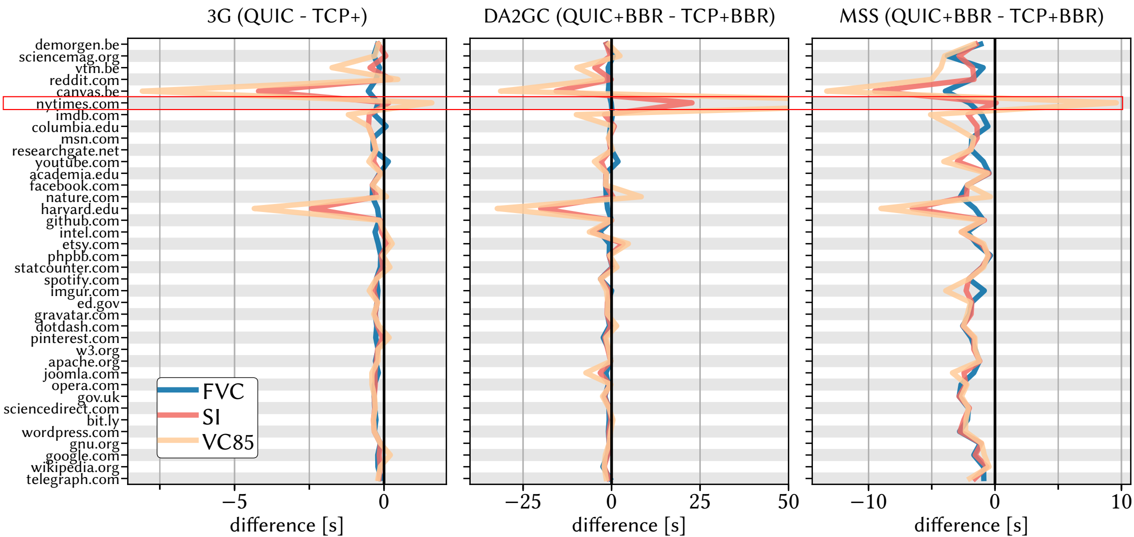


(Downlink 1.89Mbps, Uplink 1.89 Mbps, RTT 761ms, Loss 6.0%, [7])

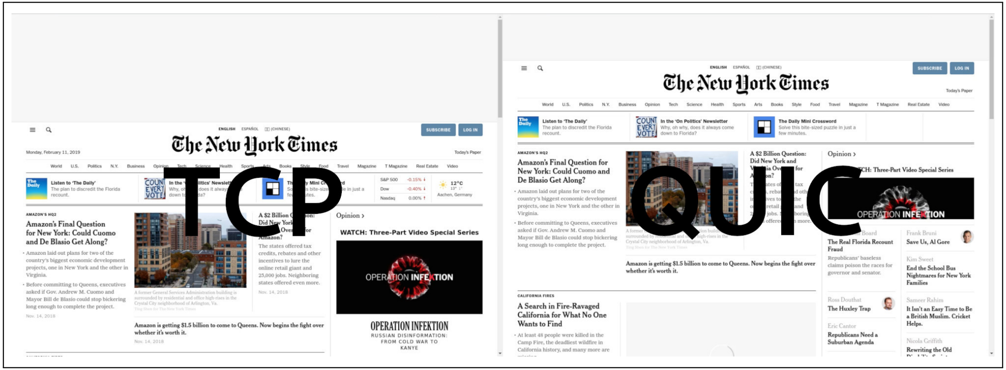
- **Congestion control impacts performance**
- **BBR outperforms CUBIC**
- **QUIC with CUBIC still faster than TCP with CUBIC**

Mean Performance Gain

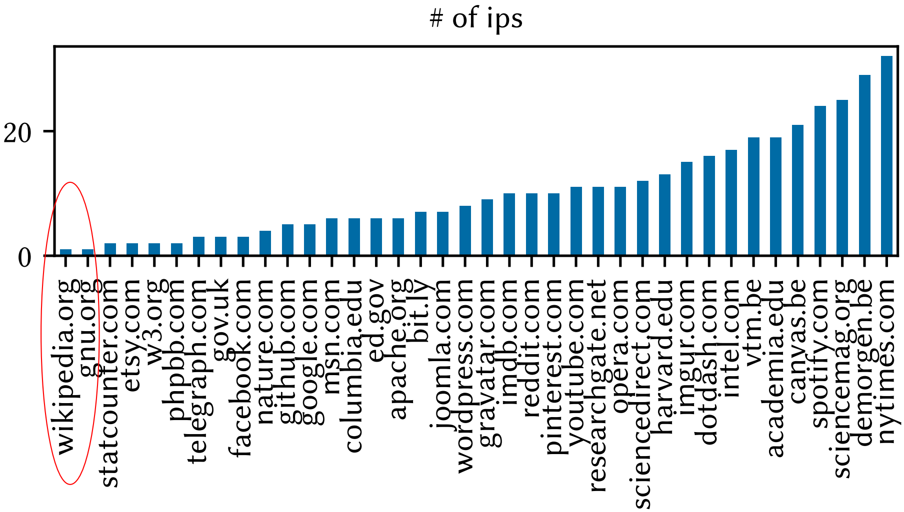



3G (QUIC - TCP+)
DA2GC (QUIC+BBR - TCP+BBR)
MSS (QUIC+BBR - TCP+BBR)
demorgen.be
sciencemag.org
vtm.be
reddit.com
canvas.be
nytimes.com
imdb.com
columbia.edu
msn.com
researchgate.net
youtube.com
academia.edu
facebook.com
nature.com
harvard.edu
github.com
intel.com
etsy.com
phpbb.com
statcounter.com
spotify.com
imgur.com
ed.gov
gravatar.com
dotdash.com
pinterest.com
w3.org
apache.org
joomla.com
opera.com
gov.uk
sciencedirect.com
bit.ly
wordpress.com
gnu.org
google.com
wikipedia.org
telegraph.com
FVC
SI
VC85
−5
0
−25
0
25
50
−10
0
10
difference [s]
difference [s]
difference [s]

Discussing Metrics



Subtracting Design Differences



Subtracting Design Differences

Mean difference under PLT with one subtracted RTT

$$\text{QUIC} - (\text{TCP+} - 1 \cdot \text{RTT})$$

Net	Website	[ms]	[RTT]
DSL	gnu.org	1.6	0.066
DSL	wikipedia.org	-3.1	-0.128
LTE	gnu.org	-30	-0.412
LTE	wikipedia.org	-13	-0.175
3G	gnu.org	-32	-0.344
3G	wikipedia.org	-54	-0.570
DA2GC	gnu.org	39	0.150
DA2GC	wikipedia.org	-1005	-3.834

Net	Website	[ms]	[RTT]
MSS	gnu.org	-1100	-1.447
MSS	wikipedia.org	-529	-0.696

Congestion Control: BBR

Net	Website	[ms]	[RTT]
MSS	gnu.org	-477	-0.628
MSS	wikipedia.org	451	0.593

Conclusion

- **Results**
  - TCP tuning is not negligible
  - Still QUIC outperforms TCP, but the gap gets narrower
  - QUIC mostly faster due to the RTT reduced connection establishment
  - Congestion control sometimes matters more than protocol choice
- **Discussion**
  - QUIC is not build to primarily improve performance
  - QUIC enables an evolvable stack especially on transport layer
  - Open question: Do users perceive QUIC as faster?

Thank you for your attention.

Literature I

P. Biswal and O. Gnawali.
Does quic make the web faster?
In *Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2016.

Breitbandmessung.
Breitbandmessung Ergebnisse als interaktive Darstellung.
https://web.archive.org/web/20181115105855/https://breitbandmessung.de/interaktive-darstellung.
Accessed: 2018-11-15.

G. Carlucci, L. De Cicco, and S. Mascolo.
HTTP over UDP: an Experimental Investigation of QUIC.
In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pages 609–614. ACM, 2015.

S. Cook, B. Mathieu, P. Truong, and I. Hamchaoui.
QUIC: Better for what and for whom?
In *International Conference on Communications (ICC)*, pages 1–6. IEEE, May 2017.

A. M. Kakhki, S. Jero, D. Choffnes, C. Nita-Rotaru, and A. Mislove.
Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols.
In *Proceedings of the Internet Measurement Conference (IMC)*, pages 290–303. ACM, 2017.

Literature II

P. Megyesi, Z. Krämer, and S. Molnár.
How quick is QUIC?
In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2016.

J. P. Rula, J. Newman, F. E. Bustamante, A. M. Kakhki, and D. Choffnes.
Mile High WiFi: A First Look At In-Flight Internet Connectivity.
In *Proceedings of the World Wide Web Conference on World Wide Web (WWW)*, pages 1449–1458, 2018.

M. Wijnants, R. Marx, P. Quax, and W. Lamotte.
HTTP/2 Prioritization and its Impact on Web Performance.
In *Proceedings of the World Wide Web Conference (WWW)*, pages 1755–1764. ACM, 2018.

Y. Yu, M. Xu, and Y. Yang.
When QUIC meets TCP: An experimental study.
In *36th International Performance Computing and Communications Conference (IPCCC)*, pages 1–8. IEEE, Dec 2017.

Y. Zheng, Y. Wang, M. Rui, A. Palade, S. Sheehan, and E. O'Nuallain.
Performance Evaluation of HTTP/2 over TLS+ TCP and HTTP/2 over QUIC in a Mobile Network.
Journal of Information Sciences and Computing Technologies, 7(1):673–682, 2018.