A Performance Perspective on Web Optimized Protocol Stacks

Konrad Wolsing, Jan Rüth, Oliver Hohlfeld, Klaus Wehrle

http://comsys.rwth-aachen.de

Hasselt, 2019-05-07

Yet another QUIC measurement?

[1] Carlucci et al. HTTP over UDP: An Experimental Investigation of QUIC. [2] Megyesi et al. How quick is QUIC? [3] Cook et al. QUIC: Better for what and for whom?. [4] Nepomuceno et al. QUIC and TCP: A Performance Evaluation. [5] Biswal et al. Does QUIC Make the Web Faster? [6] Kakhki et al. Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols.

Yet another QUIC measurement?

Unoptimized TCP stacks

- QUIC is optimized for the web
- TCP is configured for stability
- Related-work does not tune TCP

[1] Carlucci et al. HTTP over UDP: An Experimental Investigation of QUIC. [2] Megyesi et al. How quick is QUIC? [3] Cook et al. QUIC: Better for what and for whom?. [4] Nepomuceno et al. QUIC and TCP: A Performance Evaluation. [5] Biswal et al. Does QUIC Make the Web Faster? [6] Kakhki et al. Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols.

Yet another QUIC measurement?

Unoptimized TCP stacks

- QUIC is optimized for the web
- TCP is configured for stability
- Related-work does not tune TCP

Connection establishment

- QUIC requires 0 - 1 RTT
- TCP with TFO+TLS could do to so too

[1] Carlucci et al. HTTP over UDP: An Experimental Investigation of QUIC. [2] Megyesi et al. How quick is QUIC? [3] Cook et al. QUIC: Better for what and for whom?. [4] Nepomuceno et al. QUIC and TCP: A Performance Evaluation. [5] Biswal et al. Does QUIC Make the Web Faster? [6] Kakhki et al. Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols.

Yet another QUIC measurement?

Unoptimized TCP stacks

- QUIC is optimized for the web
- TCP is configured for stability
- Related-work does not tune TCP

Connection establishment

- QUIC requires 0 - 1 RTT
- TCP with TFO+TLS could do to so too

User-centered metrics

- research focuses on PLT
- PLT is not suited for user perception

[1] Carlucci et al. HTTP over UDP: An Experimental Investigation of QUIC. [2] Megyesi et al. How quick is QUIC? [3] Cook et al. QUIC: Better for what and for whom?. [4] Nepomuceno et al. QUIC and TCP: A Performance Evaluation. [5] Biswal et al. Does QUIC Make the Web Faster? [6] Kakhki et al. Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols.

Website selection

Website selection

Website list from Wijnants et al.[1] consisting of 40 domains chosen from Alexa and Moz list

- except their internal project webpage (vodlib)
- except cnet.com

⇒ total of 38 domains

[1] M. Wijnants, R. Marx, P. Quax, and W. Lamotte. Http/2 prioritization and its impact on web performance.

Website selection

Website list from Wijnants et al.[1] consisting of 40 domains chosen from Alexa and Moz list

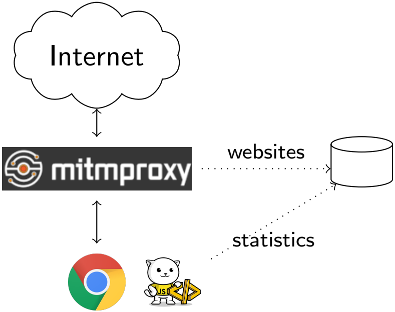
- except their internal project webpage (vodlib)
- except cnet.com

⇒ total of 38 domains

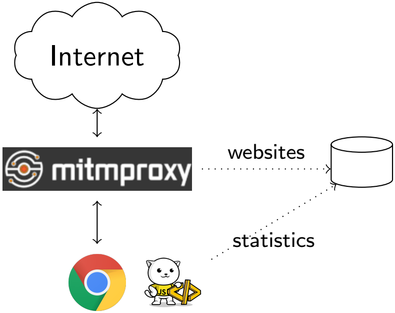
academia.edu apache.org bit.ly canvas.be columbia.edu demorgen.be dotdash.com ed.gov etsy.com facebook.com github.com gnu.org google.com gov.uk gravatar.com harvard.edu imdb.com imgur.com intel.com joomla.com msn.com nature.com nytimes.com opera.com phpbb.com pinterest.com reddit.com researchgate.net sciencedirect.com sciencemag.org spotify.com statcounter.com telegraph.com vtm.be w3.org wikipedia.org wordpress.com youtube.com

[1] M. Wijnants, R. Marx, P. Quax, and W. Lamotte. Http/2 prioritization and its impact on web performance.

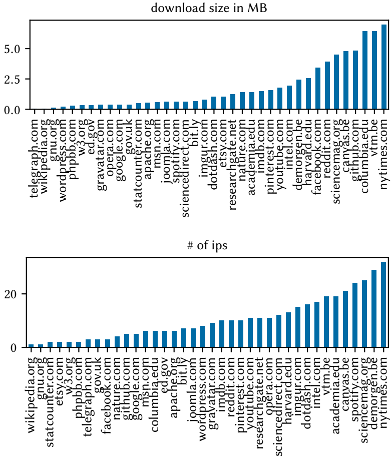
Downloading websites



Downloading websites

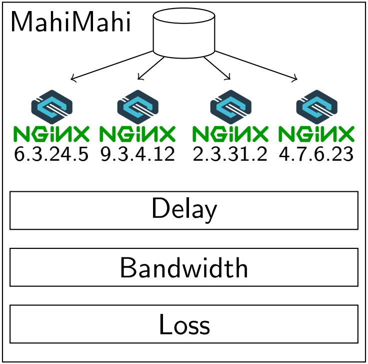


Website stats

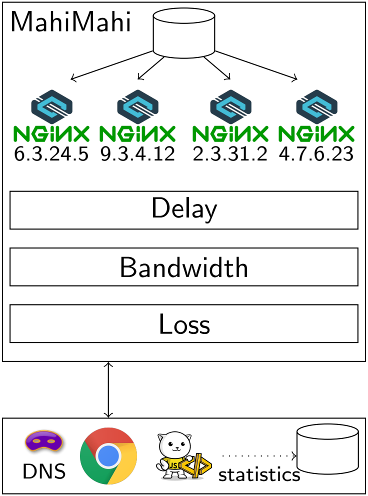

download size in MB

# of ips

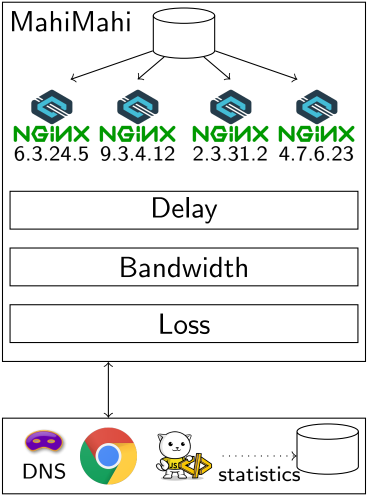
Testbed


MahiMahi
NGINX
6.3.24.5
NGINX
9.3.4.12
NGINX
2.3.31.2
NGINX
4.7.6.23
Delay
Bandwidth
Loss

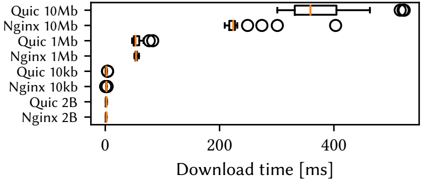
Testbed


MahiMahi
NGINX 6.3.24.5
NGINX 9.3.4.12
NGINX 2.3.31.2
NGINX 4.7.6.23
Delay
Bandwidth
Loss
DNS
JS
statistics

Testbed



Server speed test



Network configuration

	DSL [2]	LTE [2]	DA2GC [3]	MSS [3]
Uplink	5 Mbps	2.8 Mbps	.468Mbps	1.89Mbps
Downlink	25 Mbps	10.5 Mbps	.468Mbps	1.89Mbps
RTT	24ms	74ms	262ms	761ms
Loss	0.0%	0.0%	3.3%	6.6%
Queue size	12ms	200ms	200ms	200ms

[2] Breitbandmessung. https://breitbandmessung.de

[3] Rula et al. Mile high wifi: A first look at in-flight internet

Achieving comparability

Achieving comparability

TCP + TLS + HTTP

- ★ Initial window 10
- ✖ 1-RTT handshake
- ✖ Extra round trip for TLS
- ★ HTTP/1 or HTTP/2
- ✖ Head-of-line blocking

Achieving comparability

TCP + TLS + HTTP

- ★ Initial window 10
- ✖ 1-RTT handshake
- ✖ Extra round trip for TLS
- ★ HTTP/1 or HTTP/2
- ✖ Head-of-line blocking

QUIC

- ✓ Initial window 32
- ✓ 0-RTT connection setup
- ✓ No encryption overhead
- ✓ HTTP/2
- ✓ No head-of-line blocking

Achieving comparability

TCP + TLS + HTTP

- ✓ Initial window 32
- ✖ 1-RTT handshake
- ✖ Extra round trip for TLS
- ★ HTTP/1 or HTTP/2
- ✖ Head-of-line blocking

QUIC

- ✓ Initial window 32
- ✓ 0-RTT connection setup
- ✓ No encryption overhead
- ✓ HTTP/2
- ✓ No head-of-line blocking

Achieving comparability

TCP + TLS + HTTP	**QUIC**
✓ Initial window 32	✓ Initial window 32
✓ 0-RTT TCP FastOpen	✓ 0-RTT connection setup
✓ Use TLS1.3 with 0-RTT	✓ No encryption overhead
★ HTTP/1 or HTTP/2	✓ HTTP/2
✖ Head-of-line blocking	✓ No head-of-line blocking

Achieving comparability

TCP + TLS + HTTP

- ✓ Initial window 32
- ★ 1-RTT handshake
- ★ Use TLS1.3 without 0-RTT
- ★ HTTP/1 or HTTP/2
- ✖ Head-of-line blocking

QUIC

- ✓ Initial window 32
- ✓ 0-RTT connection setup
- ✓ No encryption overhead
- ✓ HTTP/2
- ✓ No head-of-line blocking

Achieving comparability

TCP + TLS + HTTP

- ✓ Initial window 32
- ★ 1-RTT handshake
- ★ Use TLS1.3 without 0-RTT
- ★ HTTP/1 or HTTP/2
- ✗ Head-of-line blocking

QUIC

- ✓ Initial window 32
- ★ no 0-RTT connection setup
- ✓ No encryption overhead
- ✓ HTTP/2
- ✓ No head-of-line blocking

Achieving comparability

TCP + TLS + HTTP

- ✓ Initial window 32
- ★ 1-RTT handshake
- ★ Use TLS1.3 without 0-RTT
- ✓ Use only HTTP/2
- ✖ Head-of-line blocking

QUIC

- ✓ Initial window 32
- ★ no 0-RTT connection setup
- ✓ No encryption overhead
- ✓ HTTP/2
- ✓ No head-of-line blocking

Achieving comparability

TCP + TLS + HTTP

- ✓ Initial window 32
- ★ 1-RTT handshake
- ★ Use TLS1.3 without 0-RTT
- ✓ Use only HTTP/2
- ✖ Head-of-line blocking

QUIC

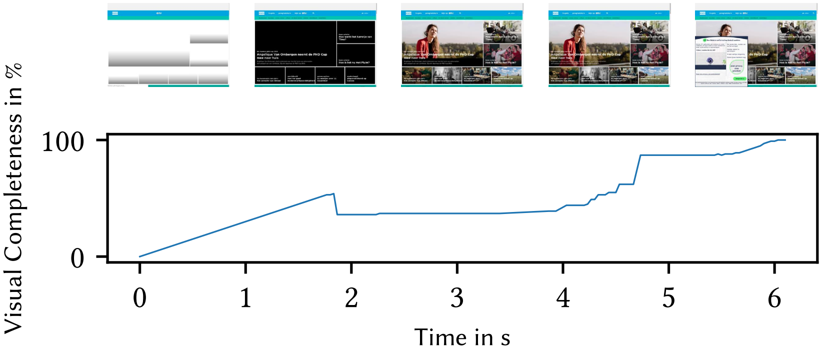
- ✓ Initial window 32
- ★ no 0-RTT connection setup
- ✓ No encryption overhead
- ✓ HTTP/2
- ✓ No head-of-line blocking

Protocol settings

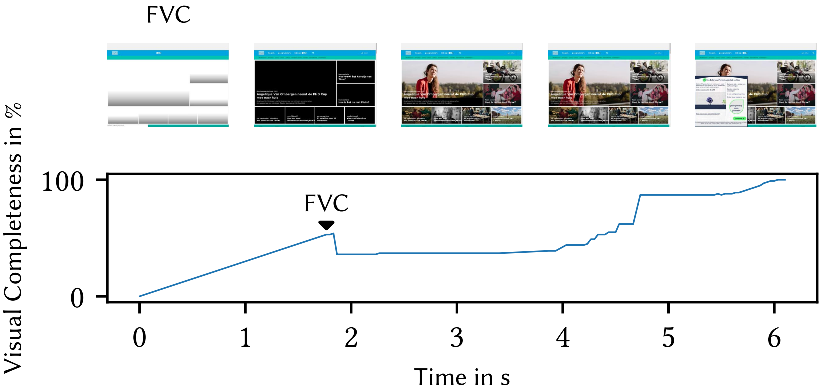
TCP	Stock TCP (Linux): IW10, Cubic.
TCP+	IW 32, Pacing, tuned network buffers, no slow start after idle, Cubic.
TCP+BBR	TCP+, but with BBR as congestion control.
QUIC	Google QUIC Version 43: IW32, Pacing, Cubic.
QUIC+BBR	QUIC, but with BBR as congestion control.

Visual Metrics

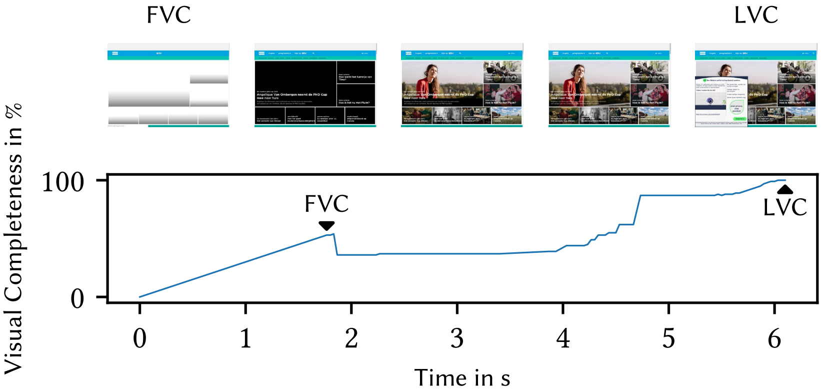
Visual Metrics



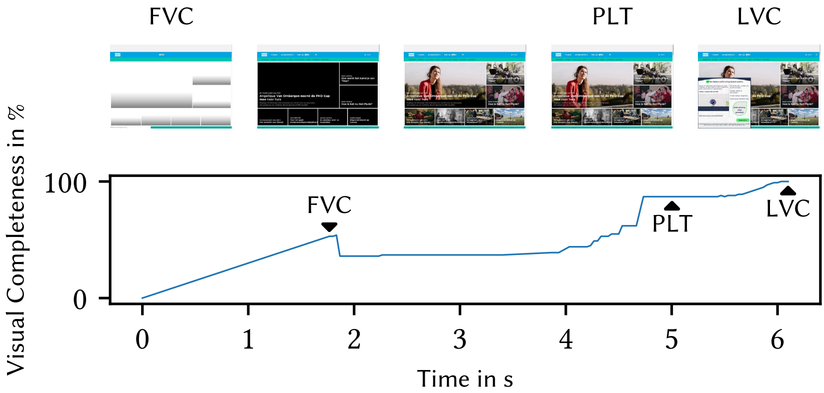
Visual Metrics



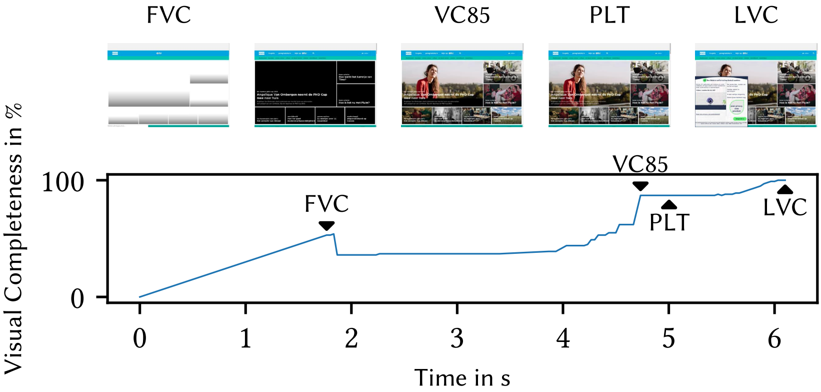
Visual Metrics



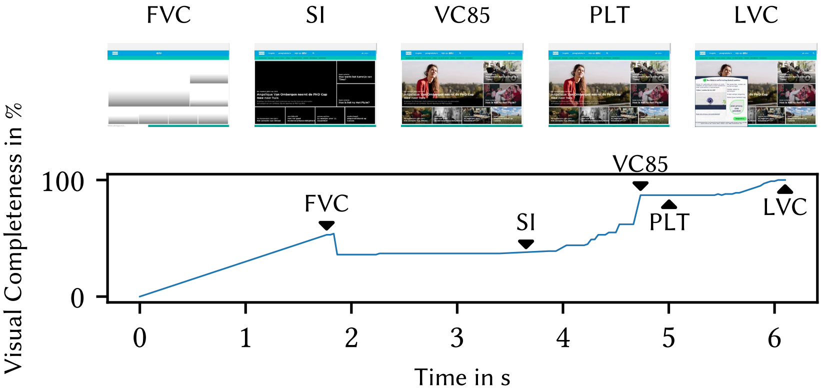
Visual Metrics



Visual Metrics

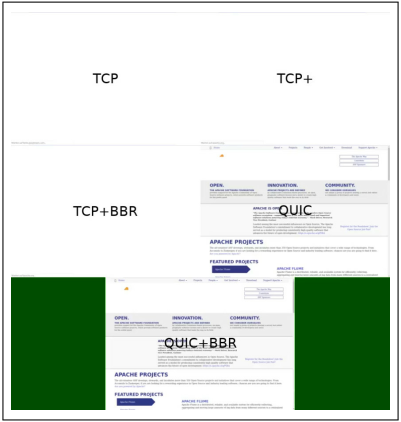


Visual Metrics



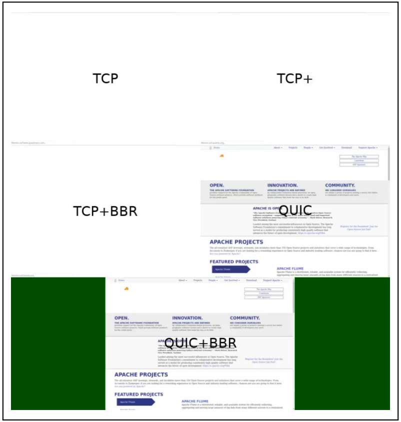
Measurement

Example video



Measurement

Example video

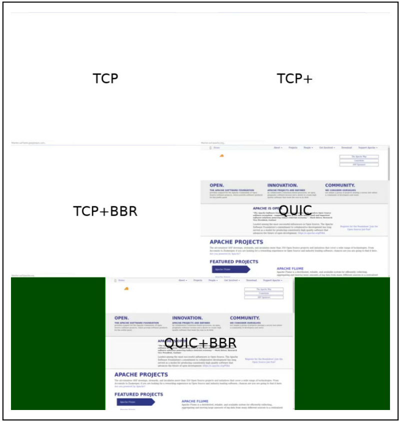


Recapitulation

- 38 different websites
- 4 network scenarios
- 5 protocol configurations
- 31 repetitions each
- 5 (visual) metrics evaluated

Measurement

Example video

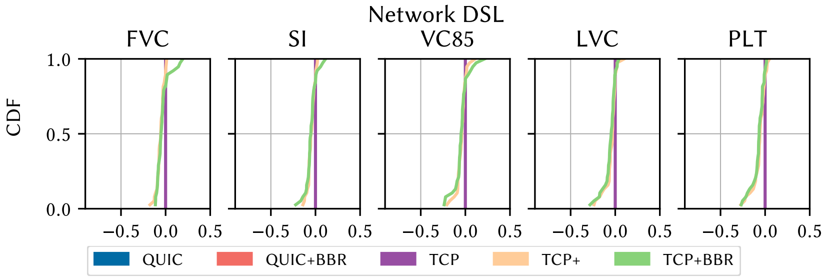


Recapitulation

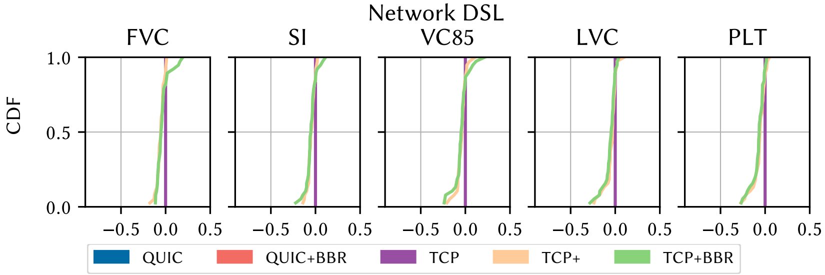
- 38 different websites
- 4 network scenarios
- 5 protocol configurations
- 31 repetitions each
- 5 (visual) metrics evaluated

$$\textit{performance gain} = \frac{\overline{X}_{new} - \overline{X}_{ref}}{\overline{X}_{ref}}$$

Performance gain DSL

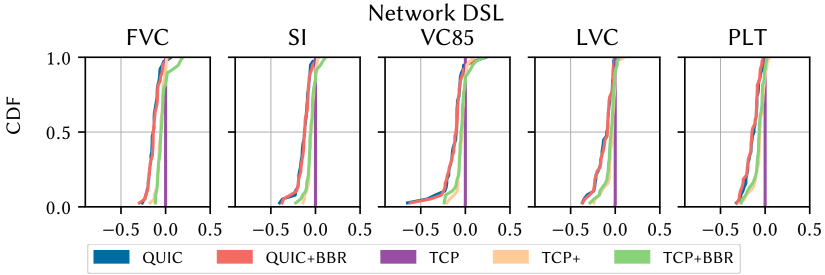


Performance gain DSL



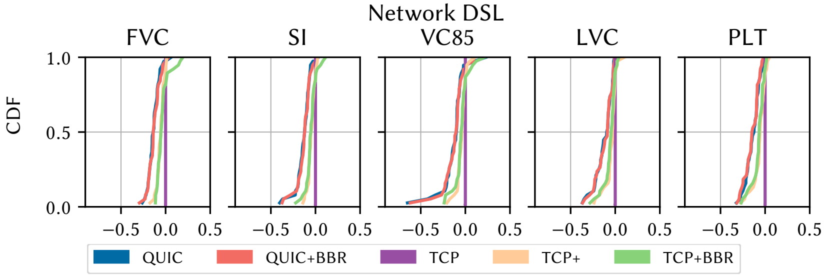
- Stock TCP is slower than the tuned variants
- Congestion control does not impact performance here

Performance gain DSL



- Stock TCP is slower than the tuned variants
- Congestion control does not impact performance here

Performance gain DSL



- Stock TCP is slower than the tuned variants
- Congestion control does not impact performance here
- QUIC outperforms even tuned TCP, but the gap gets narrower
- Step curves indicate that website size/structure seems to have little impact on performance

ANOVA test DSL (< 0.01)

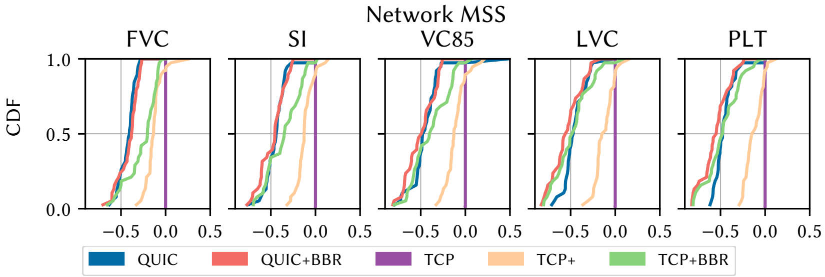
ANOVA test DSL (< 0.01)

DSL	FVC	SI	VC85	LVC	PLT
TCP+ TCP ?	28 0 10				
QUIC TCP ?					
QUIC TCP+ ?					
TCP+BBR TCP+ ?					
QUIC TCP+BBR ?					
QUIC+BBR TCP+BBR ?					
QUIC+BBR QUIC ?					

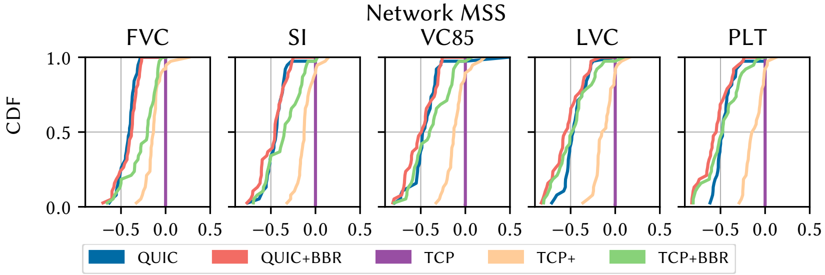
ANOVA test DSL (< 0.01)

DSL	FVC	SI	VC85	LVC	PLT
TCP+ TCP ?	28 0 10	26 0 12	24 0 14	22 0 16	25 1 12
QUIC TCP ?	36 1 1	37 1 0	35 1 2	36 0 2	35 0 3
QUIC TCP+ ?	30 2 6	32 0 6	28 1 9	29 0 9	30 0 8
TCP+BBR TCP+ ?	4 5 29	7 2 29	9 2 27	6 2 30	7 2 29
QUIC TCP+BBR ?	30 0 8	34 1 3	28 3 7	29 1 8	27 4 7
QUIC+BBR TCP+BBR ?	31 0 7	35 1 2	27 3 8	28 1 9	31 1 6
QUIC+BBR QUIC ?	3 6 29	3 6 29	2 7 29	2 5 31	7 3 28

Performance gain MSS

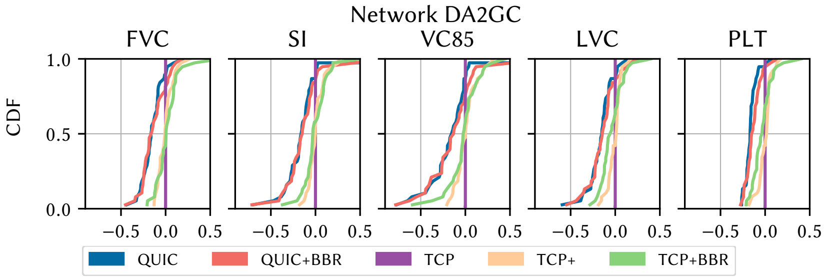


Performance gain MSS

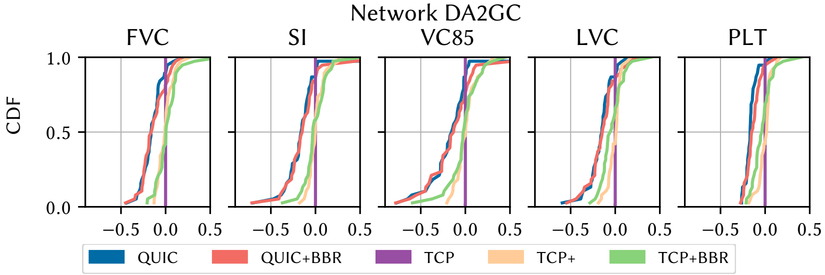


- Congestion control impacts performance gain
- BBR superior than CUBIC
- QUIC with CUBIC still faster than TCP with CUBIC

Performance gain DA2GC



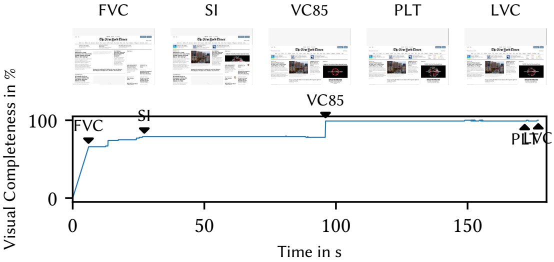
Performance gain DA2GC



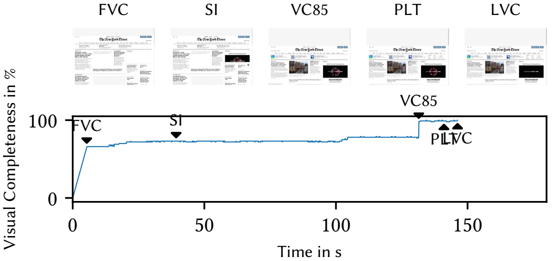
- No large difference noticeable between TCP configurations
- QUIC outperforms TCP
- For PLT, QUIC marginally faster than QUIC with BBR

Discussing Metrics

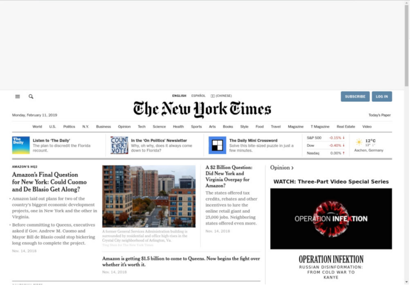
TCP



QUIC



Discussing Metrics



Monday, February 11, 2019

Listen to 'The Daily' The plan to discredit the Florida recount.

In the 'On Politics' Newsletter Why, oh why, does it always come down to Florida?

The Daily Mini Crossword Solve this bite-sized puzzle in just a few minutes.

AMAZON'S HQ2

Amazon's Final Question for New York: Could Cuomo and De Blasio Get Along?

- Amazon laid out plans for two of the country's biggest economic development projects, one in New York and the other in Virginia.
- Before committing to Queens, executives asked if Gov. Andrew M. Cuomo and Mayor Bill de Blasio could stop bickering long enough to complete the project.

Nov. 14, 2018

A former General Services Administration building is surrounded by residential and office high-rises in the Crystal City neighborhood of Arlington, Va.
Ting Shen for The New York Times

A $2 Billion Question: Did New York and Virginia Overpay for Amazon?

The states offered tax credits, rebates and other incentives to lure the online retail giant and 25,000 jobs. Neighboring states offered even more.

Nov. 14, 2018

Amazon is getting $1.5 billion to come to Queens. Now begins the fight over whether it's worth it.

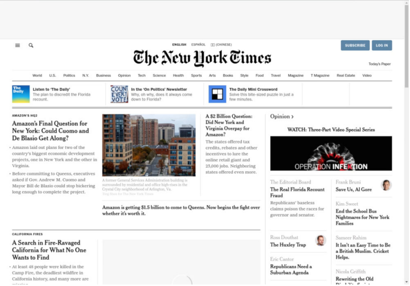
Nov. 14, 2018

Opinion ›

WATCH: Three-Part Video Special Series

OPERATION INFEKTION
RUSSIAN DISINFORMATION: FROM COLD WAR TO KANYE

TCP



Listen to 'The Daily' The plan to discredit the Florida recount.

In the 'On Politics' Newsletter Why, oh why, does it always come down to Florida?

The Daily Mini Crossword Solve this bite-sized puzzle in just a few minutes.

AMAZON'S HQ2

Amazon's Final Question for New York: Could Cuomo and De Blasio Get Along?

- Amazon laid out plans for two of the country's biggest economic development projects, one in New York and the other in Virginia.
- Before committing to Queens, executives asked if Gov. Andrew M. Cuomo and Mayor Bill de Blasio could stop bickering long enough to complete the project.

A former General Services Administration building is surrounded by residential and office high-rises in the Crystal City neighborhood of Arlington, Va.
Ting Shen for The New York Times

A $2 Billion Question: Did New York and Virginia Overpay for Amazon?

The states offered tax credits, rebates and other incentives to lure the online retail giant and 25,000 jobs. Neighboring states offered even more.

Amazon is getting $1.5 billion to come to Queens. Now begins the fight over whether it's worth it.

CALIFORNIA FIRES

A Search in Fire-Ravaged California for What No One Wants to Find

- At least 48 people were killed in the Camp Fire, the deadliest wildfire in California history, and many more are

Opinion ›

WATCH: Three-Part Video Special Series

The Editorial Board
The Real Florida Recount Fraud
Republicans' baseless claims poison the races for governor and senator.

Ross Douthat
The Huxley Trap

Eric Cantor
Republicans Need a Suburban Agenda

Frank Bruni
Save Us, Al Gore

Kim Sweet
End the School Bus Nightmares for New York Families

Sameer Rahim
It Isn't an Easy Time to Be a British Muslim. Cricket Helps.

Nicola Griffith
Rewriting the Old

QUIC

Measuring human perception

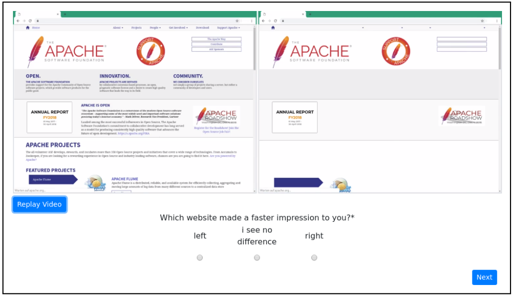
Design

User-study

https://study.netray.io

Measuring human perception

Design



Which website made a faster impression to you?*

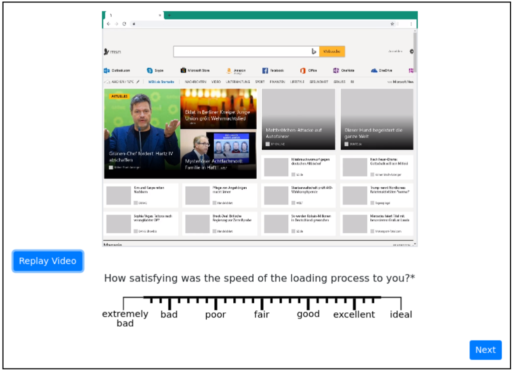
○ left ○ i see no difference ○ right

User-study

https://study.netray.io

Measuring human perception

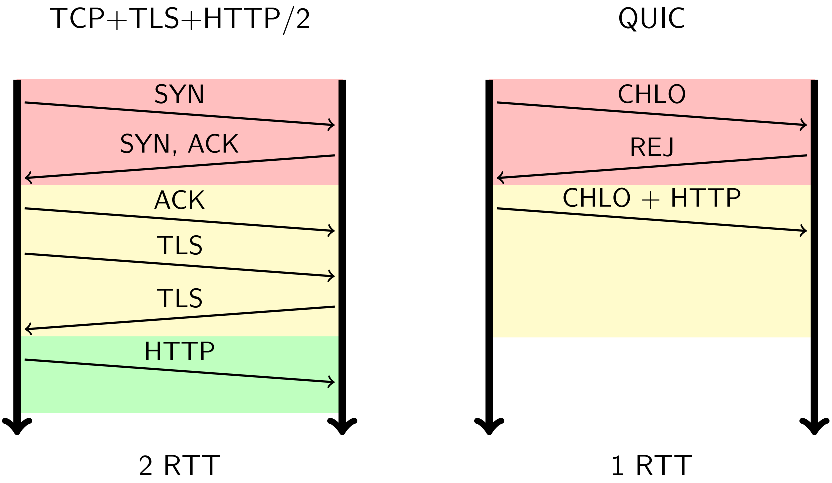
Design


How satisfying was the speed of the loading process to you?*

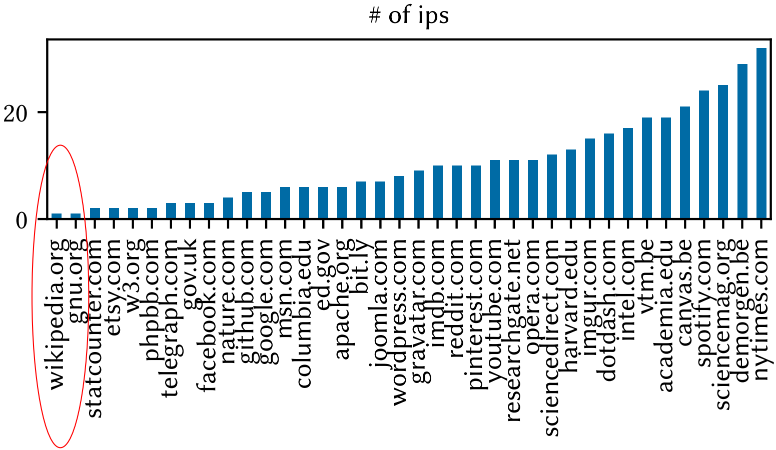
User-study

https://study.netray.io

Subtracting design flaws



Subtracting design flaws



Subtracting design flaws

Mean difference under PLT with one subtracted RTT

TCP+ QUIC

Network	Website	[ms]	[RTT]
DSL	gnu.org	1.6	0.07
DSL	wikipedia.org	-3.1	-0.13
LTE	gnu.org	-30	-0.41
LTE	wikipedia.org	-13	-0.18
DA2GC	gnu.org	39	0.15
DA2GC	wikipedia.org	-1005	-3.83

TCP+BBR QUIC+BBR

Network	Website	[ms]	[RTT]
MSS	gnu.org	-477	-0.63
MSS	wikipedia.org	451	0.59

Conclusion

Motivation

- TCP and QUIC comparisons have often been biased
- user perception was not the scope up to now

Realization

- Extended Mahimahi to support QUIC
- Tuned TCP achieving fairness

Conclusion

Motivation

- TCP and QUIC comparisons have often been biased
- user perception was not the scope up to now

Realization

- Extended Mahimahi to support QUIC
- Tuned TCP achieving fairness

Results

- Tuning TCP is not negligible
- Still QUIC outperforms TCP, but the gap gets narrower
- QUIC's performance gain mostly caused by the superior connection establishment design
- Congestion control sometimes matters more than protocol choice

User-study

https://study.netray.io