Multipath QUIC

Quentin De Coninck

https://multipath-quic.org
Outline

- Designing Multipath for (G)QUIC
- Evaluating Multipath Benefits
- Adapting to IETF QUIC
- Open Challenges and Opportunities
Why do we want Multipath?
Why Multipath QUIC?

- QUIC assumes a single-path flow
Why Multipath QUIC?

- QUIC assumes a single-path flow
Why Multipath QUIC?

- QUIC assumes a single-path flow
Why Multipath QUIC?

- QUIC assumes a single-path flow
- Multipath QUIC
Why Multipath QUIC?

- QUIC assumes a single-path flow

- Multipath QUIC
  - Bandwidth aggregation
Why Multipath QUIC?

- QUIC assumes a single-path flow

Multipath QUIC
- Bandwidth aggregation
- Seamless network handover
  - Can try new WiFi while keeping using LTE
Hem, connection migration?
Hem, connection migration?

- **Connection ID(s) to identify flow**
  - Resilient to 4-tuple change (IP, port)
Hem, connection migration?

• Connection ID(s) to identify flow
  – Resilient to 4-tuple change (IP, port)

• IETF QUIC probes “paths”
  – PATH_CHALLENGE / PATH_RESPONSE
Hem, connection migration?

- Connection ID(s) to identify flow
  - Resilient to 4-tuple change (IP, port)
- IETF QUIC probes “paths”
  - PATH_CHALLENGE / PATH_RESPONSE
- Required mechanisms for multipath
- But no simultaneous usage of paths for data exchange
Disclaimer

• In the remaining of this section, only (old) Google QUIC version is explained

• The IETF version has its dedicated section :-)

14
Designing Multipath (G)QUIC

• Connection is composed of paths
  – After handshake completion
Designing Multipath (G)QUIC

- Connection is composed of paths
  - After handshake completion
Designing Multipath (G)QUIC

- Connection is composed of paths
  - After handshake completion
Designing Multipath (G)QUIC

- Connection is composed of paths
  - After handshake completion

Performance monitoring?
Loss detection?
Path congestion control?
Designing Multipath (G)QUIC

- **Connection is composed of paths**
  - After handshake completion
Designing Multipath (G)QUIC

- **Connection is composed of paths**
  - After handshake completion

Flags | Connection ID | Path ID | Packet Number | Encrypted Payload...
Designing Multipath (G)QUIC

- **Connection is composed of paths**
  - After handshake completion

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
<th>Path ID</th>
<th>Packet Number</th>
<th>Encrypted Payload...</th>
</tr>
</thead>
</table>

Same CID
Designing Multipath (G)QUIC

- **Connection is composed of paths**
  - After handshake completion

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
<th>Path ID</th>
<th>Packet Number</th>
<th>Encrypted Payload...</th>
</tr>
</thead>
</table>

Explicit path identification

Same CID
Designing Multipath (G)QUIC

- Connection is composed of paths
  - After handshake completion

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
<th>Path ID</th>
<th>Packet Number</th>
<th>Encrypted Payload...</th>
</tr>
</thead>
</table>

Explicit path identification

Same CID

Per-path numbering space
Architecture of Multipath (G)QUIC

Connection (Connection ID)

Stream & Frame Management

Path A (PathID A)
RTT, # pkt lost,…
Packet Number

Path B (PathID B)
RTT’, # pkt lost’,…
Packet Number’

Path C (PathID C)
RTT”, # pkt lost”,…
Packet Number”
Multipath QUIC Data Transfer

Server via WiFi

Path 1: WiFi

Phone

Path 2: LTE

Server via LTE
Multipath QUIC Data Transfer

Path 1: WiFi

Path 2: LTE
Multipath QUIC Data Transfer

Server via WiFi

F CID 1 PN=1 STR(id=5)

F CID 1 PN=1 STR(id=7,off=0)

Path 1: WiFi

Server via LTE

F CID 2 PN=1 STR(id=7,off=1024)

Path 2: LTE
Multipath QUIC Data Transfer

Path 1: WiFi
- Frame: F, CID: 1, PN: 1, STR(id=5)
- Frame: F, CID: 1, PN: 2, ACK(pid=1,1), ACK(pid=2,1)

Path 2: LTE
- Frame: F, CID: 1, PN: 1, STR(id=7,off=0)
- Frame: F, CID: 2, PN: 1, STR(id=7,off=1024)
Multipath QUIC Data Transfer

Path 1: WiFi

Frames not constrained to a particular path

Path 2: LTE
Multipath Mechanisms

- Path management
- Packet scheduling
- Congestion control scheme
Path Management

• How and when paths are established?
Path Management

• How and when paths are established?
Path Management

- How and when paths are established?
Path Management

• How and when paths are established?

• ADD_ADDRESS + REMOVE_ADDRESS frames
Path Management

• How and when paths are established?

• ADD_ADDRESS + REMOVE_ADDRESS frames

• Full-mesh fashion
Packet Scheduling

- Lowest-latency first
Packet Scheduling

- Lowest-latency first

![Diagram showing two RTTs: 20 ms and 10 ms.]
Packet Scheduling

- Lowest-latency first

![Diagram showing packet scheduling]

- 20 ms RTT
- 10 ms RTT
Packet Scheduling

- Lowest-latency first

- What if the path latency is unknown?
Packet Scheduling

- Lowest-latency first
- What if the path latency is unknown?
Packet Scheduling

- Lowest-latency first

- What if the path latency is unknown?

- Schedule ALL frames (not only data)
Congestion Control Scheme

- **Multipath = need for coupled CC**
  - CUBIC would be unfair

- **Opportunistic Linked Increase Algorithm**
  - MPTCP state-of-the-art
Evaluating Multipath Benefits
Evaluation of Multipath QUIC

- **(Multipath) QUIC vs. (Multipath) TCP**
  - Multipath QUIC: based on *quic-go*
  - Linux MPTCP v0.91 with default settings
- **Mininet environment with 2 paths**
Evaluating Bandwidth Aggregation

CoNEXT’17

• 20 MB Download
  - Over a single stream
  - Collect the transfer time
• 20 MB Download
  - Over a single stream
  - Collect the transfer time

• Experimental design, WSP algorithm
Evaluating Bandwidth Aggregation

20 MB Download
- Over a single stream
- Collect the transfer time

Experimental design, WSP algorithm

2x253 network scenarios
- Vary the initial path

<table>
<thead>
<tr>
<th>Factor</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity [Mbps]</td>
<td>0.1</td>
<td>100</td>
</tr>
<tr>
<td>Round-Trip-Time [ms]</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>Queuing Delay [ms]</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>Random Loss [%]</td>
<td>0</td>
<td>2.5</td>
</tr>
</tbody>
</table>
Actual Multipath Benefit

- Experimental Aggregation Benefit
  - Multipath QUIC/TCP vs. single-path QUIC/TCP
Actual Multipath Benefit

- **Experimental Aggregation Benefit**
  - Multipath QUIC/TCP vs. single-path QUIC/TCP

-1

- Zero goodput
  - MP gives 0 Mbps

0

- = best single path
  - 3 Mbps + 5 Mbps paths
  - MP gives 5 Mbps

1

- = aggregation of all paths
  - 3 Mbps + 5 Mbps paths
  - MP gives 8 Mbps
Actual Multipath Benefit

• **Experimental Aggregation Benefit**
  - Multipath QUIC/TCP vs. single-path QUIC/TCP

-1
Zero goodput
MP gives 0 Mbps

0
3 Mbps + 5 Mbps paths
MP gives 5 Mbps

1
3 Mbps + 5 Mbps paths
MP gives 8 Mbps

• **Results depends on the first path used**
  - Handshake latency over initial path
Benefits of Multipath - No Loss

CoNEXT’17

GET 20 MB, 253 scenarios low-BDP-no-loss

- 48% for MPTCP vs. TCP
- 77% for MPQUIC vs. QUIC

Ex. Aggregation Benefit

Protocol

MPTCP vs. TCP

MPQUIC vs. QUIC

Best path first
Worst path first
Benefits of Multipath - No Loss

CoNEXT’17

% scenarios multipath has EAB >= 0, regardless of first path used

GET 20 MB, 253 scenarios low-BDP-no-loss

Exp. Aggregation Benefit

1.5

1.0

0.5

0.0

-0.5

-1.0

-1.5

MPTCP vs. TCP

MPQUIC vs. QUIC

Protocol

48%

77%

Best path first

Worst path first
Benefits of Multipath - Losses

CoNEXT’17

GET 20 MB, 253 scenarios low-BDP-losses

Exp. Aggregation Benefit

MPTCP vs. TCP

MPQUIC vs. QUIC

Protocol

32%
62%

-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5

Best path first
Worst path first
And with Real Networks?

Real World WiFi/LTE Measurement

Download Time [s]

QUIC
MPQUIC

24MB

QUIC
MPQUIC

100MB
Network Handover Support

- Apple MPTCP deployment mainly for handover
  - Main use case for Siri
Network Handover Support

- Apple MPTCP deployment mainly for handover
  - Main use case for Siri

- Request/Response traffic
  - 750 bytes request/responses every 1/3 s
  - Measure delay seen by client
    
    15ms RTT, 100% loss after 3 s
    Path 1

    25ms RTT
    Path 2

    Client
    Router 1
    Router 2
    Server
Multipath TCP Handover
Multipath TCP Handover

![Graph showing delay to answer request vs sent time with a peak at 4 seconds for MPTCP.]
MPTCP Handover Explained

![Diagram showing MPTCP handover with 15 ms and 25 ms delays]
MPTCP Handover Explained

Req

15 ms

25 ms
MPTCP Handover Explained

15 ms

25 ms
MPTCP Handover Explained

RTO

Req

15 ms

25 ms
MPTCP Handover Explained

RTO

15 ms

25 ms

Req
MPTCP Handover Explained

RTO

15 ms

25 ms

Res
MPTCP Handover Explained

RTO

15 ms

25 ms

4G

Wi-Fi

Internet Gateway
MPTCP Handover Explained

RTO

15 ms

25 ms

Res

RTO
MPTCP Handover Explained
What about Multipath QUIC?
What about Multipath QUIC?
How is it possible?
How is it possible?

FCID 1  PN  STR(Req)

15 ms

25 ms
How is it possible?
How is it possible?
How is it possible?
How is it possible?
How is it possible?
Adapting to IETF QUIC
Issues with (G)QUIC Design

- **Path ID in clear-text public header**
  - Easy to correlate paths :-(

- **IETF QUIC changed a lot in ~2 years**
  - GQUIC very different from current IETF version
  - Source/Destination Connection IDμ

- **Core idea: use CIDs as implicit path ID**
Negotiating Multipath Usage

Server path 1

Client

Server path 2
Negotiating Multipath Usage

Server path 1
CHLO(TP"max_paths" = 20)

Client

SHLO(TP"max_paths" = 2)

Server path 2
Negotiating Multipath Usage

Client

Server path 1

CHLO(TP”max_paths” = 20)

SHLO(TP”max_paths” = 2)

NEW_CID(Path ID=1, CID A)
NEW_CID(Path ID=2, CID B)

NEW_CID(Path ID=1, CID C)
NEW_CID(Path ID=2, CID D)

Server path 2
Negotiating Multipath Usage

Server path 1

CHLO(TP"max_paths" = 20)

SHLO(TP"max_paths" = 2)

NEW_CID(Path ID=1, CID A)
NEW_CID(Path ID=2, CID B)

NEW_CID(Path ID=1, CID C)
NEW_CID(Path ID=2, CID D)

Pkt(CID A, ...)

Client

Server path 2

Pkt(CID B, ...)
Negotiating Multipath Usage

Server path 1
CHLO(TP"max_paths" = 20)

SHLO(TP"max_paths" = 2)

NEW_CID(Path ID=1, CID A)
NEW_CID(Path ID=2, CID B)

NEW_CID(Path ID=1, CID C)
NEW_CID(Path ID=2, CID D)

Pkt(CID A, ...)

Pkt(CID C, ...)

Client

Server path 2

Pkt(CID B, ...)

Pkt(CID D, ...)
Architecture of Multipath QUIC

Connection

(Master Source Connection ID, Master Destination Connection ID)

Stream & Frame Management

Path A (PathID A)
Path SCID A
Path DCID A’
RTT, # pkt lost,...
Packet Number

Path B (PathID B)
Path SCID B
Path DCID B’
RTT’, # pkt lost’,...
Packet Number’

Path C (PathID C)
Path SCID C
Path DCID C’
RTT”, # pkt lost”,...
Packet Number”

MPQUIC-ID
Summary of Changes

- Transport parameter for MP usage
- Wait end of handshake before MP usage
- Adding PathID varint in frames
  - NEW_CONNECTION_ID
  - RETIRE_CONNECTION_ID
  - ACK
- New frames
  - ADD_ADDRESS
  - REMOVE_ADDRESS
  - MAX_PATHS
  - PATH_UPDATE
Open Challenges and Opportunities
Multipath Scheduling

- Multiple paths choice, like MPTCP...
- ... but with more than 1 stream!
Multipath Scheduling

- Multiple paths choice, like MPTCP...
- ... but with more than 1 stream!

A Stream-Aware Multipath QUIC Scheduler for Heterogeneous Paths

Paper # XXX, XXX pages

Alexander Rabitsch
Karlstad University
alexander.rabitsch@kau.se

Per Hurtig
Karlstad University
per.hurtig@kau.se

Anna Brunstrom,
Karlstad University
anna.brunstrom@kau.se
Congestion Control Scheme

• How to remain fair but efficient?
  – And with multipath?
Congestion Control Scheme

• How to remain fair but efficient?
  - And with multipath?
Congestion Control Scheme

• How to remain fair but efficient?
  – And with multipath?

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

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Flow</th>
<th>Avg. throughput (std. dev.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUIC vs. TCP</td>
<td>QUIC</td>
<td>2.71 (0.46)</td>
</tr>
<tr>
<td></td>
<td>TCP</td>
<td>1.62 (1.27)</td>
</tr>
<tr>
<td>QUIC vs. TCPx2</td>
<td>QUIC</td>
<td>2.8 (1.16)</td>
</tr>
<tr>
<td></td>
<td>TCP 1</td>
<td>0.7 (0.21)</td>
</tr>
<tr>
<td></td>
<td>TCP 2</td>
<td>0.96 (0.3)</td>
</tr>
<tr>
<td>QUIC vs. TCPx4</td>
<td>QUIC</td>
<td>2.75 (1.2)</td>
</tr>
<tr>
<td></td>
<td>TCP 1</td>
<td>0.45 (0.14)</td>
</tr>
<tr>
<td></td>
<td>TCP 2</td>
<td>0.36 (0.09)</td>
</tr>
<tr>
<td></td>
<td>TCP 3</td>
<td>0.41 (0.11)</td>
</tr>
<tr>
<td></td>
<td>TCP 4</td>
<td>0.45 (0.13)</td>
</tr>
</tbody>
</table>

Table 4: Average throughput (5 Mbps link, buffer~30 KB, averaged over 10 runs) allocated to QUIC and TCP flows when competing with each other. Despite the fact that both protocols use Cubic congestion control, QUIC consumes nearly twice the bottleneck bandwidth than TCP flows combined, resulting in substantial unfairness.
Handover situations

• How does Multipath QUIC help under mobility in wireless environment?
  – Especially with ≠ path priorities
Handover situations

• How does Multipath QUIC help under mobility in wireless environment?
  – Especially with ≠ path priorities

Asymmetric Paths

- So far, assume that paths are symmetric...
- ... what if they are not?
And All The Others...

• Don’t hesitate to discuss your own challenges :-)

Thanks for your attention!
References

Backup slides
(MP)TCP vs. (MP)QUIC - No Loss
CoNEXT’17

GET 20 MB, 506 simulations low-BDP-no-loss

CDF

Time TCP / QUIC
(MP)TCP vs. (MP)QUIC - No Loss

CoNEXT'17

GET 20 MB, 506 simulations low-BDP-no-loss

CDF

Time TCP / QUIC

Single-path
(MP)TCP vs. (MP)QUIC – No Loss

CoNEXT’17

GET 20 MB, 506 simulations low-BDP-no-loss

Time TCP / QUIC

Single-path

TCP better

QUIC better
(MP)TCP vs. (MP)QUIC - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss
(MP)TCP vs. (MP)QUIC – No Loss

CoNEXT’17

GET 20 MB, 506 simulations low-BDP-no-loss

MPQUIC better than MPTCP in 85% of cases
(MP)TCP vs. (MP)QUIC - No Loss

CoNEXT’17

MPQUIC better than MPTCP in 85% of cases

Time TCP / QUIC
Time MPTCP / MPQUIC

GET 20 MB, 506 simulations low-BDP-no-loss

Path 1: 49.4 ms RTT, 18.90 Mbps, 82 ms queuing delay
Path 2: 10.6 ms RTT, 0.43 Mbps, 11 ms queuing delay
(MP)TCP vs. (MP)QUIC – No Loss

CoNEXT’17

GET 20 MB, 506 simulations low-BDP-no-loss

MPQUIC better than MPTCP in 85% of cases

Path 1: 27.2 ms RTT, 0.14 Mbps, 34 ms queuing delay
Path 2: 46.4 ms RTT, 49.72 Mbps, 47 ms queuing delay
Path 1: 49.4 ms RTT, 18.90 Mbps, 82 ms queuing delay
Path 2: 10.6 ms RTT, 0.43 Mbps, 11 ms queuing delay
(MP)TCP vs. (MP)QUIC – Losses

CoNEXT’17

GET 20MB, 506 simulations, low-BDP-losses

CDF

Time Ratio

1.0
0.8
0.6
0.4
0.2
0.0
Time TCP / QUIC
Time MPTCP / MPQUIC
(MP)TCP vs. (MP)QUIC – **Losses**

**CoNEXT’17**

**GET 20MB, 506 simulations, low-BDP-losses**

- **Time TCP / QUIC**
- **Time MPTCP / MPQUIC**

**QUIC better handles losses**
(MP)TCP vs. (MP)QUIC - **Losses**

**CoNEXT’17**

**GET 20MB, 506 simulations, low-BDP-losses**

- **QUIC better handles losses**
  - **TCP SACK**: 2-3 blocks
  - **QUIC SACK**: limited to available packet space
Short Files, Multipath Less Useful...

GET 256 KB, 253 scenarios low-BDP-no-loss

Exp. Aggregation Benefit

16%

5%

MPTCP vs. TCP

MPQUIC vs. QUIC

Best path first

Worst path first