MP-DCCP

for enabling transfer of UDP/IP traffic over multiple data paths in multi-connectivity networks

draft-amend-tsvwg-multipath-dccp-03
draft-amend-tsvwg-multipath-framework-mpdccp-01
draft-amend-tsvwg-dccp-udp-header-conversion-01

IETF 106 Meeting, TSVWG, Singapore, November 2019

KARLSTADS UNIVERSITET

CITY
UNIVERSITY OF LONDON
EST 1894

LIFE IS FOR SHARING.

MOTIVATION SUMMARIZED

1. 3GPP ATSSS and Hybrid Access require multipath network protocols, currently both relying on proprietary inflexible solutions or standardized MPTCP
2. The UDP share increases because of QUIC

→ A multipath solution for UDP, complementing the standardized MPTCP, is required and adressed in this proposal

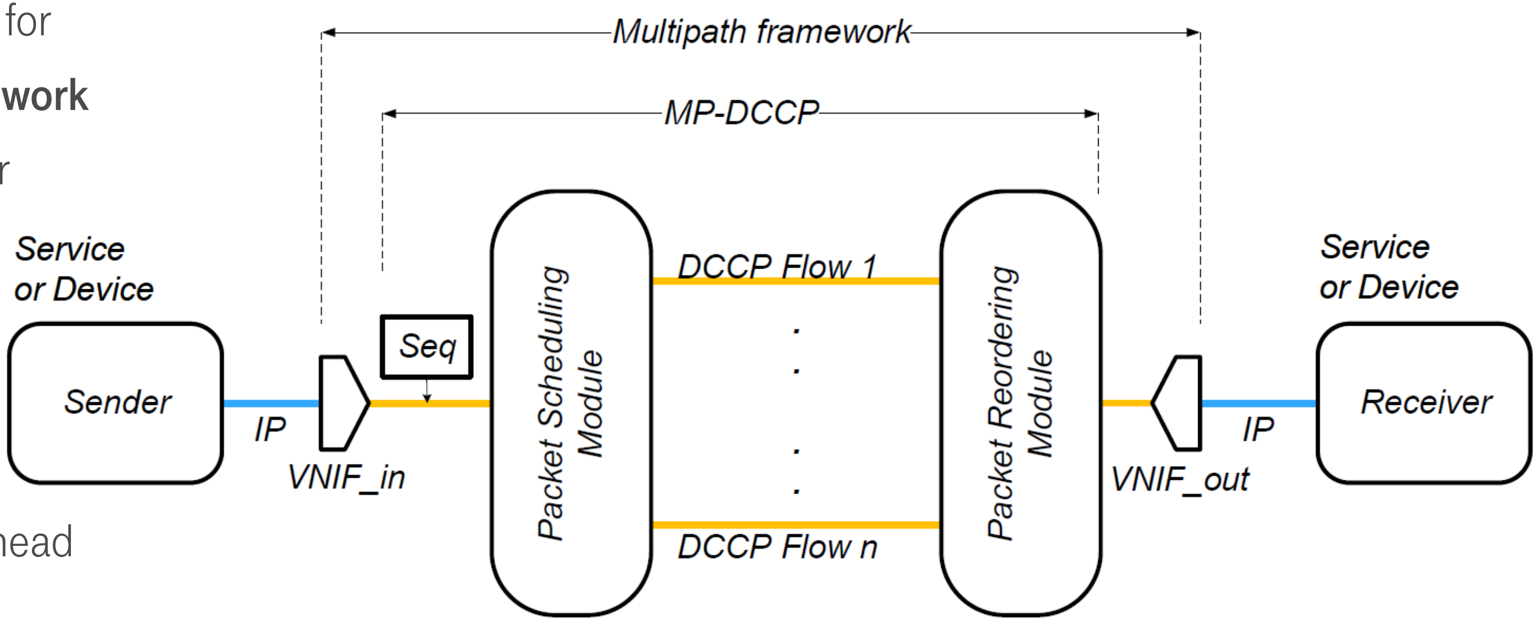
→ The basic DCCP protocol is selected due to its unreliable nature, however keeping a state and employs congestion control for path estimation purposes

Detailed motivation can be found in IETF104 presentation: https://datatracker.ietf.org/meeting/104/materials/slides-104-tsvwg-sessb-43-markus-amend-multipath-dccp-00

SOLUTION: MP-DCCP FOR UDP MULTIPATH TRANSMISSION

Making a **MP-DCCP** setup accessible for non DCCP traffic the **multipath framework** provides Virtual Network Interfaces for encapuslation purposes.

Successful DCCP transmission over uncontrolled networks and middleboxes can be ensured by overhead free **header conversion**.



https://tools.ietf.org/html/draft-amend-tsvwg-multipath-dccp-03

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

WORK SINCE IETF 105

Mainly worked on implementation and generating results in real world scenarios

Major updates to the MP-DCCP draft document, no updates on the others

NEW: https://tools.ietf.org/html/draft-amend-tsvwg-multipath-dccp-03; @Github

UNMODIFIED: https://tools.ietf.org/html/draft-amend-tsvwg-multipath-framework-mpdccp-01; @Github

UNMODIFIED: https://tools.ietf.org/html/draft-amend-tsvwg-dccp-udp-header-conversion-01; @Github

→ Updated MP-DCCP draft: Adding handshake procedure, DCCP feature for Multipath and Option for multipath purposes

→ BBR implementation for (MP-)DCCP

Next Steps

→ Work further on the MP-DCCP protocol mechanisms inside the draft

→ Align further the drafts with the 3GPP and BBF requirements

→ Elaborating the challenges of congestioned controlled traffic over MP-DCCP in agreed research collaborations with further companies

NEW: MP-DCCP PROTOCOL FEATURE AND OPTION

Number	Meaning	Rec'n Rule	Initial Value	Req'd
0	Reserved			
1	Congestion Control ID (CCID)	SP	2	Y
2	Allow Short Seqnos	SP	0	Y
3	Sequence Window	NN	100	Y
4	ECN Incapable	SP	0	N
5	Ack Ratio	NN	2	N
6	Send Ack Vector	SP	0	N
7	Send NDP Count	SP	0	N
8	Minimum Checksum Coverage	SP	0	N
9	Check Data Checksum	SP	0	N
10	Multipath Capable	SP	0	N
11-127	Reserved			
128-255	CCID-specific features			

Figure 3: Proposed Feature Set

Multipath Capable feature for negotiation during handshake

Type	Option Length	Meaning	DCCP-Data?
0	1	Padding	Y
1	1	Mandatory	N
2	1	Slow Receiver	Y
3-31	1	Reserved	
32	variable	Change L	N
33	variable	Confirm L	N
34	variable	Change R	N
35	variable	Confirm R	N
36	variable	Init Cookie	N
37	3-8	NDP Count	Y
38	variable	Ack Vector [Nonce 0]	N
39	variable	Ack Vector [Nonce 1]	N
40	variable	Data Dropped	N
41	6	Timestamp	Y
42	6/8/10	Timestamp Echo	Y
43	4/6	Elapsed Time	N
44	6	Data Checksum	Y
45	variable	Multipath	Y
46-127	variable	Reserved	
128-255	variable	CCID-specific options	-

Figure 4: Proposed Option Set

Multipath option to exchange multipath specific information

NEW: MP-DCCP PROTOCOL MULTIPATH OPTIONS

Multipath options for

- setting up a MP-DCCP connection securely
- adding or removing subflows securely
- enable receiver side re-ordering
- Giving path priority

```
+--------+--------+--------+--------+--------
  |00101101| Length | MP_OPT | Value(s) ...
  +--------+--------+--------+--------+--------
   Type=45

         Option
   Type  Length  MP_OPT        Meaning
   ----  ------  -------       -----
    45     7     0 =MP_CONFIRM Confirm reception and processing of
                               an MP_OPT option
    45     7     1 =MP_JOIN    Join path to an existing MP-DCCP flow
    45     3     2 =MP_FAST_CLOSE Close MP-DCCP flow
    45     var   3 =MP_KEY     Exchange key material for MP_HMAC
    45     7     4 =MP_SEQ     Multipath Sequence Number
    45     23    5 =MP_HMAC    HMA Code for authentication
    45     12    6 =MP_RTT     Transmit RTT values
    45     TBD   7 =MP_ADDADDR    TBD
    45     TBD   8 =MP_REMOVEADDR TBD
    45     TBD   9 =MP_PRIO       TBD

                   Figure 5: MP_OPT Option Types
```

NEW: MP-DCCP PROTOCOL HANDSHAKE PROCEDURE

Initial flow setup negotiating **MP_Capable** feature and **MP_KEY** option with host specific Keys.

A successful handshake make both sides aware of the host specific key material.

Following the initial path setup, subsequent flows can be established, negotiating again **MP_Capable** and using **MP_JOIN** option to merge paths with the initial path. Token (TB) generated from the key material, random nonces (RA/RB) and **HMAC** secure the process.

The session establishment and the security concept is similar to MPTCP.

```
         Host A                                               Host B
------------------------                                    ----------
Address A1     Address A2                                   Address B1
----------     ----------                                   ----------
    |              |                                             |
    |   DCCP-Request + MP_CAPABLE                                |
    |------- MP_KEY(Key-A) --------------------------------->|
    |<---------------------- MP_KEY(Key-B) ---------------|
    |   DCCP-Response + MP_CAPABLE agreed                        |
    |              |                                             |
    |   DCCP-Ack   |                                             |
    |--------- MP_KEY(Key-A) + MP_KEY(Key-B) ------------>|
    |              |                                             |
    |              |DCCP-Request + MP_CAPABLE                    |
    |              |--- MP_JOIN(TB,RA) ------------------->|
    |              |<------MP_JOIN(TB,RB) + MP_HMAC(A)-----|
    |              |DCCP-Response                                |
    |              |                                             |
    |              |DCCP-Ack                                     |
    |              |-------- MP_HMAC(B) ------------------->|
    |              |<----------------------------------------|
    |              |DCCP-ACK                                     |
```

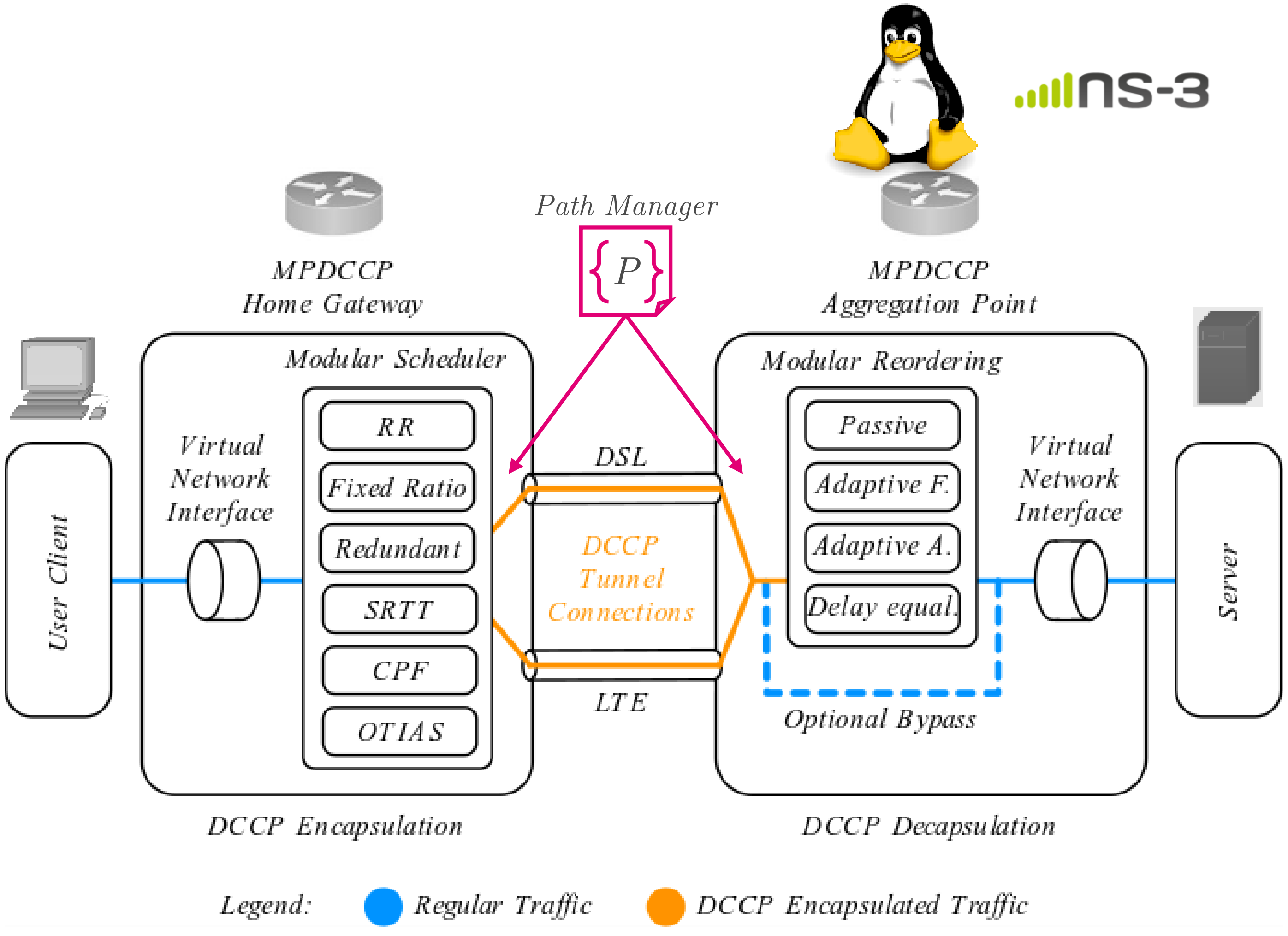
Figure 7: Example MP-DCCP Handshake

ANALYSIS AND RESULTS – TESTBED AND NS3 SIMULATIONS

Prototype is available inside Linux Kernel and ns-3 for residential and mobile use case each

- support seamless handover and path aggregation
- modular scheduler for distributing traffic
- modular re-assembly to compensate latency differences
- modular path manager to establish DCCP flows dynamically
- DCCP-UDP conversion to connect through non-DCCP aware middleboxes

→ Analysis Objective – test the ability of the framework to improve/maintain QoS/QoE on volatile paths

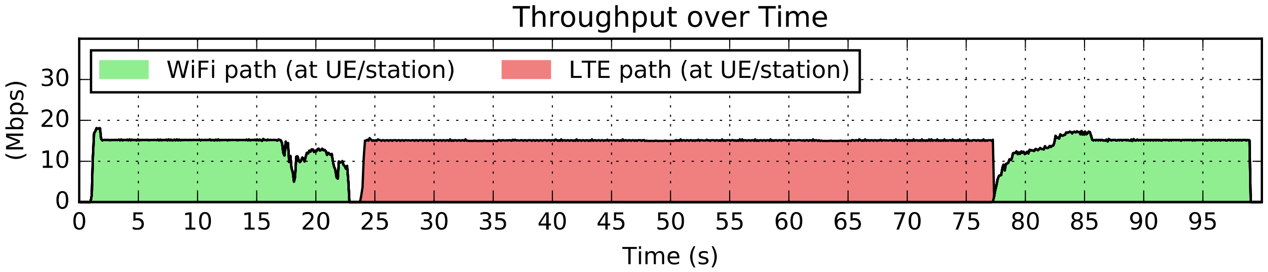


https://arxiv.org/pdf/1907.04567.pdf

IETF105 ICCRG presentation with more results

REMEMBER: SWITCHING AND AGGREGATION– NS3, UDP TRAFFIC

Switching in case of path failure or bad path conditions

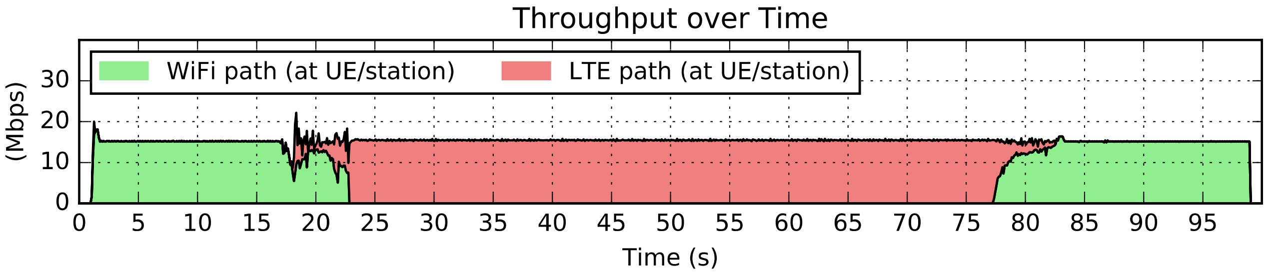


After detecting the physical loss, stream is handed over to cellular without connectivity brea

When WiFi returns, stream is handed over to Wi-Fi again

and aggregation, simultaneous path usage

is supported



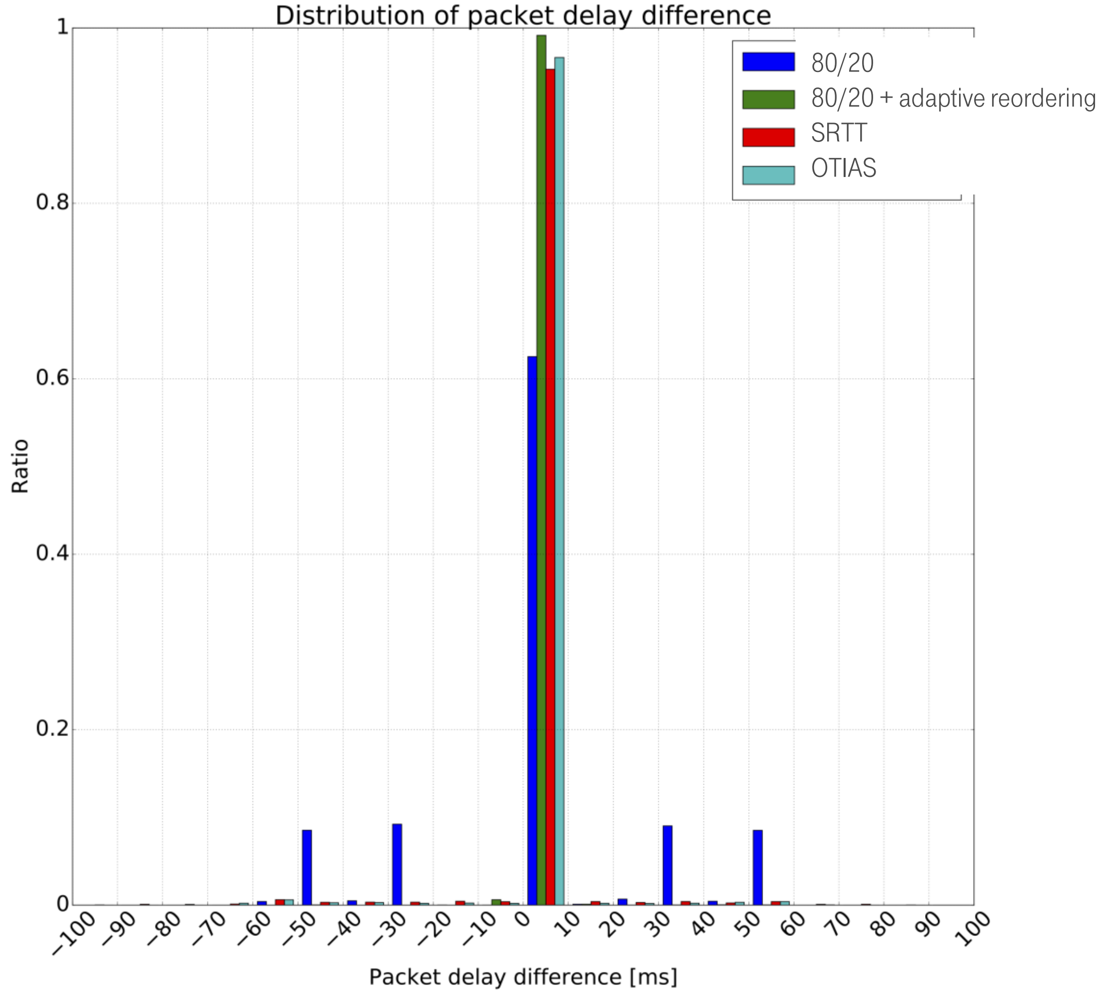
Additionaly to the scenario on top, using path aggregation combined with path prioritzation on WiFi enables a smooth handover, keeping QoS stable

Compare:
https://datatracker.ietf.org/meeting/105/materials/slides-105-tsvwg-sessa-62-dccp-extensions-for-multipath-operation

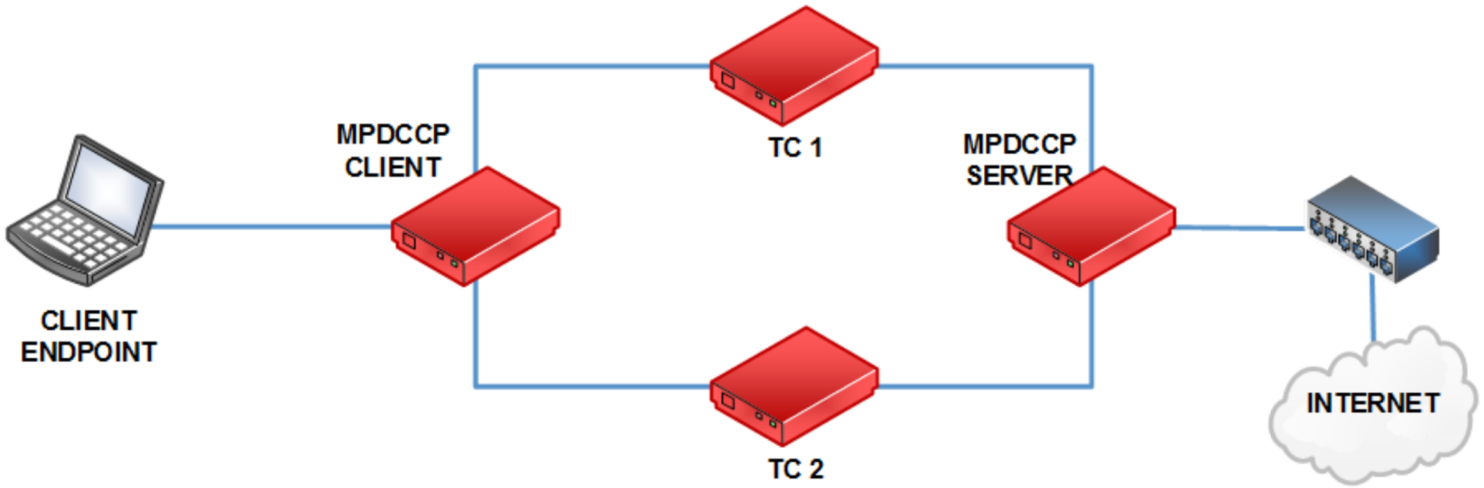
REMEMBER: MANAGING PACKET DELAY VARIATION USING SCHEDULING OR REORDERING

Path heterogeneity requires in practice a receiver side re-ordering when paths are simultaneously used to compensate latency differences.

Compare:
https://datatracker.ietf.org/meeting/105/materials/slides-105-tsvwg-sessa-62-dccp-extensions-for-multipath-operation



NEW: REAL WORLD RESULTS WITH YOUTUBE (QUIC) 1/2



- Chrome browser (QUIC enabled)
- Embedded YouTube player
- Skipping to unbuffered part of video at 10, 30, 50, 70, 90, 110s
  - Forcing buffering (stalling)
  - Always at the same unbuffered parts of the video

- Network conditions change at 60s - more on next slide
- Total duration 120s
- Playback ratio = Playing time/Total time (120s)

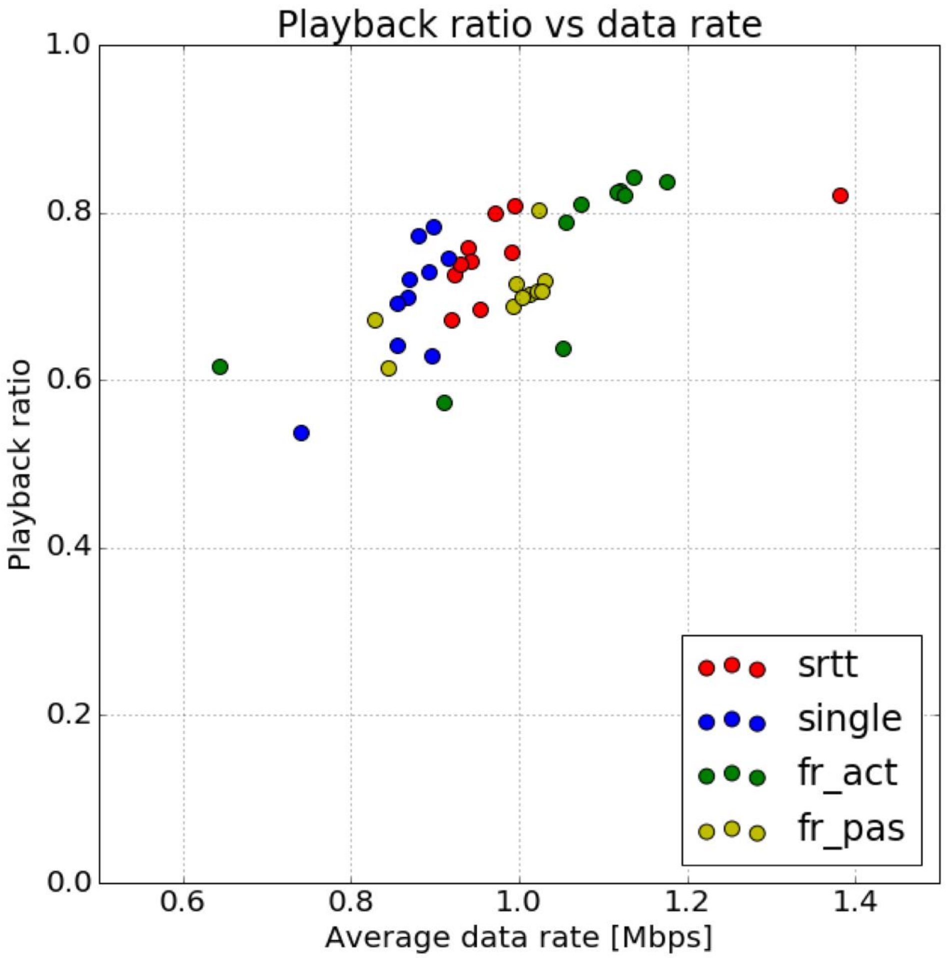
  Always < 1 due to initial loading and skipping

NEW: REAL WORLD RESULTS WITH YOUTUBE (QUIC) 2/2

Path		Bandwidth	Latency
1	$t \leq 60s$	1Mbps	10ms
	$t > 60s$	1Mbps	90ms
2		1 Mbps	50 ms

The **highest gain could be reached by path aggregation** using the fixed ratio scheduler (80:20) combined with re-ordering over the one without re-ordering and srtt scheduler **compared to no aggregation using single path**.

However, more tests will follow including more path heterogeneity and volatility. The brand new BBR implementation promises first good results...



CONCLUSION & NEXT STEPS

The prototype implementation and simulation show very good first results according to the demands of Steering, Switching and Splitting of 3GPP ATSSS and BBF Hybrid Access.

Key components identified for efficient multipath: scheduling, congestion control, re-ordering

Possible interference of MP-DCCP congestion control with congestion control of piggybacked traffic adressed at ICCRG

Linux Kernel BBR implementation for MP-DCCP prototype/simulation available, results expected for IETF107

Discussions with operators and vendors have been initiated, however additional support is always welcome.

MP-DCCP is an on purpose protocol, unlike (MP-)QUIC, dedicated and simple for multi-connectivity architectures. This promises a rapid development and standardization if pushed jointly.

Drafts require further action to get some maturity, make your points at Github: MP-DCCP, Framework, DCCP/UDP conversion

Please use tsvwg@ietf.org or markus.amend@telekom.de to get in touch with us.

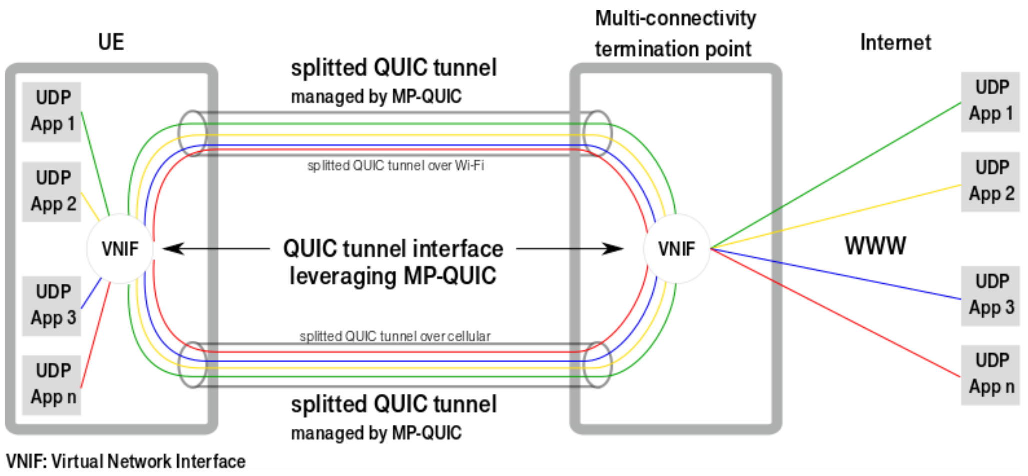
Further documents

Paper with detailed results: https://arxiv.org/pdf/1907.04567.pdf, IETF 104 presentation, IETF 105 ICCRG presentation

BACKUP

WHY NOT USE MP-QUIC INSTEAD OF MP-DCCP?

(MP-)QUIC according to https://tools.ietf.org/html/draft-ietf-quic-transport-24 is a reliable and end-to-end encrypted protocol. Its application for enabling multipath transfer for UDP/QUIC traffic only works as QUIC tunnel, managed by MP-QUIC.



- **Useless encryption is applied and requires resources**
  - UDP as guest: Turns UDP into reliable transmission ☹
    - QUIC as guest: Encryption over Encryption, otherwise like TCP below ☹☹
  - TCP as guest: TCP's CC + reliable in-order delivery over outer QUIC's CC + reliable in-order delivery ☹☹☹

REQUIRED (MP-)QUIC ADAPTATIONS

In case MP-QUIC shall become an alternative for ATSSS and Hybrid Access like network architectures, it would require a paradigm change:

→ Configurable encryption for

- reducing the useless overhead in case of QUIC over MP-QUIC (likely)
- designing a MP-QUIC ↔ QUIC converter (unlikely)

→ Deal with unreliable traffic to some extent and remove at least the reliable and in-order delivery feature

- Unreliable traffic support requires a complete re-work of current MP-QUIC framework, which bases on QUICs reliable and in-order delivery.

→ Define a QUIC tunnel protocol