A MULTIPATH EXTENSION TO THE QUIC MODULE IN NS-3

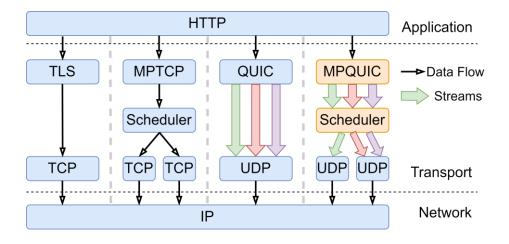
Shengjie Shu





CURRENT PROBLEMS & MOTIVATION





MPTCP:

- Throughput Aggregation
- Congestion Shift

Issues:

- Connection Breakage
- Head-of-Line Blocking
- •

QUIC:

- 0-RTT Handshake
- Stream Multiplexing
- Frame Structure

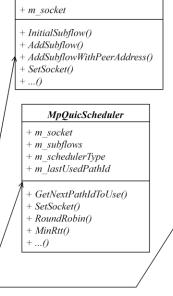
QuicL4Protocol CreateSocket() CreateUdpSocket() UdpBind() UdpSend() UdpRecv() ForwardUp() SendPacket() Is0RTTHandshakeAllowed() + Allow0RttHandshake() + AddPath() + ReDoUdpConnect() + ...0 OuicSocketBase + m enableMultipath: + m pathManager; + m scheduler: + m subflows: + m currentPathId: + m currentFromAddress: SendInitialHandshake() OnReceivedFrame() OnSendingAckFrame() AvailableWindow() SendDataPacket() ReceivedData() + SendAddAddress() + SendPathChallenge() + SubflowInsert() + CreatePathManager() + CreateScheduler() + OnReceivedAddAddressFrame() + OnReceivedPathChallengeFrame()

IMPLEMENTATION





- Path Identification
- Path Management
- Reliable Data Transmission
- Packet Scheduling
- Congestion Control



OuicCongestionOps

MpOuicPathManager

QuicHeader

OuicSubHeader

+ CreateAddAddress()

OuicSocketTxBuffer

+ m subflowSentList

NextSequence()

OnAckUpdate()

Retransmission()

+ AddSentList()

+ FindSentList()

TcpCongestionOps

IncreaseWindow()

PktsAcked()

GetSsThresh()

...()

+ ...0

GetNewSegment()

DetectLostPacket()

- CreateMpAck()

- SetPathId() - GetPathId()

+ m pathId

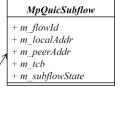
+ SetPathId()

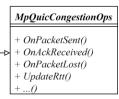
+ GetPathId()

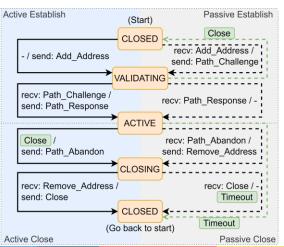
m pathId

m address

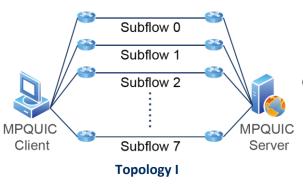
·...0

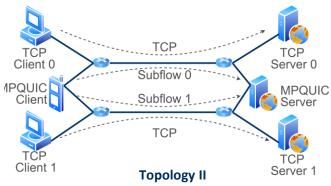


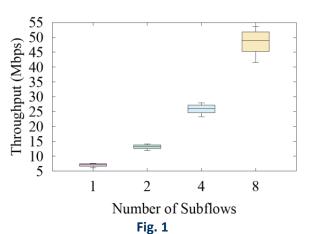


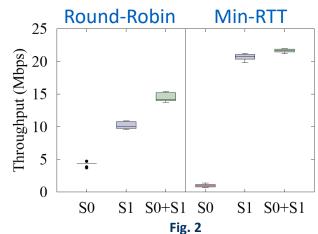


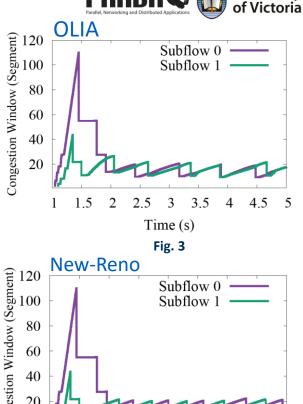
RESULTS



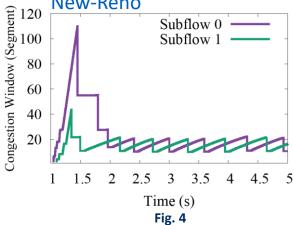








University



CONCLUSION & FUTURE IMPROVEMENT



- Presented an ns-3 implementation of MPQUIC
- Overcome the challenges of advertising multiple addresses, separating transmission paths, and extending the structure of algorithms
- Verify the correctness, scalability, flexibility, extensibility

- Implement the currently missing features and keep updating in the future
- Explore better scheduling and congestion control algorithms

THANK YOU FOR LISTENING! & ANY QUESTIONS?





