SCTP Tail Loss Recovery Enhancements

SCTP TLR

draft-nielsen-tsvwg-sctp-tlr-02.txt

K. Nielsen, A. Brunström, R. de Santis, M.Tuexen, R. Stewart

Tsvwg, IETF 94, Yokohama

SCTP TLR Background

- Goal of SCTP TLR:

  Reduce latency while scaling throughput

- Legacy SCTP Fast Recovery (FR) not able to adequately or timely repair losses in tails of flows
- Result is lenghty Loss Recovery by T3-timeout, detrimental to performance
- SCTP TLR is
  - New Systematic Approach to Timer Driven Loss Recovery that extends, embeds and replaces a number of prior special purpose approaches: TCP TLP [1], TCP TLPR [4], SCTP/TCP Early Retransmission RFC5827, SCTP/TCP RTO Restart, FACK, ..
  - Pt. supplements Existing RFC4960 Fast Recovery
  - Evolved from TCP TLP[1]. Might evolve aside TCP RACK and QUIC
  - Ver 01 presented at tsvwg IETF90, IETF91

SCTP TLR - Simplified View

- Timer Based Loss Detection and FR

  TSN declared lost when

  SACK of higher TSN has arrived &

  time >~ 1.5RTT since TSN sent

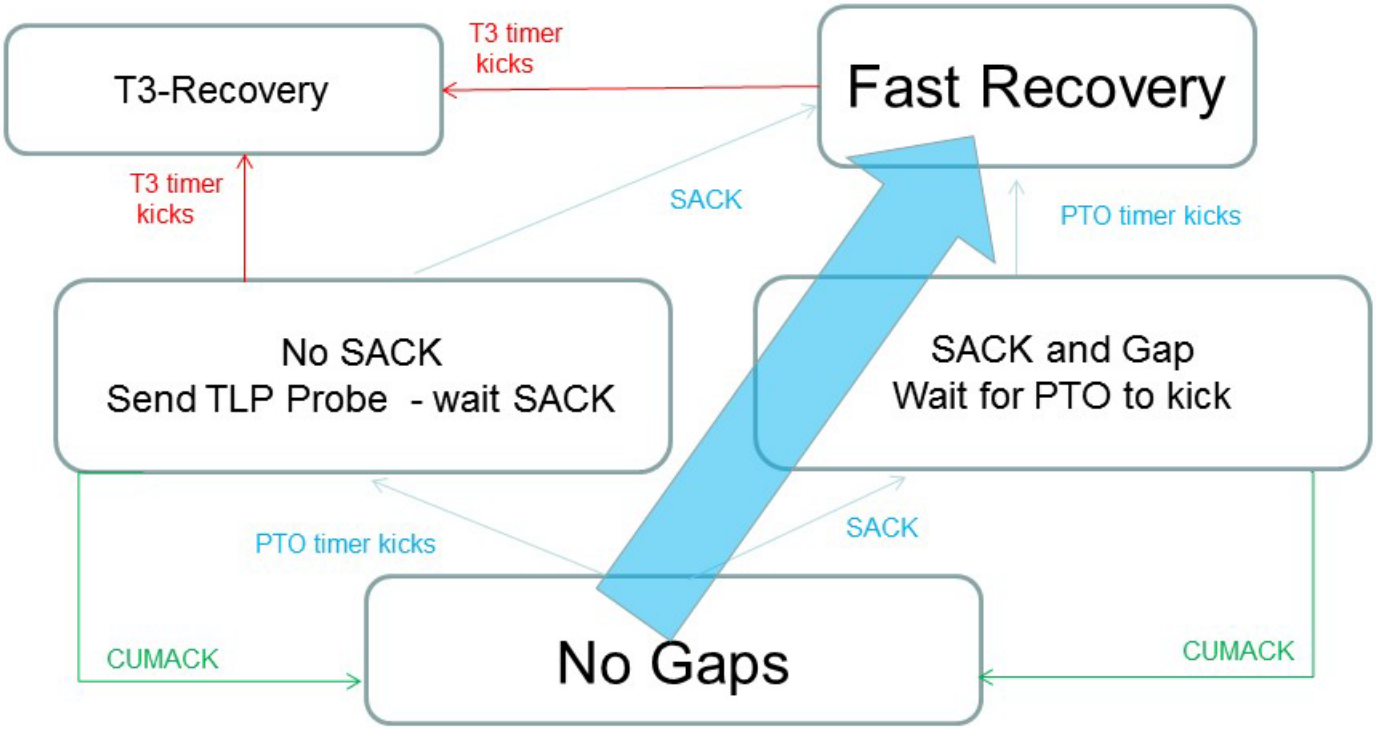
- Timer Restart Principles

  Timer reset as Timer-Time_TSN_latest_sent

- Elimination of ”unnecessary” Delay_ack

  SCTP I-bit eliminates unnecessary delays

PTO Timer Driven Loss Detection



- SACK of higher TSN & PTO time elapsed → Enter FR
- PTO Timer driven Tail Loss Probing (insp. TCP TLP [1])
  - TLProbe Packet (TLPP) sent if no SACK within PTO
  - Loss Demasking as from TCP TLP [1] applied to SCTP

Tail Loss Improvements of FR & Timer Driven Loss Detection within FR

- TCP RFC6675 Tail Loss improved SCTP FR
  - RFC6675 Last Resort features added to SCTP FR
    - Nextseq 3): If SACK of higher TSN received → allow to FR ""FR-TLR""
    - Nextseq 4): Rescue of FR Tail → FR of Exit Point for SACK to drive FR "FR-TLP"
  - RFC6675 mis-indication from content of SACK, not number of SACKs

- Timer Driven Loss Detection during SCTP FR
  - TSNs classified as lost and expelled from flight size during Fast Recovery when: T_latest(TSN) > PTO & SACK of higher TSN
    - NB: RFC6675 Nextseg 3) counts twice in flight size and sent after new data only. SCTP CC is standard RFC5681/RFC6675

SCTP TLR Timers

- PTO timers

  PTO1= 1.5 SRTT + MAX(RTTVAR, Delay_ack)

  PTO2= 1.5 SRTT + RTTVAR

- PTO timer restart: PTO reset as: PTO-Time_TSN_latest_sent (except when PTO ”not trusted”)

  [OPEN]

  - SACK frequency dependency, No special re-ordering awareness, Freshness (fresh RTT feed)

- T3 timer restart: T3 timer during FR and T3-Recovery set as:

  RTO-Time_TSN_latest_sent

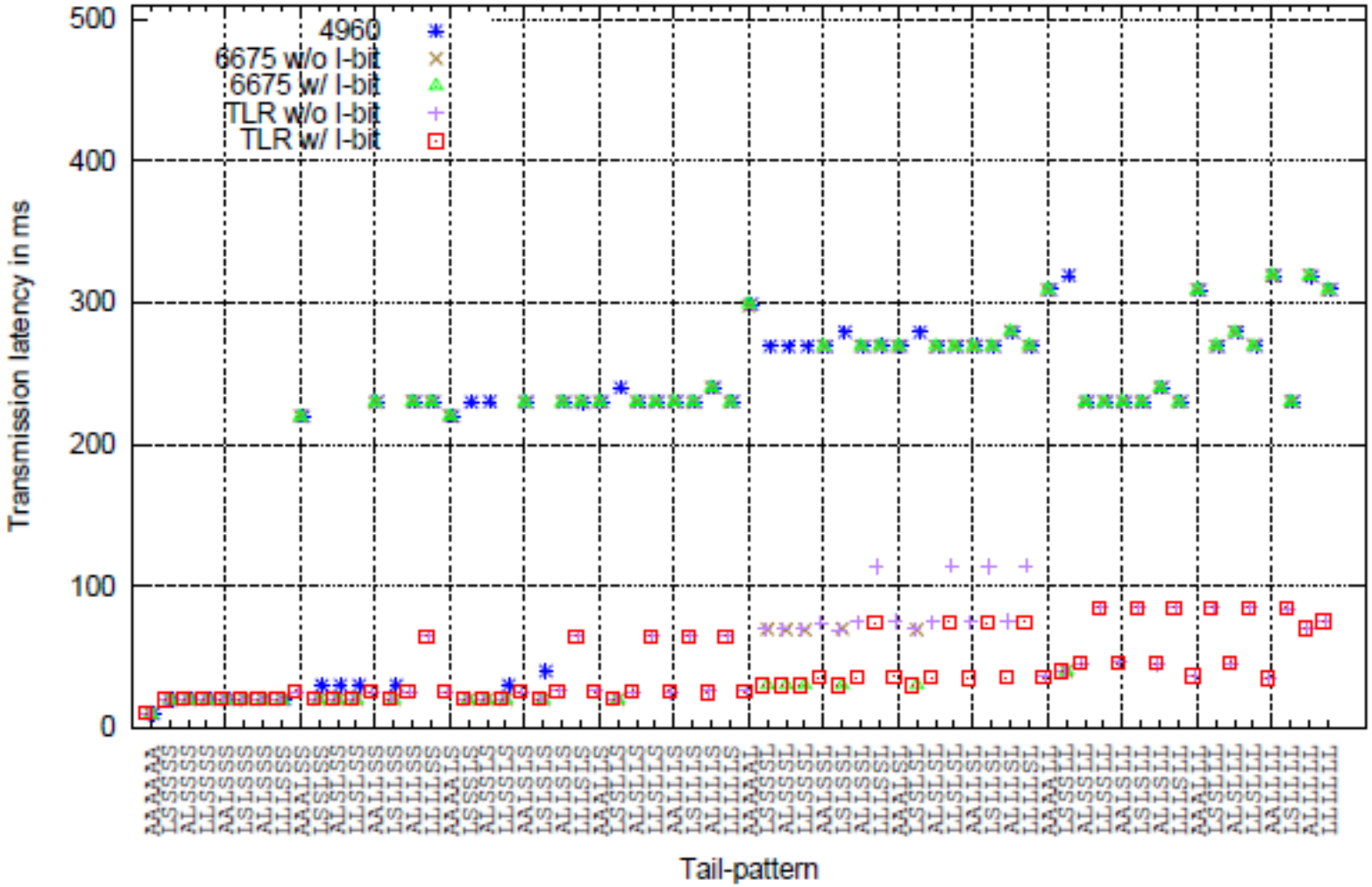
- Elimination of Delay_ack latency
  - TLPP and RFC6675 Rescue sent with I-bit = Immediate SACK-bit
  - PTO2 timer used when I-bit set and peer complies with RFC7053

Not added in tests (yet) (Not strictly speaking TLR related):

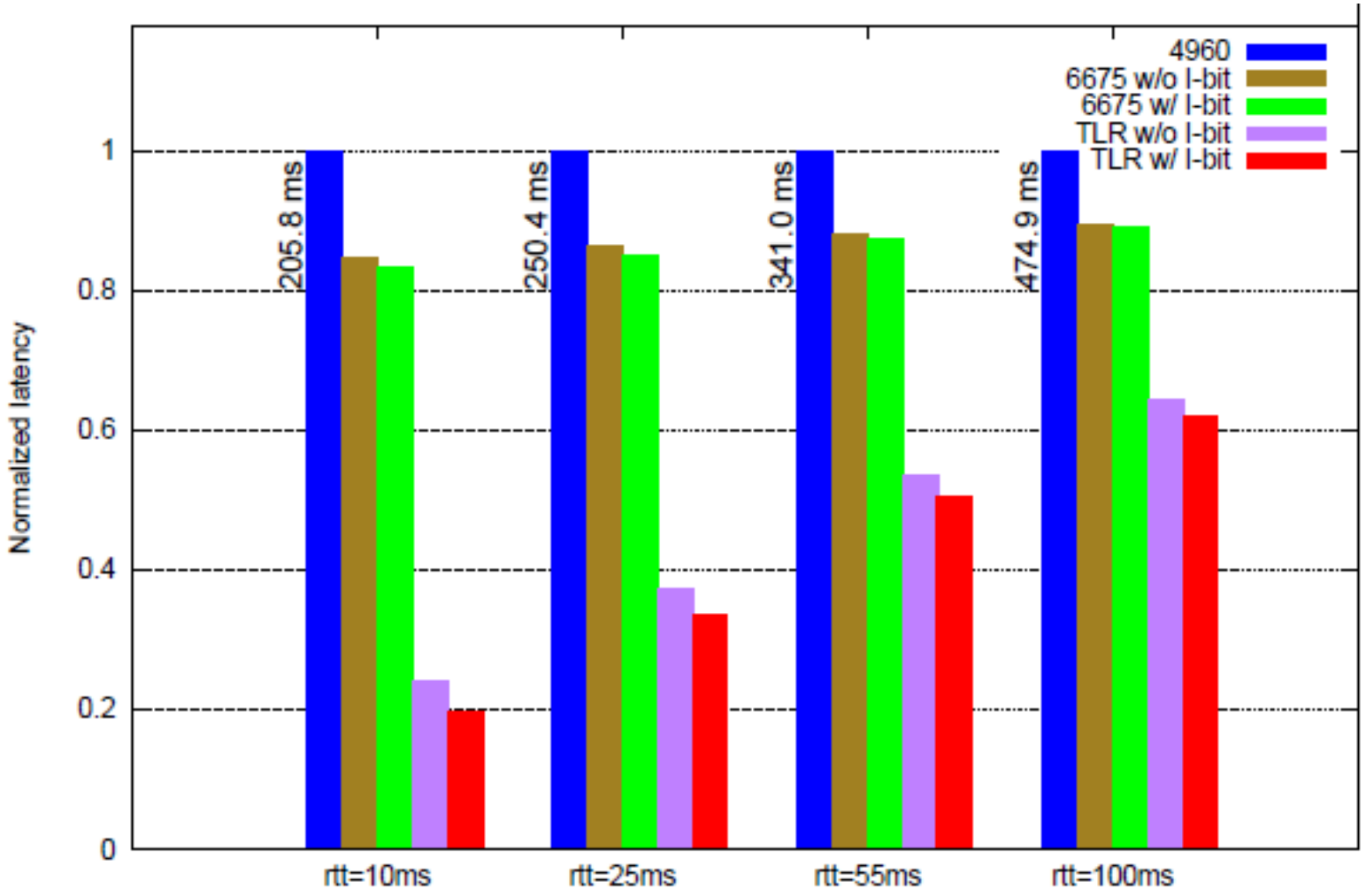
- I-bit on first RTX packet after T3-timeout ✓
- I-bit on last packet (RTX only?) prior to CWND limitation, other. [Open]

Two Way Completion Time
Tails of 6 Packets

RTO_MIN=200 Msecs, Delay_Ack=40msecs, RTT=10 msecs



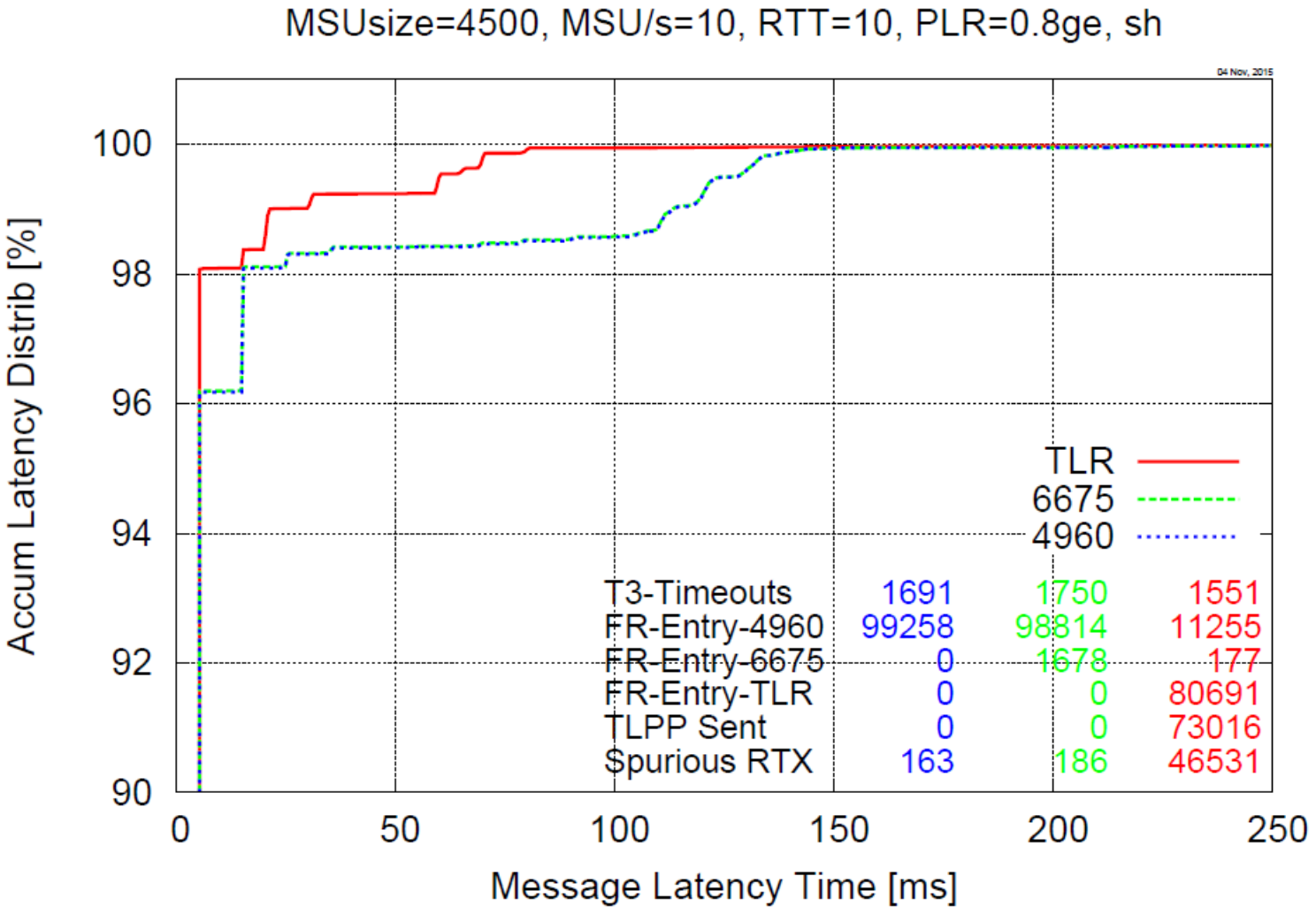
Two Way Completion Time
Tails of 6 Packets



RTO_MIN=200 Msecs, Delay_Ack=40msecs

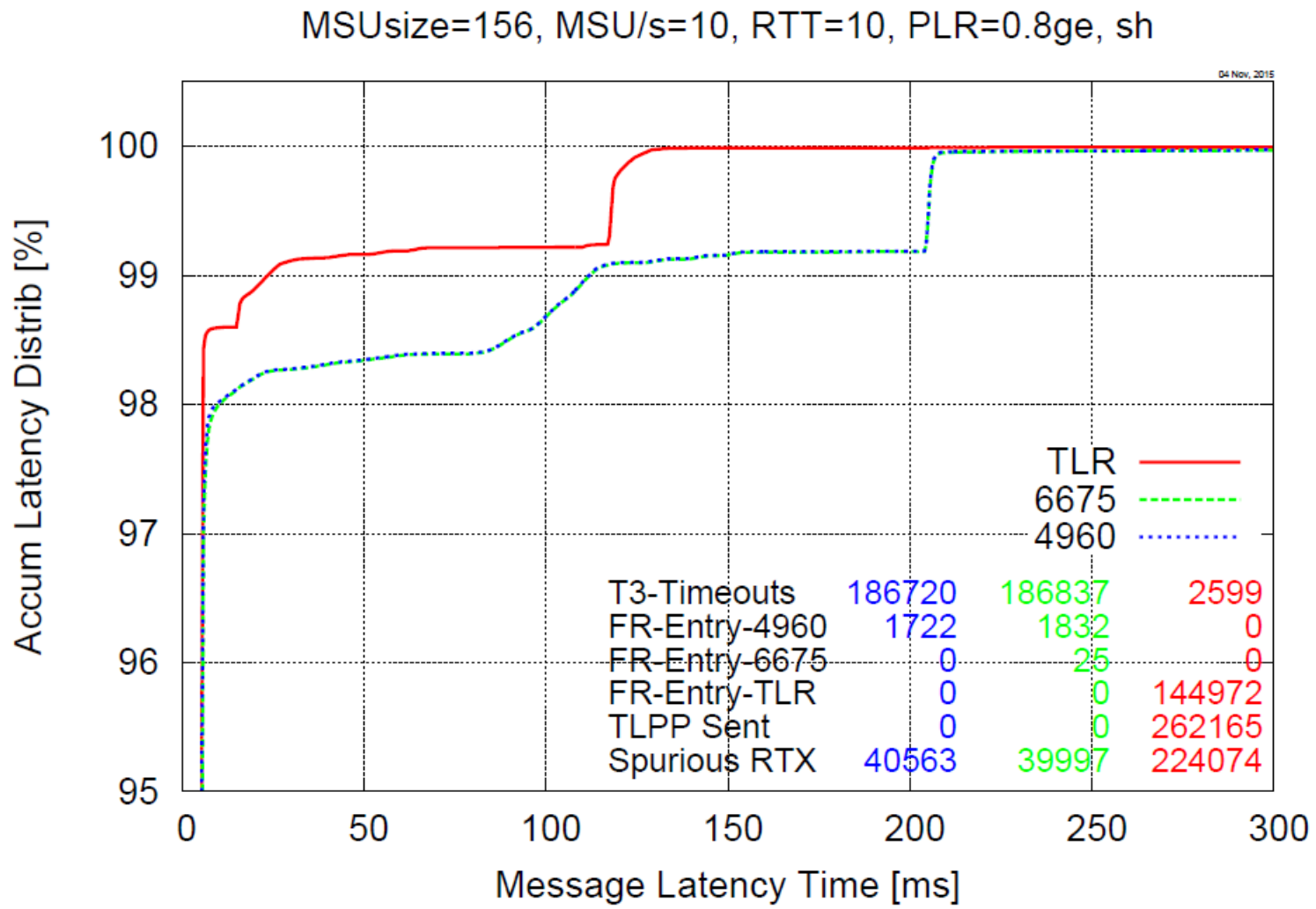
Results SCTP Single Homing

- Diameter Signalling Traffic Profile

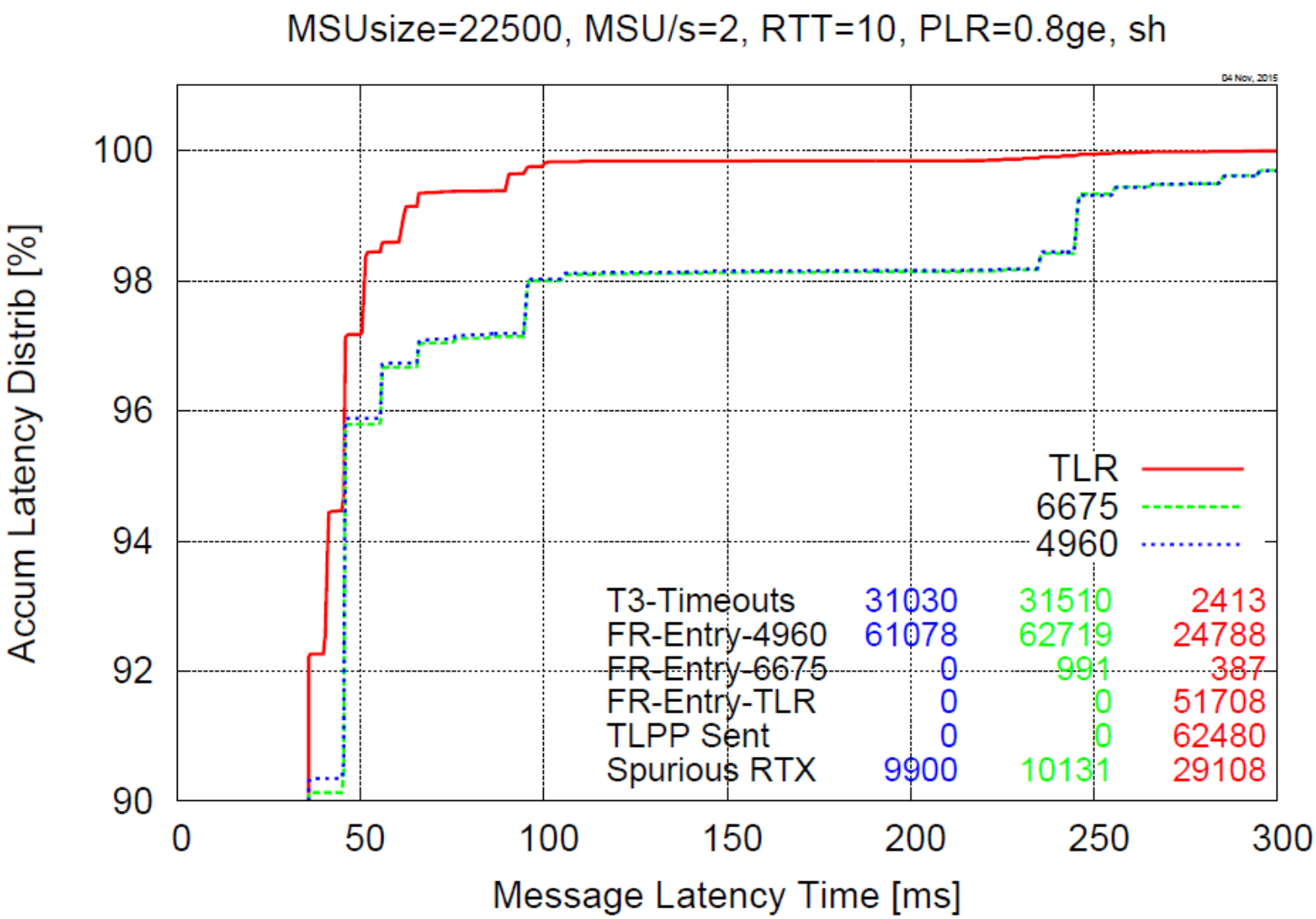


Results SCTP Single Homing

- SGaAP Signalling Traffic Profile



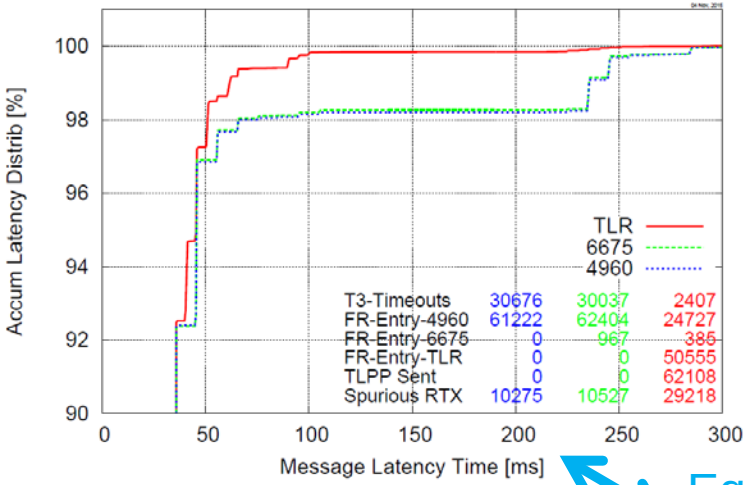
Results SCTP Single Homing

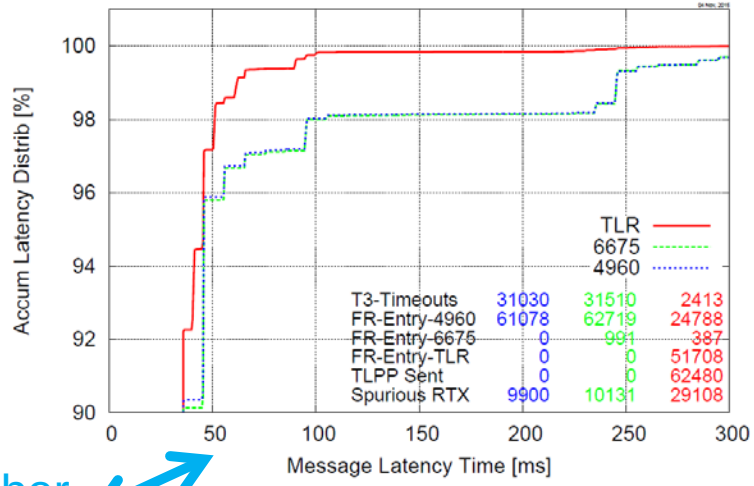


SCTP TLR for Multi Homing

- Is SCTP TLR always good for SCTP MH ?
  - Better to get a T3 and go use alternate path w/ slow start from IW=4 ?
- Results for signaling profiles:
  - With equal path characteristic SCTP TLR better or equal to RFC4960
  - FR CC on existing path and slow start CC on alternative path equally adequate/cannot outweigh T3-timeout latency
  - Slow Start CC on alternate path amends T3-timeout latency
    - But here RFC4960 observed to **result in more spuriously retransmitted data than SCTP TLR**
- Results likely to change with differences in the path characteristics/traffic. More testing to be done…

MSUsize 22500, MSU/s=2, RTT=10, PLR=0.8



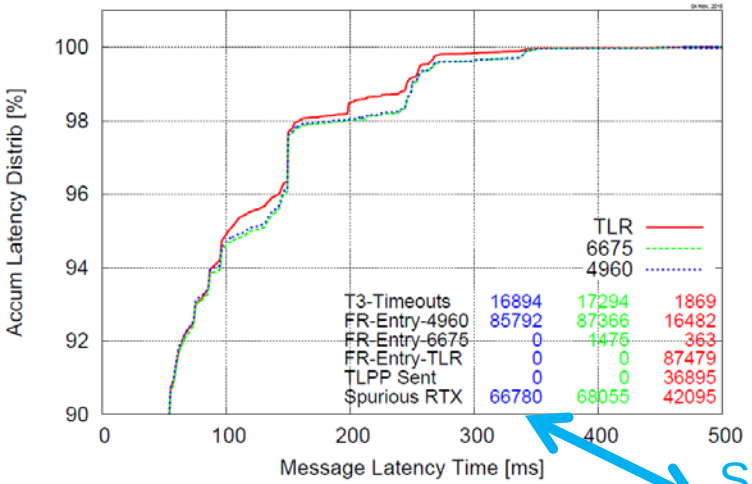


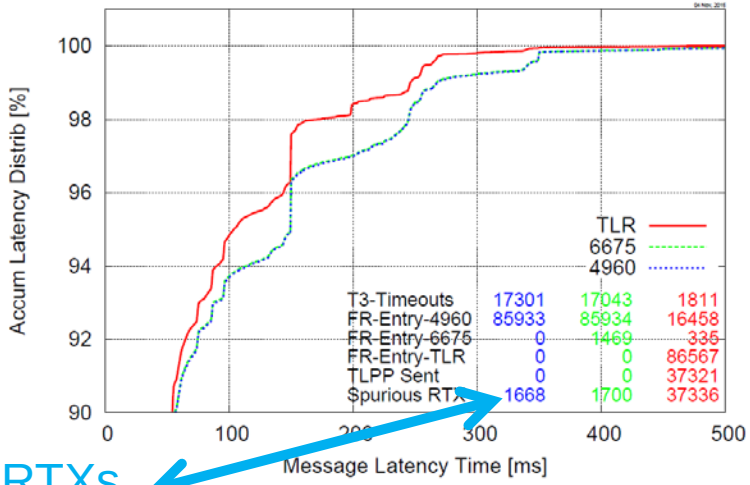
SCTP MH

Equiv number of spurious RTX

SCTP SH

MSUsize 4500, MSU/s=10, RTT=100, PLR=0.8





SCTP MH

Spurious RTXs Increased by factor of 40 !

SCTP SH

SCTP TLR and new SACK

- Proposed Unambuigous SACK for SCTP
  - See Appendix of draft-nielsen-tsvwg-sctp-tlr-02.txt
  - Usage for SCTP TLR in bulk of draft-nielsen-tsvwg-sctp-tlr-02.txt
- Enable SCTP sender to distinguish SACK of original transmission and retransmission:
  - Accurate - RTT measurements, - Loss Detection, - CWND Management, - MH management
- NOT IMPLEMENTED (YET)
  - Format to be evaluated.
  - Also ?
    - Potentially add DELAY_ACK setting, SACK frequency (perhaps), delay of highest SACK'ed TSN (compare/from QUIC)

Status and NEXT Steps

- Implementation of new SCTP TLR improvements in Ericsson SCTP SW in progress
  - “RFC6675” parts are running in deployment
- Specification and design is work in progress. Present version:
  - `Draft-nielsen-tsvwg-sctp-tlr-02.txt`
  - Includes MH&SH operation
- More consolidation needed before asking for adoption for standardisation by IETF

QUESTIONS

References

[1] Dukkipati et al., Tail Loss Probe (TLP): An Algorithm for Fast Recovery of Tail Losses, Expired work. http://tools.ietf.org/html/draft-dukkipati-tcpm-tcp-loss-probe-01

[2] Dukkipati et al, "Proportional Rate Reduction for TCP", Proceedings of the 11th ACM SIGCOMM Conference on Internet Measurement 2011, Berlin, Germany, November 2011.

[3] Hurtig et al, TCP and SCTP RTO Restart, draft-ietf-tcpm-rtorestart-04, Work In Progress

[4] M. Rajiullah et al., , "An Evaluation of Tail LossRecovery Mechanisms for TCP", ACM SIGCOMM Computer Communication Review 45,1, 1 2015.