MP-DCCP and congestion control in congestion control

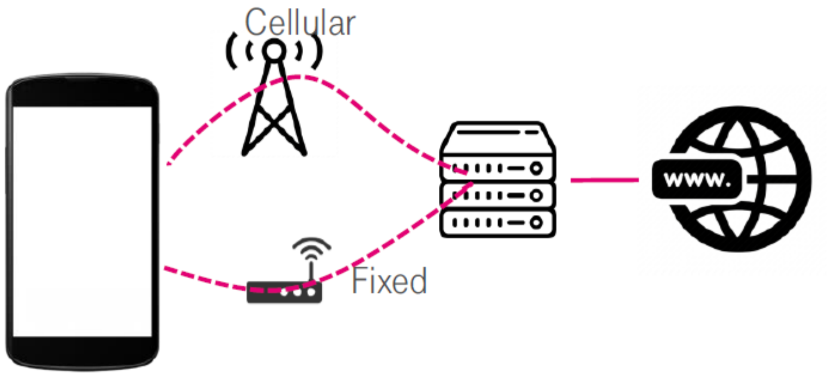
Anna Brunstrom, Karlstad University
Results by Marcus Pieskä

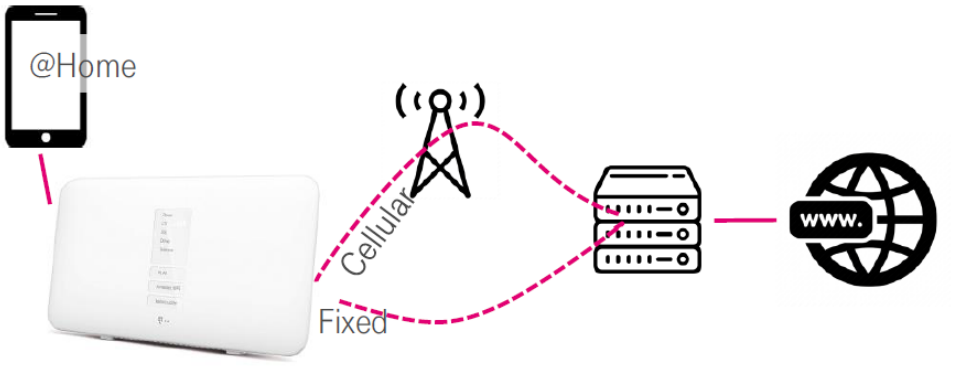
IETF 110

LIFE IS FOR SHARING.

MP-DCCP Framework

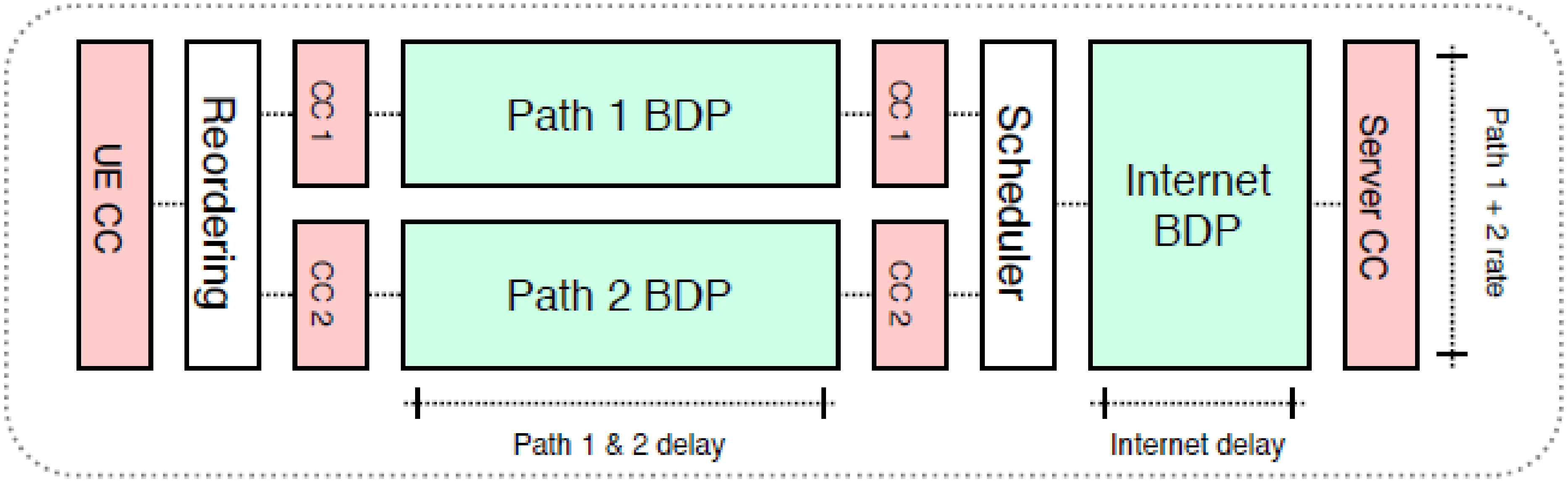
- Multipath solution for UDP – IP traffic
- Performs link aggregation by using DCCP as the protocol
  - One DCCP tunnel per path
- Use cases: Mobile device multi-connectivity in 3GPP ATSSS, residential multi-connectivity based on Hybrid Access





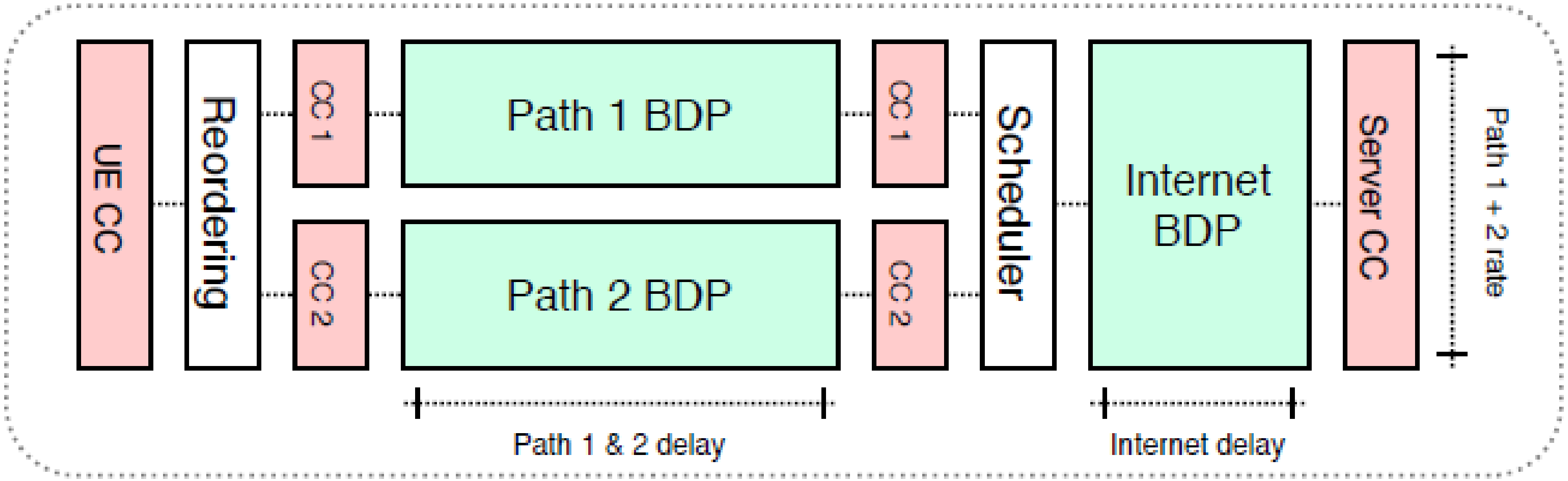
Congestion control in congestion control

- Tunneling solution results in nested congestion controls
- Multipath brings added complexity
- We use uncoupled congestion control over the two paths



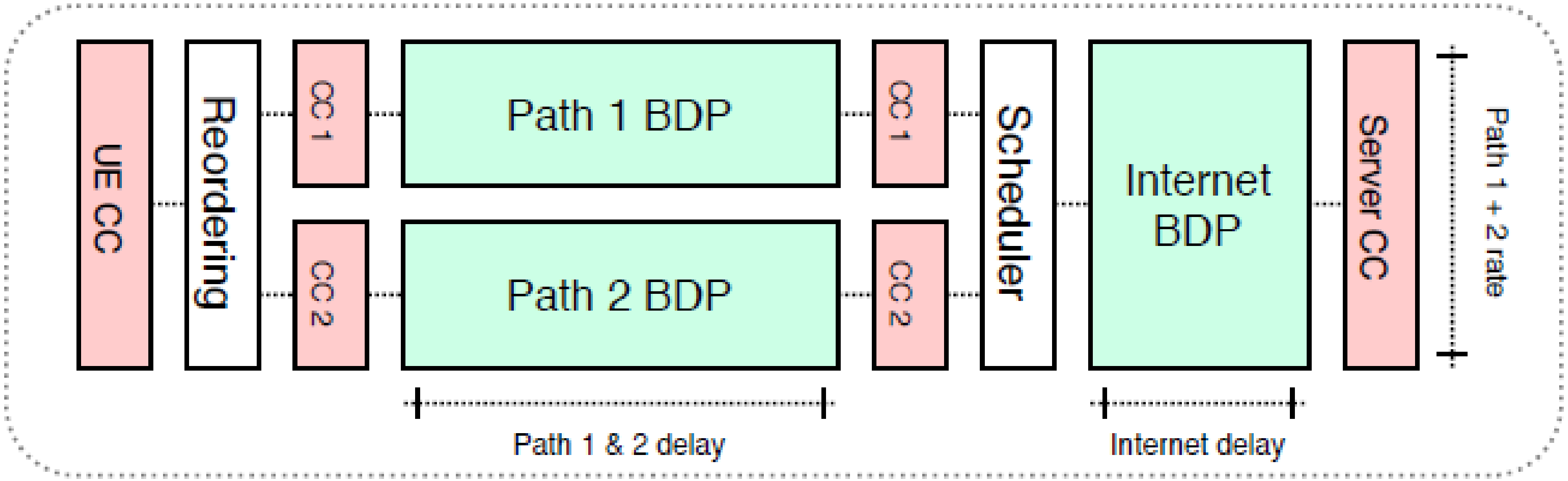
Congestion control in congestion control

- Cheapest path first (strict priority) scheduler
  - Sends data on cheapest path whenever available
  - If the congestion window of the cheapest path is full, it sends on the next available path
- Reordering using an adaptive time limit based on monitored delay difference between the paths



Congestion control in congestion control

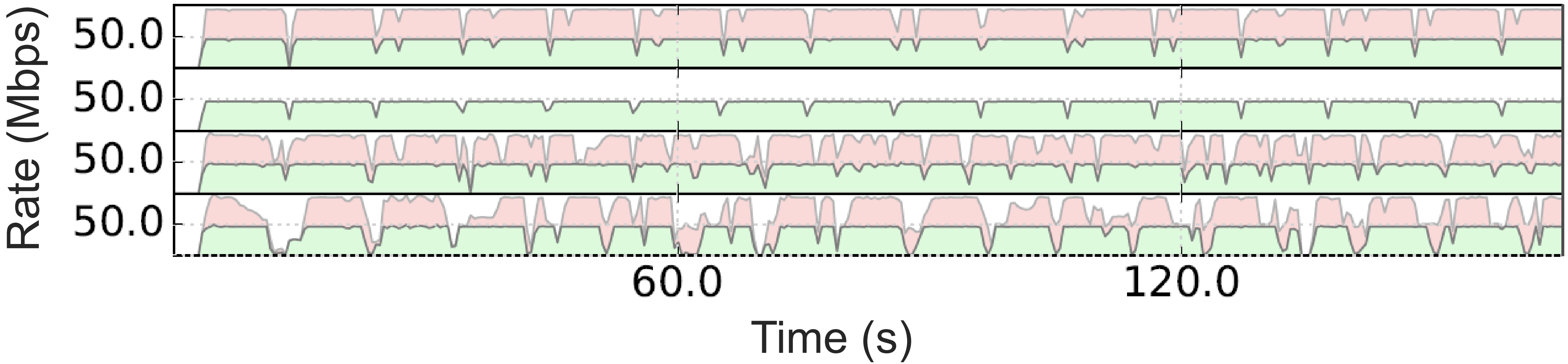
- Key challenge: aggregation of capacity over the two paths
- Using the second path before E2E congestion control reacts and slows down the sending rate



Congestion control in congestion control

- Key challenge: aggregation of capacity over the two paths
- Using the second path before E2E congestion control reacts and slows down the sending rate

Tunnel throughput over time, stacked with path1 green and path2 red, for four sample configurations



User-space implementation of multipath framework

- MP-DCCP tunnel framework in user-space offering flexible experimentation
- Program that accepts packets from a Linux TUN-device
- Encapsulate packets with information like path sequence numbers and timestamps
- Scheduling occur over single path sockets
- Solution is only loosely attached to DCCP and allows for other tunnel protocols

- Results from Android experiments with kernel-level implementation presented in tsvwg during IETF 109

Default parameters and CCs used for Mininet experiments

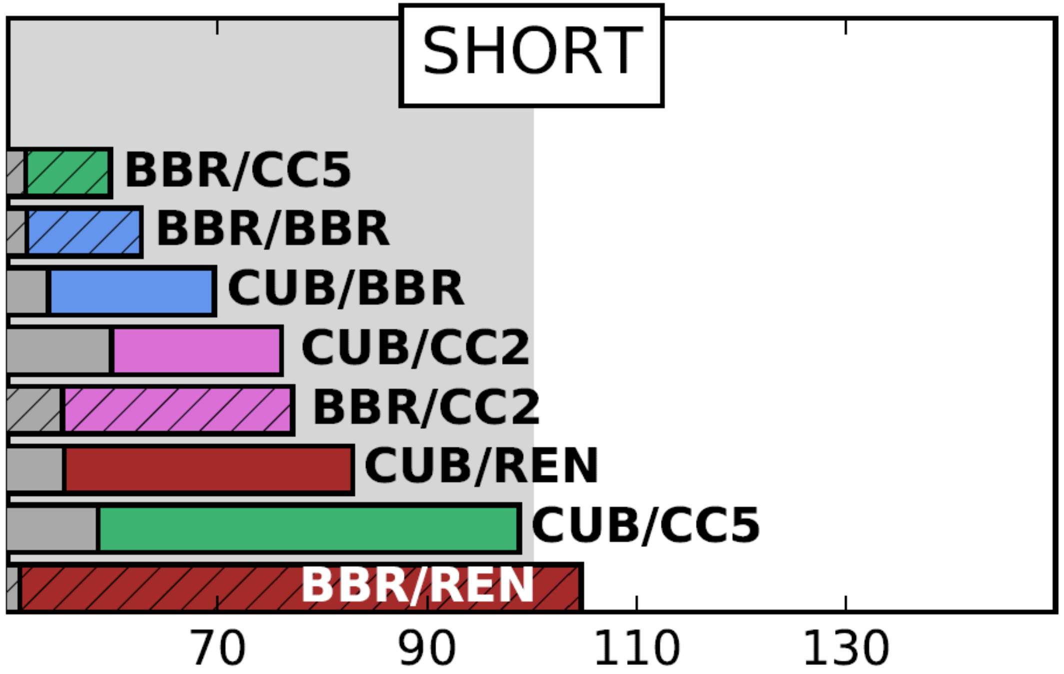
Deployment:	Congestion control set
End-to-End:	TCP-Cubic, TCP-BBR
Tunnel:	TCP-NewReno, TCP-BBR, CCID2, **CCID5***

Parameter:	Value
Server link RTT:	20 ms
Path 1 RTT:	20 ms
Path 2 RTT:	40 ms
Path 1 & 2 bandwidth:	50 Mbps
Path 1 & 2 MAC buffer:	500 pkt
Path 1 & 2 queuing discipline	FIFO

*Implementation of BBRv1 within the modular DCCP CCID framework

Impact of different congestion control combinations

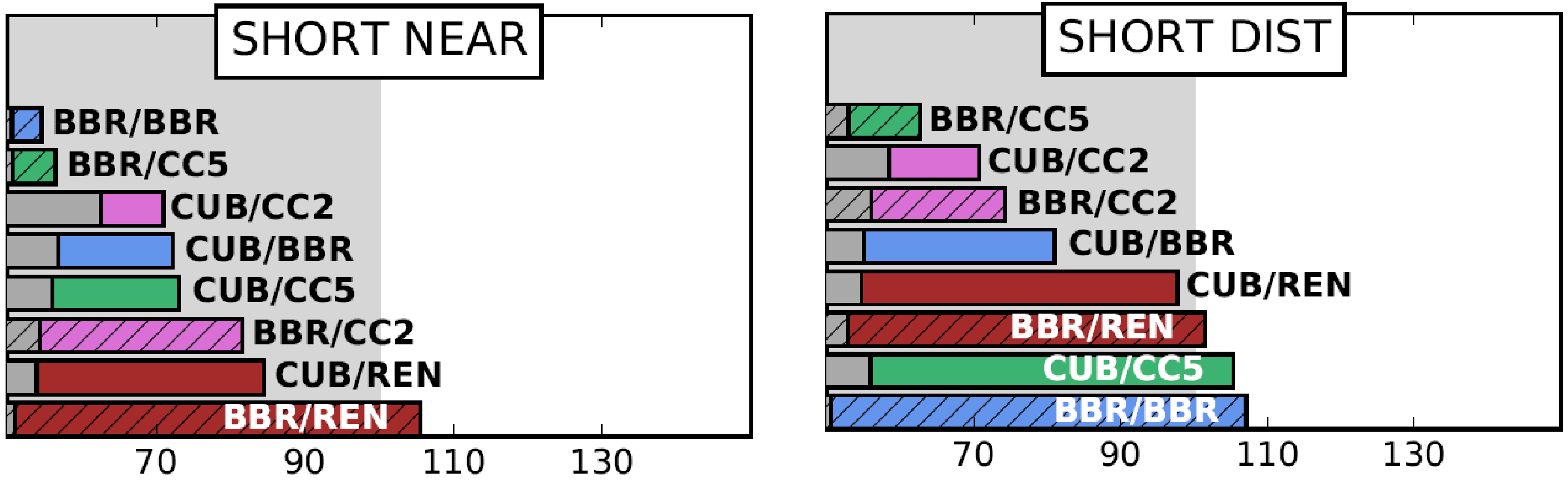
Flow completion time (FCT) as percentage of single path FCT



- BBR is performing better as a tunnel protocol, no loss and reacts faster
- BBR over Reno performs poorly here, BBR reacts before second path is used

Impact of proxy location

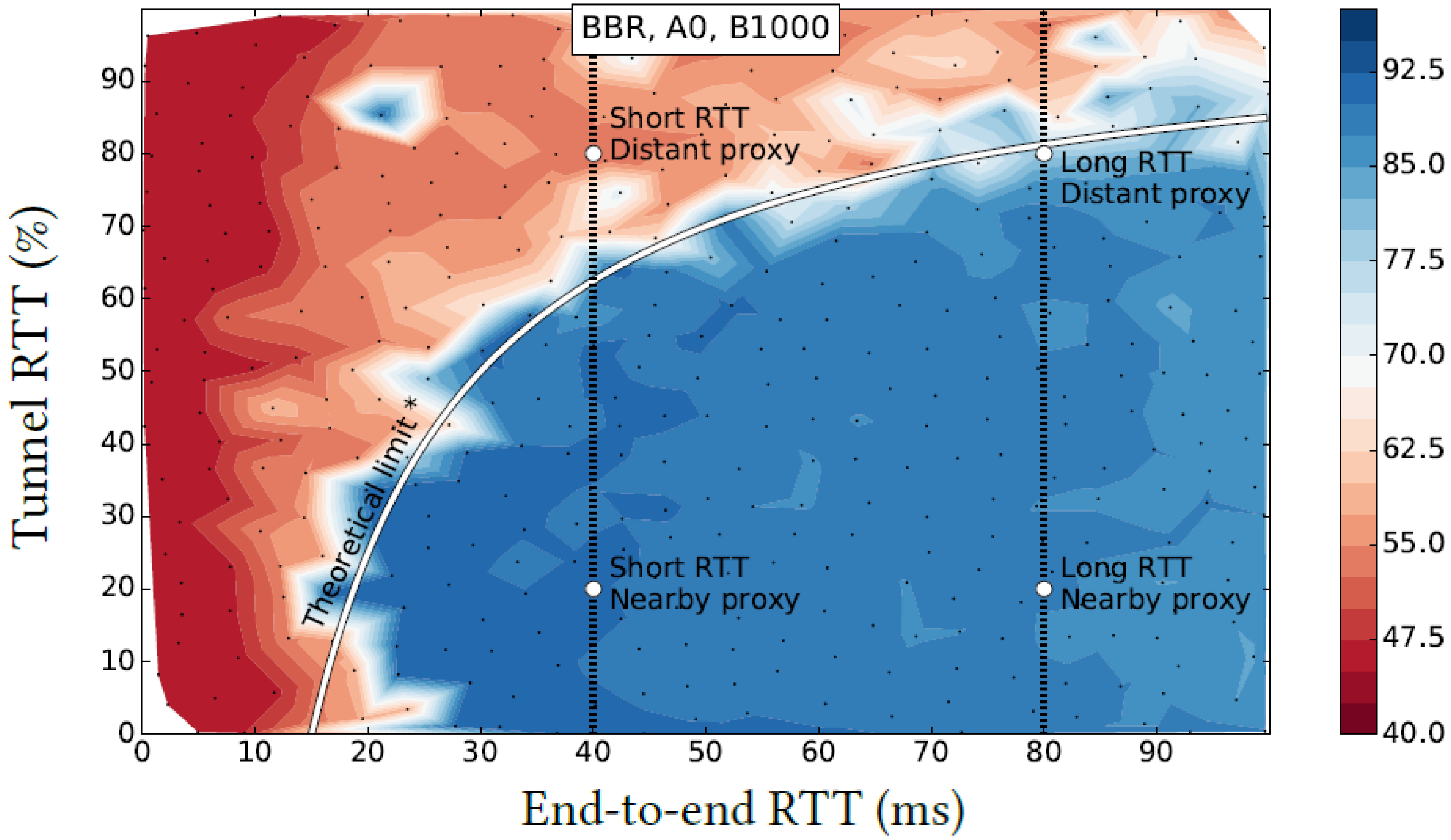
Flow completion time (FCT) as percentage of single path FCT



- Having the proxy closer (near) to the UE improves performance as this increases the difference between the tunnel RTT and E2E RTT

Impact of server (E2E RTT) and proxy location, detailed view for BBR over BBR

E2E throughput as a function of deployment scenarios



Summary

- Congestion control in congestion control has important performance impacts on multipath tunneling frameworks
- Complex interaction between many factors
  - Scheduling and reordering mechanism used
  - Choice of congestion controls
  - Placement of proxy functionality
  - Path characteristics
- Overall BBR performs better than Reno as CC for the tunnel
- Having the proxy close to the user is typically beneficial
- This is ongoing work and more results are coming…

Related drafts

- https://tools.ietf.org/html/draft-amend-tsvwg-multipath-dccp-04
- https://tools.ietf.org/html/draft-amend-tsvwg-multipath-framework-mpdccp-01
- https://tools.ietf.org/html/draft-amend-tsvwg-dccp-udp-header-conversion-01
- https://tools.ietf.org/html/draft-amend-iccrg-multipath-reordering-02
- https://tools.ietf.org/html/draft-bonaventure-iccrg-schedulers-01