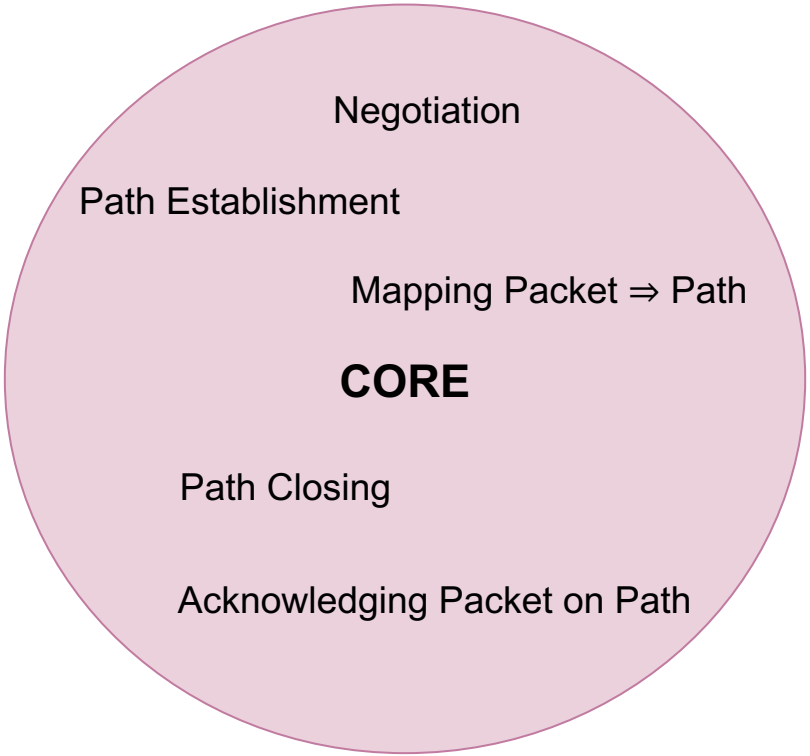
Comparing MPQUIC Proposals

Focus on Core Components

Quentin De Coninck
October 18, 2021

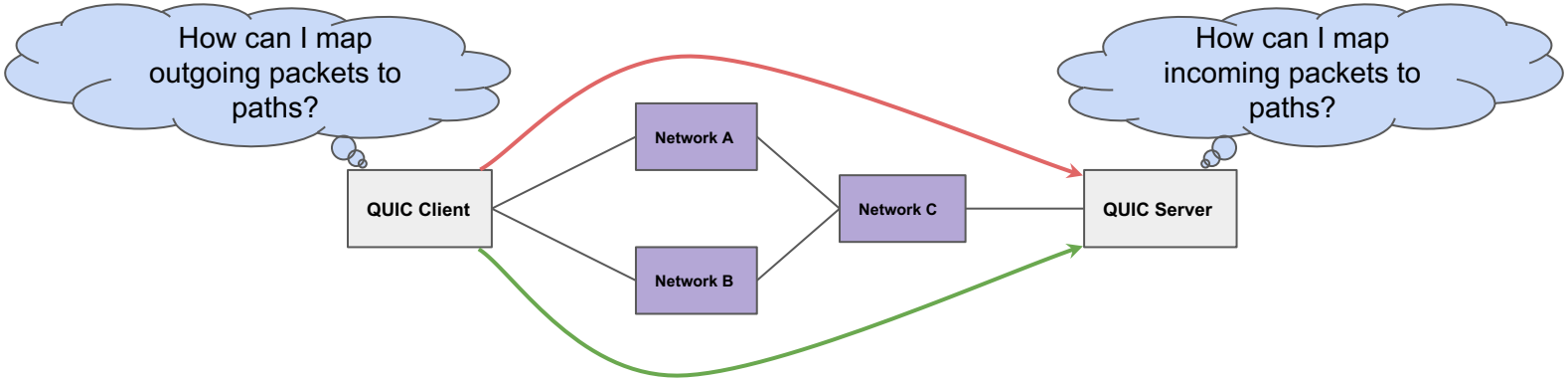
Components of the Multipath Core



Common Points Between All Drafts

- Multipath negotiation through QUIC transport parameters
- Multipath usage only for 1RTT packets
  - Initial and Handshake Packet Number Spaces untouched
- Path validation process before using a new path
- Frames/packets can be spread over any “active path”
- Sender maintains congestion control state per path

Establishing Paths and Mapping Packets to Paths

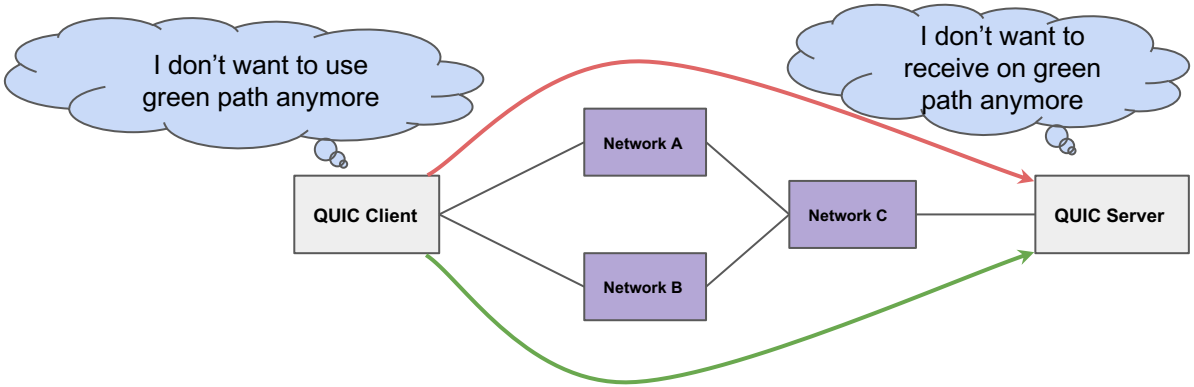


- Path validation process = path establishment
  - Use different Connection IDs for different 4-tuples
  - If process succeeds ⇒ path can be used (“active path”)
    - Several “active paths” can be used concurrently

A Note on Path Bidirectionality

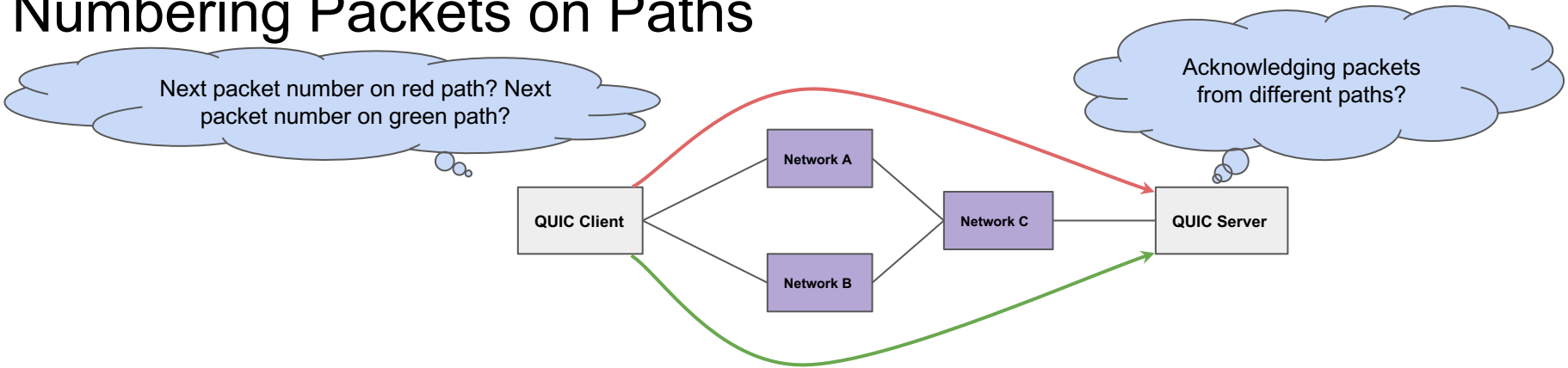
- Most drafts focuses on using bidirectional paths
  - As it builds on (bidirectional) path validation
  - huitema-01, deconinck-02, liu-04
- deconinck-07 proposes unidirectional paths
  - But requires address exchange (if S -> C) and modifying path validation process
  - → Future extension, not part of multipath core

Closing Paths



- Client stops sending, server notices no usage after some time
  - huitema-01 in simple multipath mode
- Explicit frame requesting closure of paths
  - E.g., a receiver wanting to close a path
  - liu-04, (deconinck-02, deconinck-07)

Numbering Packets on Paths



- One Application Data Packet Number Space for all paths, ACK frame acknowledges packets
  - huitema-01
- One Application Data Packet Number Space for each path, MP_ACK/ACK_MP frame acknowledges packets
  - liu-04, deconinck-02, deconinck-07