论文分享 - TECC: Towards Efficient QUIC Tunneling via Collaborative Transmission Control

Jiaxing Zhang[1,2], Furong Yang[1], Ting Liu[1,2], Qinghua Wu[2,3], Wu Zhao[1], Yuanbo Zhang[1], Wentao Chen[1], Yanmei Liu[1], Hongyu Guo[1], Yunfei Ma[1], Zhenyu Li[2,3]

[1] Alibaba Group, [2] University of Chinese Academy of Sciences, [3] Purple Mountain Laboratories, China

TL;DR

Background

QUIC-in-QUIC (mostly H3-in-H3) might be desirable:

- Application Gateway
- Load balancer
- VPN / Private Relay

Background

QUIC-in-QUIC (mostly H3-in-H3) might be desirable:

- Application Gateway
- Load balancer
- VPN / Private Relay

rfc9297 & rfc9298 “MASQUE”

Background

QUIC-in-QUIC (mostly H3-in-H3) might be desirable:

- Application Gateway
- Load balancer
- VPN / Private Relay

rfc9297 & rfc9298 “MAS**Q**UE” : Multiplexed Application Substrate over **QUIC** Encryption

Background

Datagrams travels inside the tunnel in two modes:

- Each connection = 1 stream
- Directly sent as unreliable QUIC datagram

Background a.k.a. The Problem

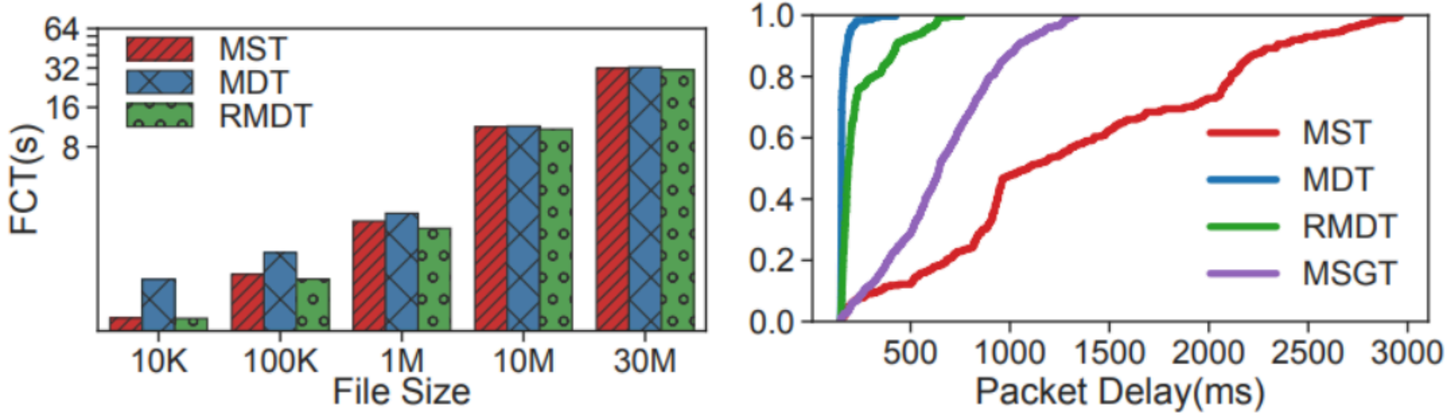
Datagrams travels inside the tunnel in two modes:

- Each connection = 1 stream
- Directly sent as unreliable QUIC datagram

Problem:

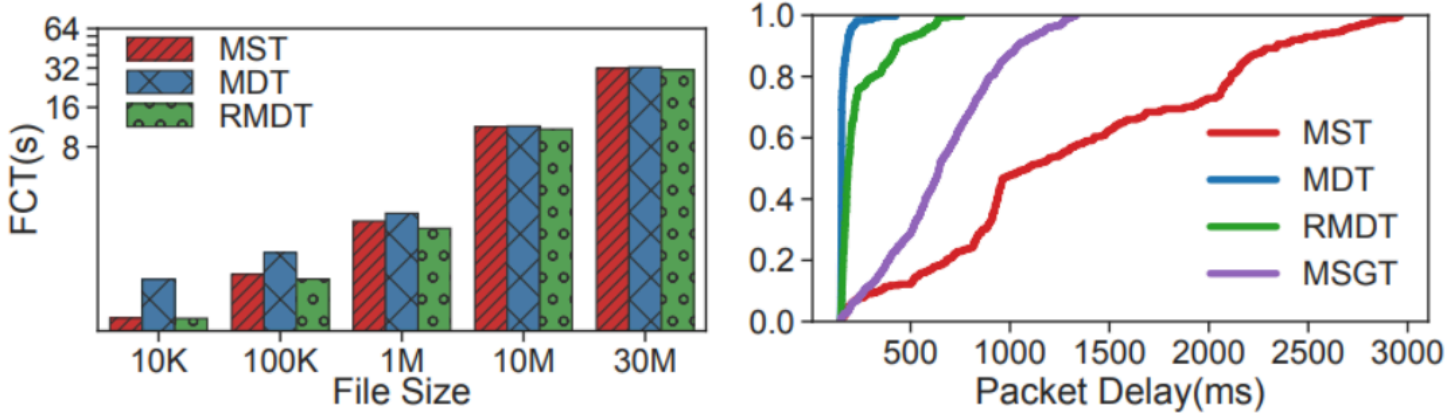
- **Retransmission**: can happen at both level, outer retransmission messes with inner congestion ctrl.
- **Congestion control**: inner and outer congestion ctrl may behave differently.

Retransmission or not?



(a) FCTs of E2E transfers (b) CDF of E2E packet delay

Retransmission or not?



(a) FCTs of E2E transfers (b) CDF of E2E packet delay

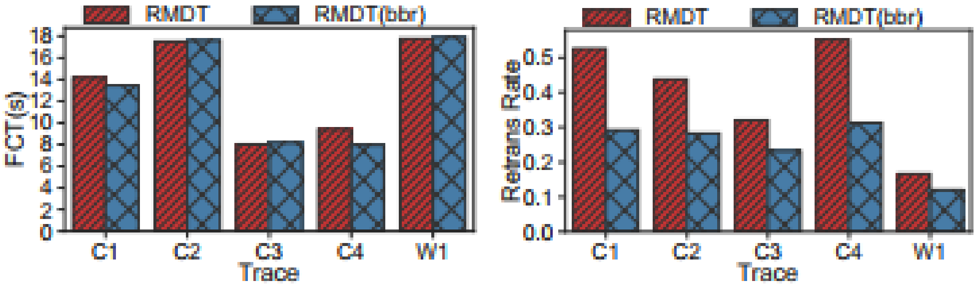
Datagram mode with retransmission works the best.

Nested Congestion Control

Less congestion $\Rightarrow$ Faster E2E time?

Nested Congestion Control

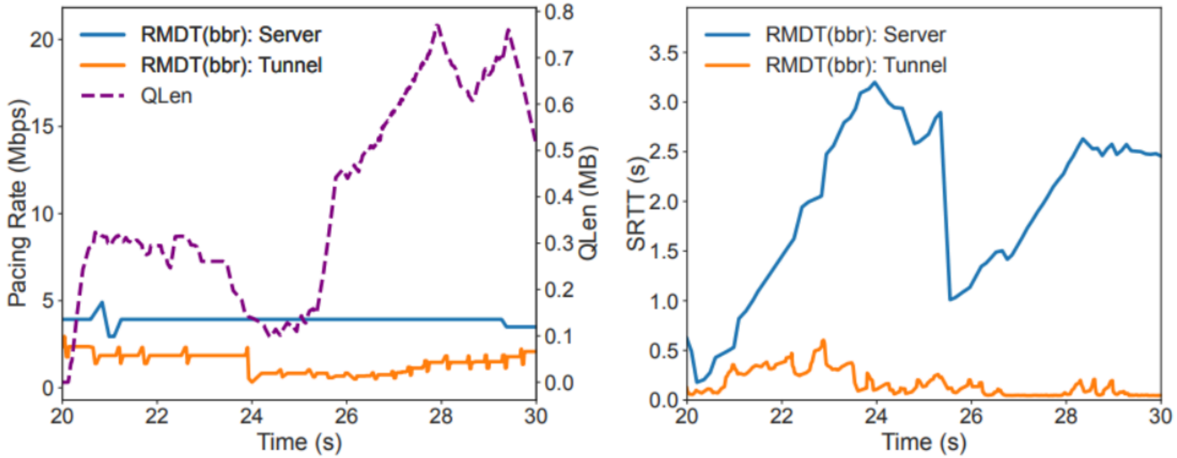
Less congestion $\nRightarrow$ Faster E2E time!



(a) FCT (b) Retransmission rate

Figure 3: The comparison between RMDT with and without CC

Nested Congestion Control



(a) Pacing rate (b) SRTT

Figure 4: The impact of nested CC: the pacing rate and SRTT of the tunnel and E2E connection are depicted.

A Dilemma

- No retransmission on tunnel ⇒ Long flow completion time(FCT) on short messages
-

A Dilemma

- No retransmission on tunnel $\Rightarrow$ Long flow completion time(FCT) on short messages
- Has retransmission on tunnel $\Rightarrow$ **Packet drops becomes invisible to E2E server!** Inner congestion control lags behind

TECC

- Keep RMDT (Datagram mode with retransmission)
- Enable collaborative congestion control

1. Tunnel send **bandwidth** and **queue** information to server
2. Server updates it pacing (sending rate)

TECC

- Keep RMDT (Datagram mode with retransmission)
- Enable collaborative congestion control

1. Tunnel send **bandwidth** and **queue** information to server
2. Server updates it pacing (sending rate)

$$Sr(t) = \mu \cdot Tr(t) + \lambda$$

TECC

- Keep RMDT (Datagram mode with retransmission)
- Enable collaborative congestion control

1. Tunnel send **bandwidth** and **queue** information to server
2. Server updates it pacing (sending rate)

$$Sr(t) = \mu \cdot Tr(t) + \lambda$$

Servers don’t use local data to do CC!

TECC

- Matching sending rates at server and tunnel egress
- Penalize large queue usage for fairness

Ideally: $Sr(t) = Tr(t) - q(t)/\theta$.

θ repersents the penality given by queue size.

TECC

- Matching sending rates at server and tunnel egress
- Penalize large queue usage for fairness

Ideally: $Sr(t) = Tr(t) - q(t)/\theta$.

θ repersents the penality given by queue size.

Also: Expect sudden drop in bandwidth

TECC

- Matching sending rates at server and tunnel egress
- Penalize large queue usage for fairness

Ideally: $Sr(t) = Tr(t) - q(t)/\theta$.

θ repersents the penality given by queue size.

Also: Expect sudden drop in bandwidth

Finally: Exponentially Weighted Moving Average to dampen the reports

TECC

Algorithm 1 Sender algorithm

Data: Tunnel Server Feedback: $Tr(t)$, $q(t)$, T_t

Result: Server: $Sr(t)$

function `UpdateSenderRate()`:

$e(t) \leftarrow \frac{q(t)+\delta r(t)Tr(t)}{\theta Tr(t)}$

$U(t) \leftarrow \max\{1-e(t), 1-max\_pf\}$

$U \leftarrow (1-ewma\_weight) \cdot U + ewma\_weight \cdot U(t)$

$Sr(t) = U \cdot Tr(t) + \frac{MSS}{T_s}$

return $Sr(t)$

Questions Unanswered by the Paper

1. Parameters θ, T_s depends on **real RTT**, is this approximated in runtime by smoothed detected RTT?
2. Is using server data to assist in CC able to produce some better result?
3. A lot of parameters are set based on experimentation. Can it be explained?

Anyway…

Anyway...

Evaluation!

Anyway…

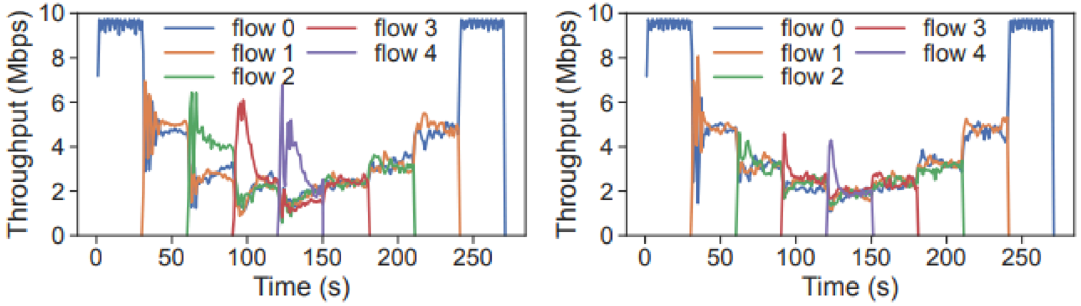
Evaluation!

- Extra good in mobile network! Performs very well under high (15%) packet loss rate.

Anyway…

Evaluation!

- Extra good in mobile network! Performs very well under high (15%) packet loss rate.
-



(a) TECC w/o AI (b) TECC with AI

Figure 15: Fairness among competing tunnel flows

In realworld…

	mean	p95	p99	p999
MST	2.7%	1.9%	6.0%	17.5%
TECC	3.9%	4.5%	13.3%	36.0%

Table 3: Improvements of MST and TECC