neət

Understanding Multistreaming for Web Traffic: An Experimental Study

M. Rajiullah[†], A. C. Mohideen[⋆], F. Weinrank[‡], R. Secchi[⋆], G. Fairhurst[⋆] and A. Brunstrom[†]

[†]Karlstad University, Karlstad, Sweden
[⋆]University of Aberdeen, Aberdeen, U.K.
[‡]FHM, Munster, germany

Horizon 2020
European Union funding
for Research & Innovation

Outline

- In the Internet, Web is still the king
- HTTP/1.1 known issues
- A way forward – change http?
- Web Model & Dataset
- Tools And Experiment Setup
- Benefit of Parallelism
- Impact of Processing Time, Loss
- Discussion of Experiment Setup
- Conclusion
- Q & A
- Future of Web Protocol

In the Internet, Web is still the king

- Browser-based services are popular, e.g. search, entertainment, productivity, business, social and personal communication
- Latency is the most important factor impacting browsing experience.
- Slow browsing is not just *annoying* to end-users, but also *costly* for content owners.

HTTP/1.1 known issues

- HTTP/1.1 remains the de-facto standard for loading web pages
- Web pages have evolved:
  - Pages with many objects/resources
  - Objects with complex dependencies
  - Head-of-Line blocking in HTTP/1.1 makes things slow
- Multiple transport connections help:
  - Can download many objects in parallel
  - But, shortcomings – more state, more contention
  - Domain sharding increases parallelism *even more*
  - Other solutions like spriting, inlining and concatenation of resources also have their own shortcomings

neat

A way forward – change http?

- Application-based improvement using Google SPDY, IETF Standard HTTP/2.0
- Transport-based proposals, Google QUIC, IETF QUIC?
- So what should transport for web look like?
  - Multi-streaming (one transport flow, multiple streams)
- We compare multi-streaming using SCTP against multiple TCP connections for web to understand the benefits across a range of usage:
  1. We present a web model
  2. We evaluate the impact of RTT, loss and capacity

Web Model & Dataset

- Utilised a public web performance dataset*
- Dataset contains graphs representing dependency between HTTP resources and their processing time at the client
- We categorized the web pages according to the total size of all resources in a page
- The total was used to divide pages into 6 bins (size-ranks), labeled A to F

* X. S. Wang *et al.*, “How Speedy is SPDY?” in *11th USENIX Symposium on Networked Systems Design and Implementation* , Seattle, Apr. 2014, pp. 387–399.

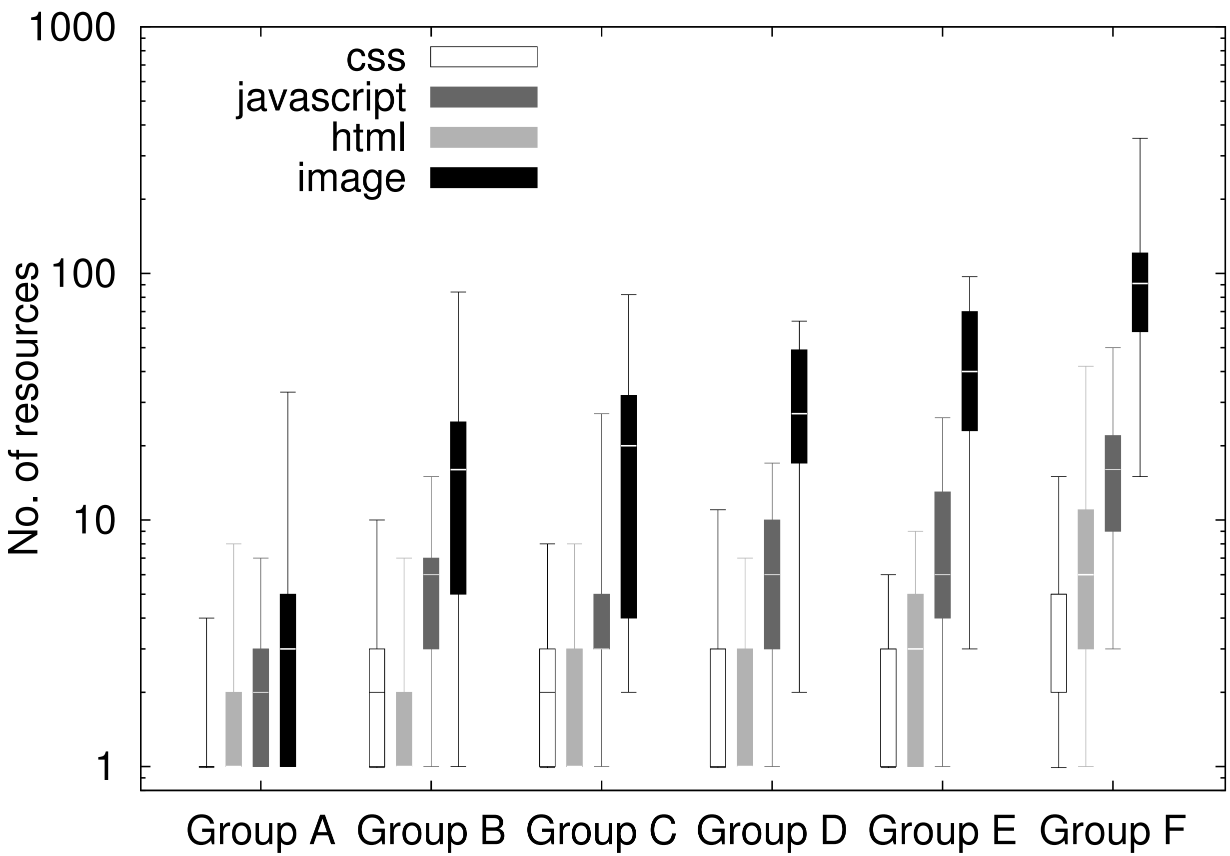
neat

Web Model (1)

Group Name	Size-Rank (KB)	Size (KB) and # res. at 5%	Size (KB) and # res. at 50%	Size (KB) and # res. at 95%
A	0.05-118	0.05 (1)	23 (6)	109 (39)
B	119-565	129 (3)	325 (21)	532 (67)
C	566-873	567 (6)	690 (25)	846 (69)
D	874-1242	878 (6)	964 (45)	1183 (82)
E	1243-1945	1286 (24)	1546 (55)	1901(119)
F	1946-3315	2070 (49)	2454 (127)	3309 (228)

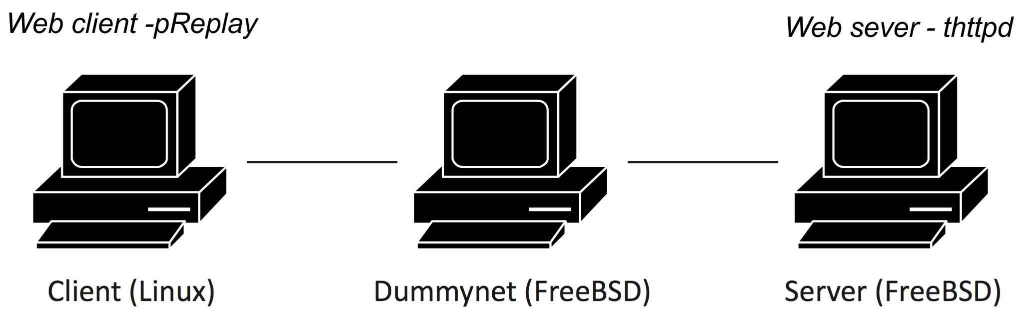
- Correlation between page size and number of resources
- Pages of similar sizes have quite dissimilar compositions

Web Model (2)



- In all cases, the most common resources are images

Tools And Experiment Setup

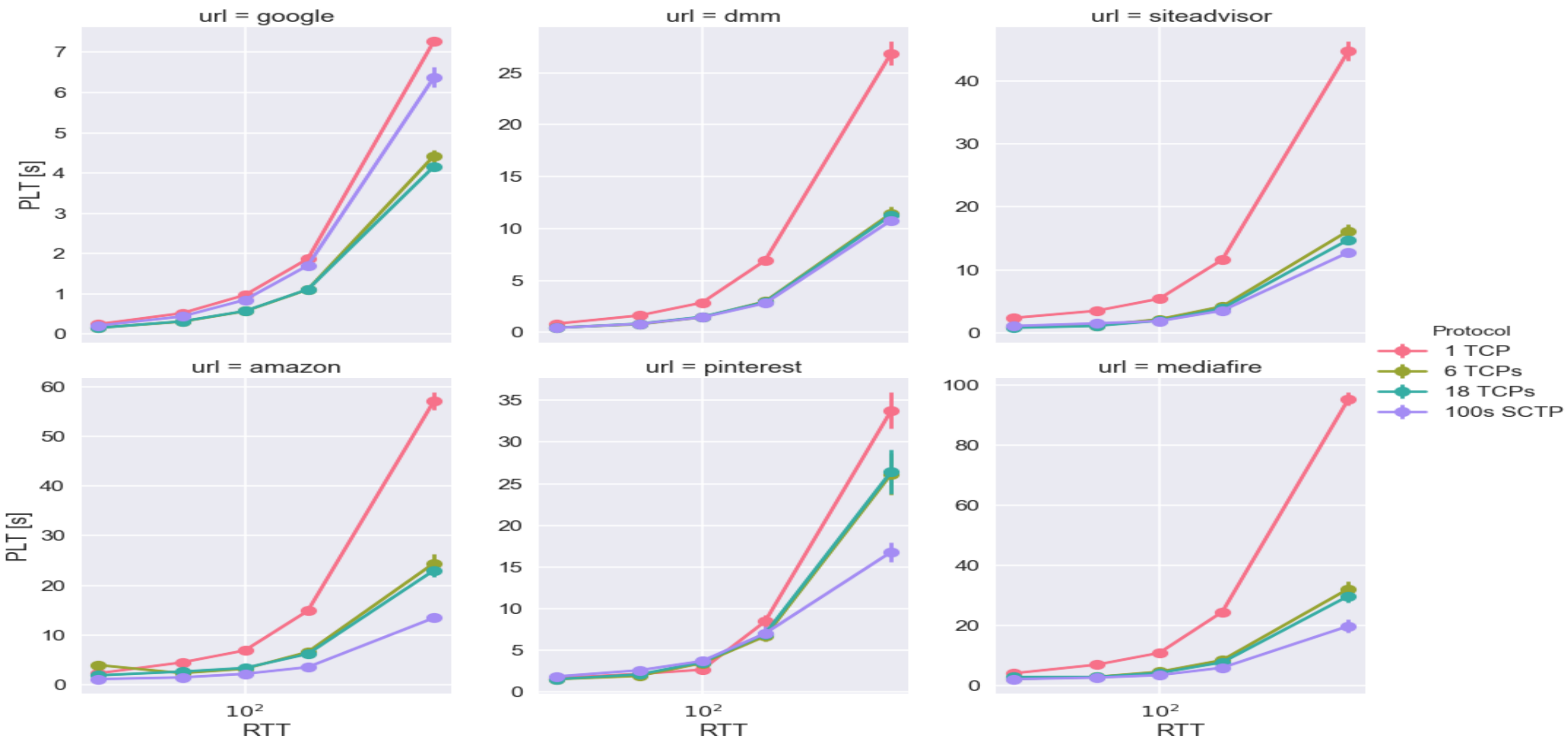


Experiment parameters		
Category	Factor	Range/value
Network	RTTs	20, 50, 100, 200, 800 ms
	Bottleneck Capacity	2, 10, 100 Mbps
	Packet loss	No loss, 1.5%, 3%
TCP/SCTP	IW	client (IW 3), server (IW 10)
	CWND validation	no
	# parallel TCP flows	1, 6, 18
	# streams in SCTP	1, 6, 18, 100
Requests	Cookie Size	NULL, 512 B, 2K

Page Load Time

- We explore
  - Impact of parallelism (no added loss)
  - Impact of processing time
  - Impact of loss

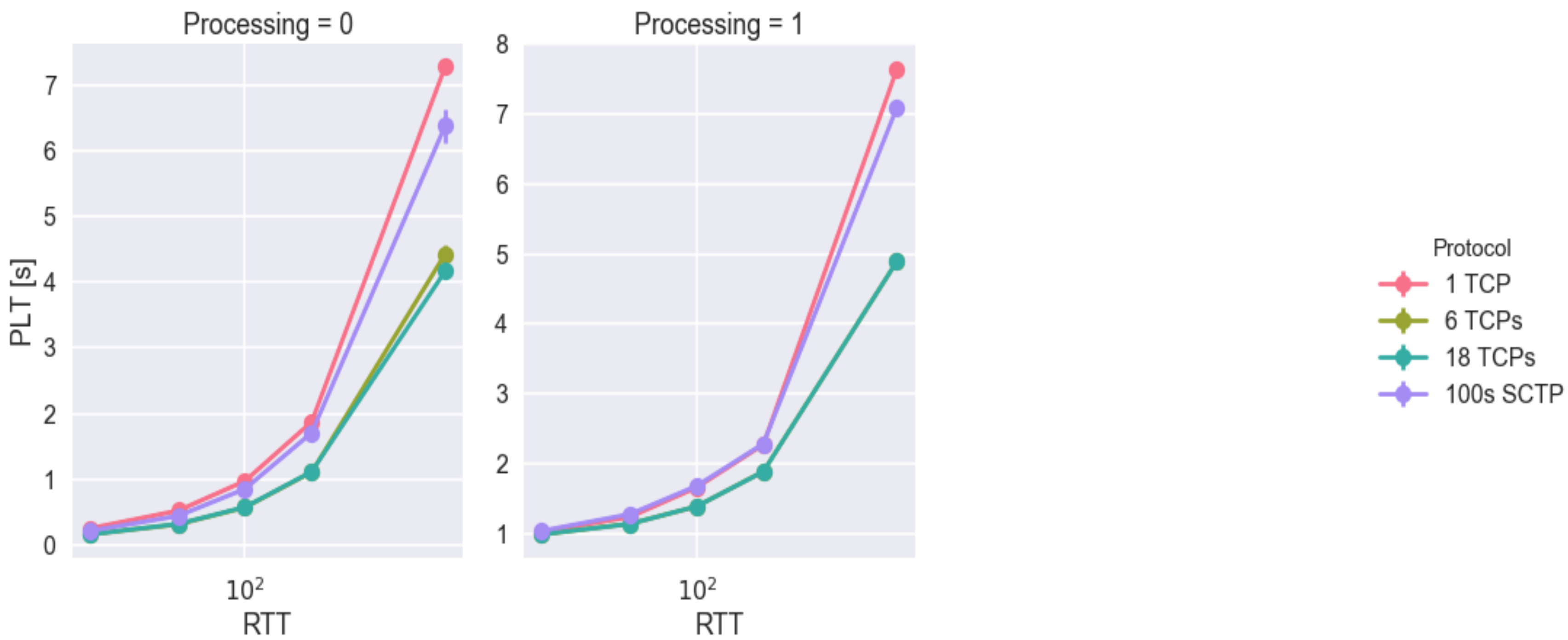
Benefit of Parallelism



- Multi-streaming provides similar to better performance
- Multi-streaming shows more benefit in higher RTT

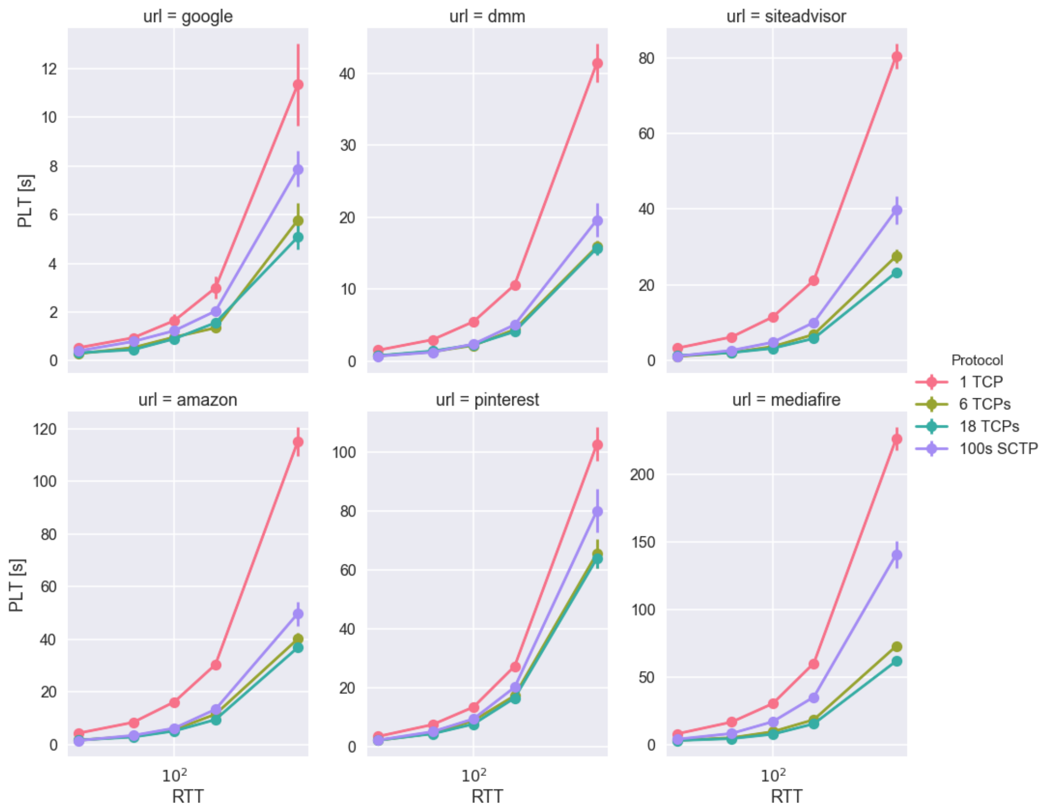
neət

Impact of Processing Time



- Upper bound of performance from processing time
- Processing time inflates PLTs

Impact of Loss



- Parallelism helps TCPs when loss happens (but can be aggressive)
- Multi-streaming improves on head of line blocking but its conservative congestion control inflates the PLT

neat

Discussion of Experiment Setup

- A key benefit of multistreaming is the lightweight cost for additional streams
- No domain sharding
- We only consider pseudo-random link loss

Conclusion

- We used a data-driven workload
- Our results commented on how mechanisms were impacted by the level of parallelism and RTT
- Key transport explored multistreaming, parallelism, shared and individual congestion control
- Multi-streaming enabled rapid utilisation of available bottleneck capacity
- A clear cost in terms of performance is the single congestion-control context, although could have benefits in fairer sharing with other flows.

Future of Web Protocol

- Our evaluation (of multistreaming) is inline with the current HTTP1.1 vs. HTTP2 debate
- QUIC solves the Head-of-line problem from single connection using UDP

NEAT and SCTP

- Web is still the most important use case for future Internet
- SCTP can be leveraged by a client, but currently not widely used by web servers
- NEAT can help gradual deployment
  - Our results can inform policy in the NEAT stack

THE END

THANK YOU FOR LISTENING