On the Validity of Credit-Based Shaper Delay Guarantees in Decentralized Reservation Protocols

Lisa Maile

Rechnernetze und Kommunikationssysteme
Friedrich-Alexander Universität,
Erlangen-Nürnberg, Germany

lisa.maile@fau.de

Introduction

Goal

Problem:
Worst case latency formulas for CBS in TSN standards. But they do not cover the actual worst case.

Result:
Real-time streams may miss their “promised” deadline.

Solution:
Adapting the decentral admission control scheme to offer safe guarantees.

Based on conference publication (copyright by ACM):
L. Maile, D. Voitlein, A. Grigorjew, K.-S. J. Hielscher, and R. German, *“On the Validity of Credit-Based Shaper Delay Guarantees in Decentralized Reservation Protocols,”* in *Proceedings of the 31st International Conference on Real-Time Networks and Systems (RTNS ’23).* Dortmund, Germany. Association for Computing Machinery (New York, NY, USA), Jun. 2023, pp. 108–118. doi: 10.1145/3575757.3593644.

Accepted version accessible via http://arxiv.org/abs/2311.04513

Decentralized Reservation Protocols in TSN

Decentralized Reservation Protocols

Protocols for Decentralized Resource Reservations

1997
Resource Reservation Protocol (RSVP)

commonly using IntServ
and per-flow shaping

2010
Stream Reservation Protocol (SRP)

for Credit-Based Shaper
with per-class shaping
IEEE 802.1Qat

2018
Resource Allocation Protocol (RAP)

support planned for various schedulers
including Credit-Based Shaper
IEEE P802.1Qdd (draft)

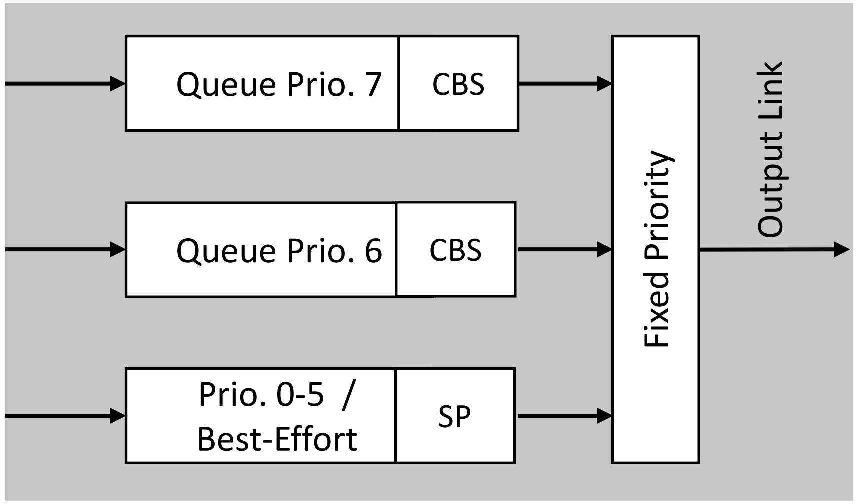
Credit-Based Shaper

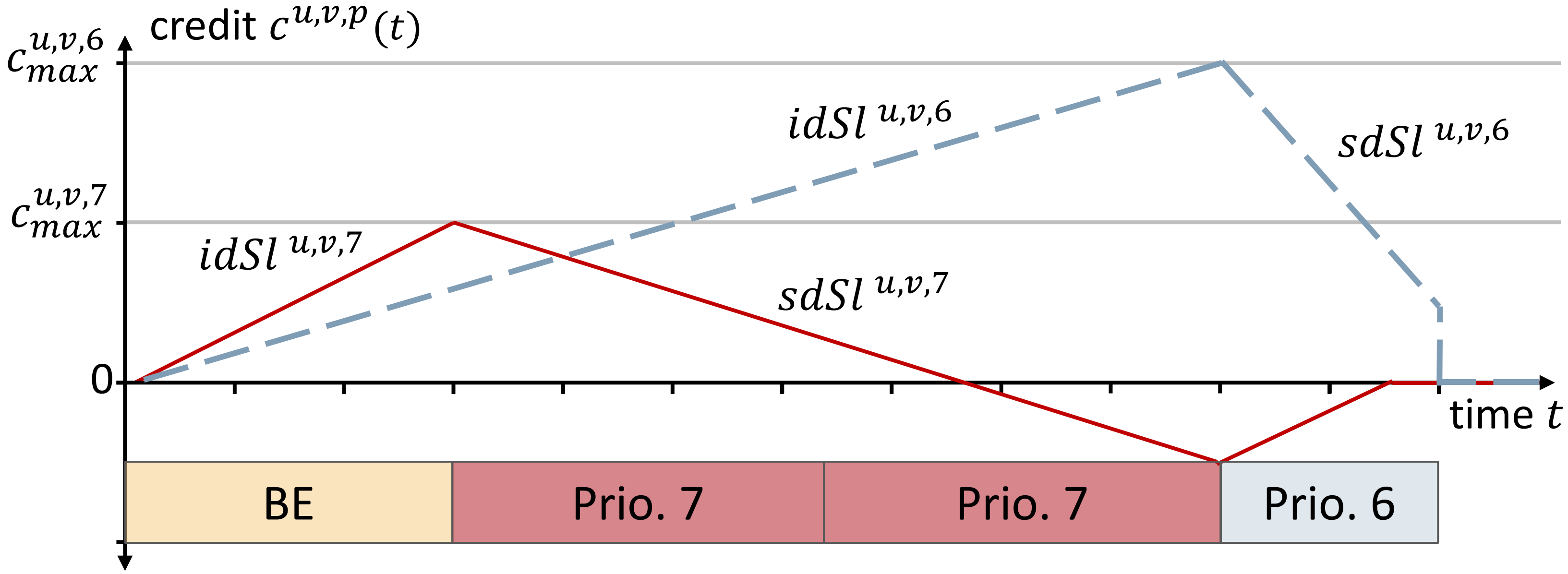
Recap

Credit-Based Shaper (CBS)

- per-queue shaping
- $idSl$ is the reserved bandwidth for each queue

Example Output Port (node u to v)





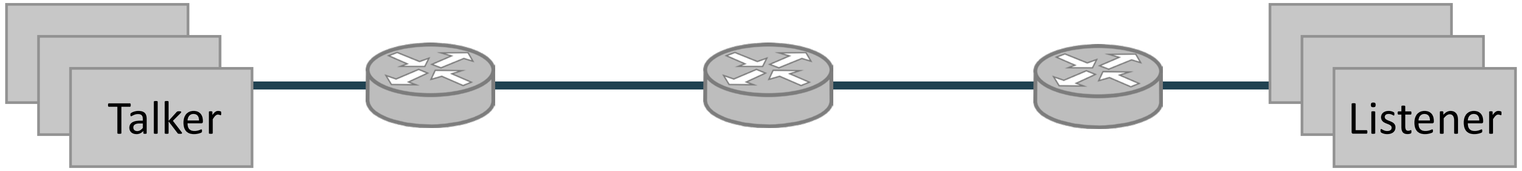
Decentralized Reservation Protocols

General Procedure

Decentralized Admission Control

1) Talker advertises the availability of data → including TSpec and delay-requirement

Stream / Flow ID
TSpec: Frame Size & CMI
Priority
E2E delay requirement
Acc. Delay (over path)



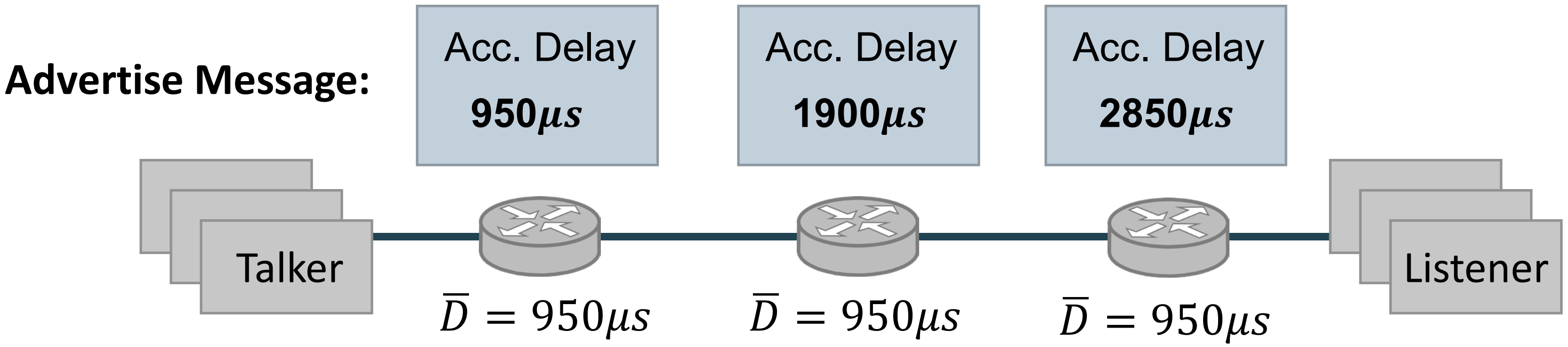
Decentralized Reservation Protocols

General Procedure

Decentralized Admission Control

1) Talker advertises the availability of data → including TSpec and delay-requirement

2) Advertisements are distributed through the network → per-hop delay bounds $\overline{D}$ accumulated in protocol field, with $\overline{D}$ from standard formulas



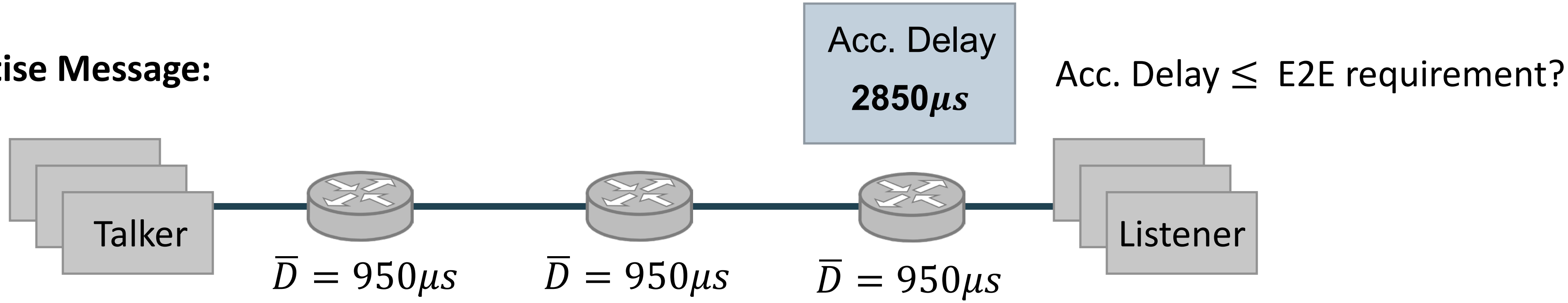
Decentralized Reservation Protocols

General Procedure

Decentralized Admission Control

1) Talker advertises the availability of data → including TSpec and delay-requirement
2) Advertisements are distributed through the network → per-hop delay bounds $\overline{D}$ accumulated in protocol field, with $\overline{D}$ from standard formulas
3) Listeners receive the advertisements, if interested in the stream: respond with subscription → check: accumulated delay $\leq$ end-to-end delay-requirement

Advertise Message:

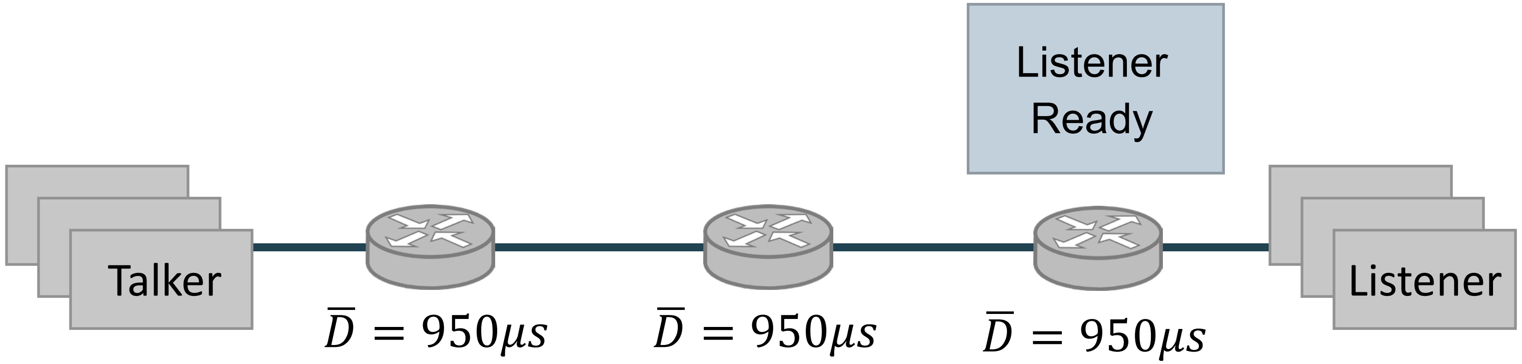


Decentralized Reservation Protocols

General Procedure

Decentralized Admission Control

1) Talker advertises the availability of data → including TSpec and delay-requirement
2) Advertisements are distributed through the network → per-hop delay bounds $\overline{D}$ accumulated in protocol field, with $\overline{D}$ from standard formulas
3) Listeners receive the advertisements, if interested in the stream: respond with subscription → check: accumulated delay ≤ end-to-end delay-requirement
4) Bridges check required bandwidth and reserve stream
5) Transmission of data



Latency Calculation in the Standards

Existing Formulas

Existing worst case latency calculations in the TSN standards:

“Rule of Thumb” in IEEE Std 802.1BA-2021

"2 ms [...] for SR Class A can be met for 7 hops of 100 Mbit/s Ethernet
if the maximum frame size on the LAN is 1522 octets“

Counterproof by Christian Boiger (IEEE 802 Plenary Meeting, 09 November 2010)

Latency Calculation in the Standards

Existing Formulas

Existing worst case latency calculations in the TSN standards:

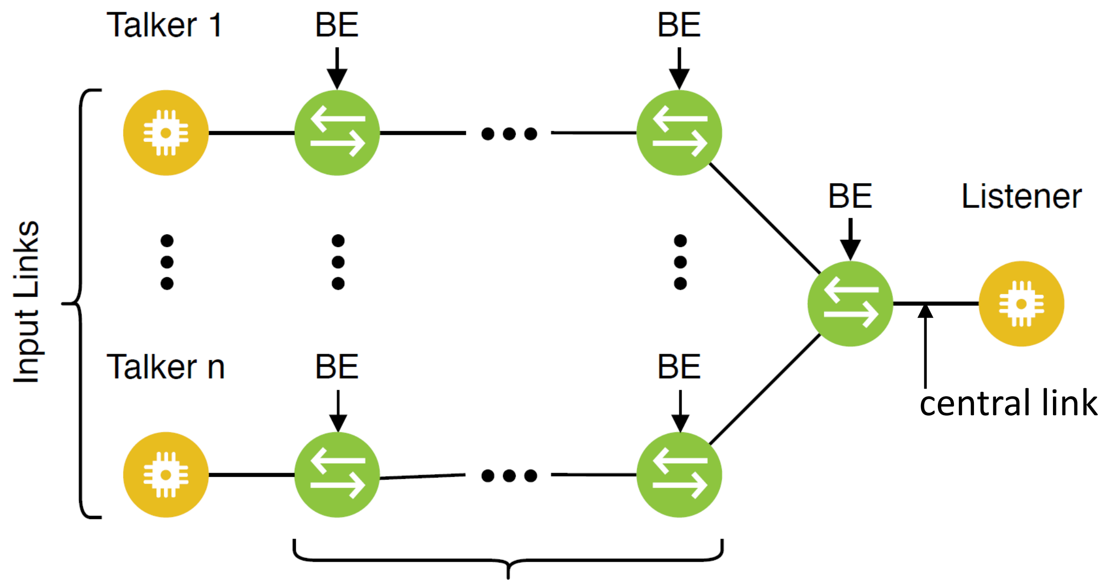
IEEE 802.1BA-2021 Section 6.6	$$\overline{D} = t_{proc} + \underbrace{t_{L_{max}}}_{\text{other pr.}} + \underbrace{\left(\frac{idSl}{C} \cdot CMI - t_{L_{FoI}}\right) \cdot \frac{C}{idSl}}_{\text{same priority}} + t_{L_{FoI}-IPG}$$
IEEE 802.1Q-2022 Section L.3.1	$$\overline{D} = t_{inQueue} + t_{int} + t_L + t_{prop} + t_{sf}$$ includes $$t_{queue} = \begin{cases} L_{max}/C & \text{for prio. 7} \\ (L_{max} + L^{(7)})/(C - idSl^{(7)}) & \text{for prio. 6} \end{cases}$$
Reference [B3] in IEEE 802.1Q-2022, see p. 1529	$$\overline{D} = \left(\underbrace{L_{max}}_{\text{other pr.}} + \underbrace{2 \cdot (R_{max} - L_{FoI}) - \left\lceil \frac{R_{max} - L_{FoI}}{N} \right\rceil}_{\text{same priority}} + L_{FoI}\right) \cdot t_{oct}$$ $$R_{max} = \left\lceil \frac{CMI}{t_{oct}} \cdot \frac{idSl}{C} \right\rceil, \; N = \min\left(\lvert \mathcal{L}^{-} \rvert, \left\lfloor \frac{R_{max} - L_{FoI}}{L_{min}} \right\rfloor\right)$$

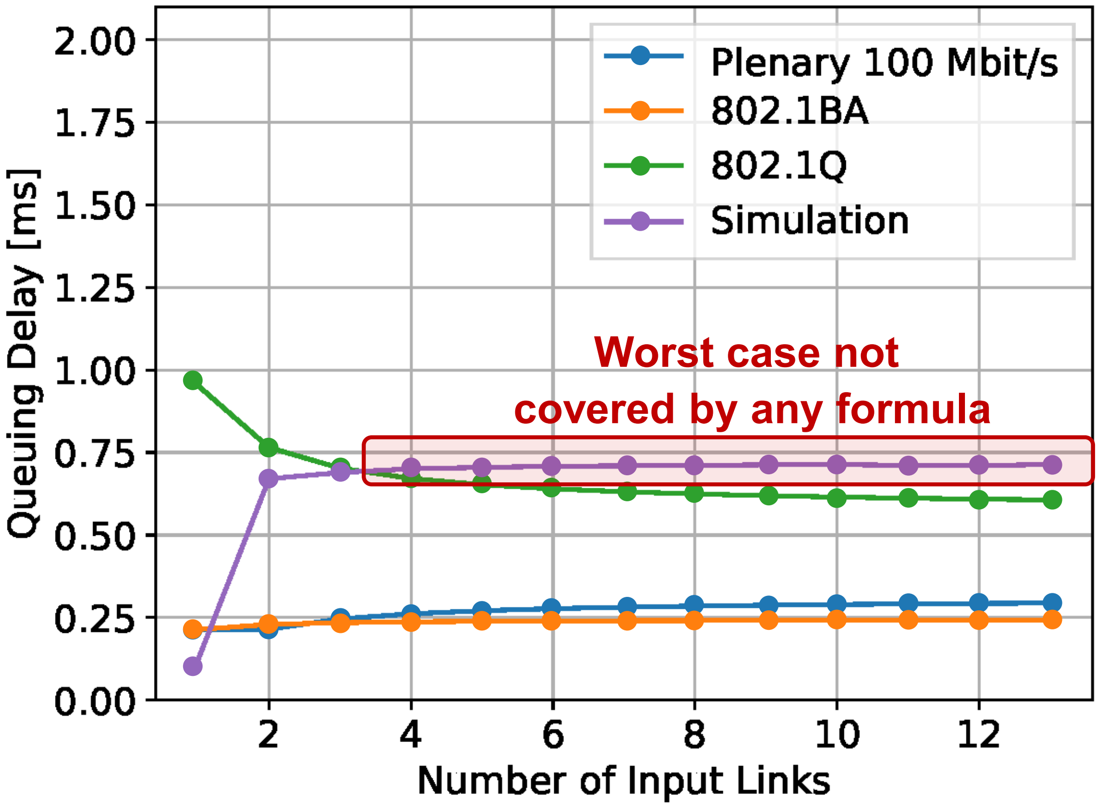
Latency Calculation in the Standards

Counterexamples by Simulation

Simulation Results

- link capacity $100 Mbit/s$, $idSl = 0.75$, priority 7, Talker CMI $= 125\mu s$
- constant central link utilization $75 Mbit/s$
- packet size adapted to number of Input Links (max. 13 due to min. Ethernet size)
- measured max. queueing delay at output port before central link





Unbounded Local Latencies

Unbounded Latency

Proof Overview

Problem:

Decentral architecture → locally available information for worst case delay calculation.

Given	Wanted
– **data rate** of all connected links – **idleSlopes** of all queues – **TSpec** of all reserved streams (stream ID, priority, sending interval, and data) – before each new reservation: **check available data rate and idleSlope**	→ **maximum local latency** may not change afterwards

Theorem:

With the given information for the decentralized TSN reservation protocols the maximum local latencies in CBS are unbounded.

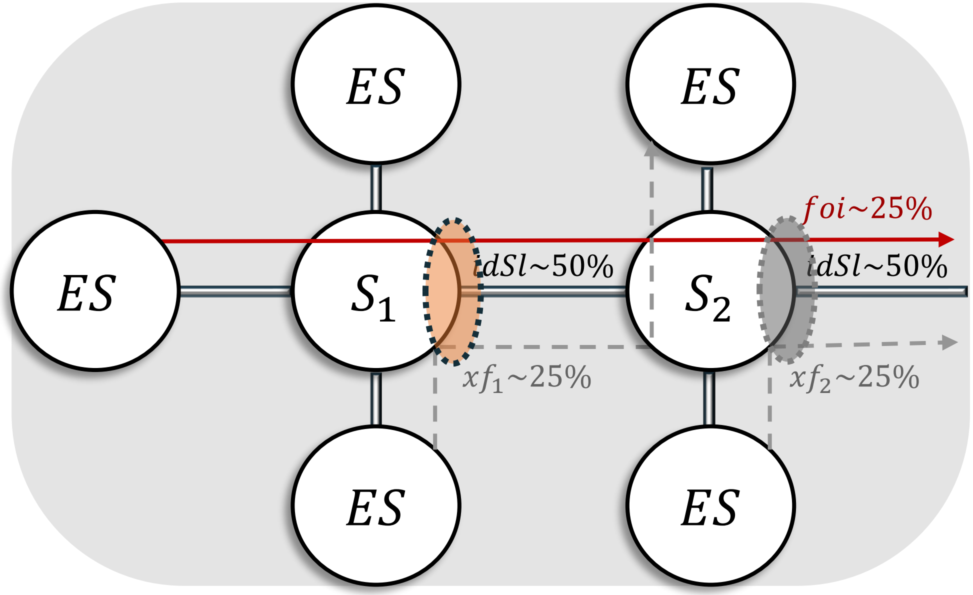
Proof:

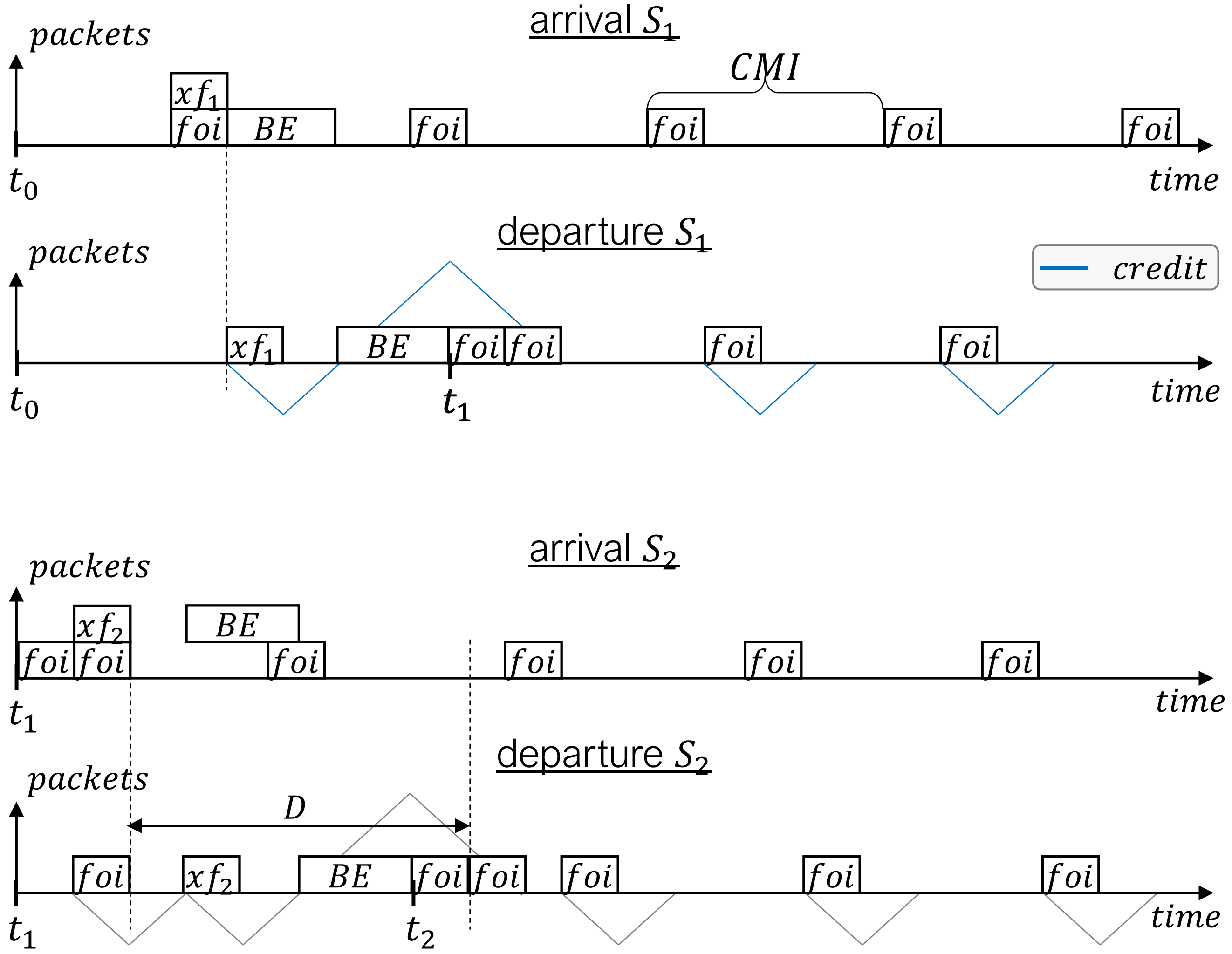
Details in publication, idea illustrated in the following → based on recursion.

Unbounded Latency

Illustration of Proof with Recursion

Recursion Level 0:

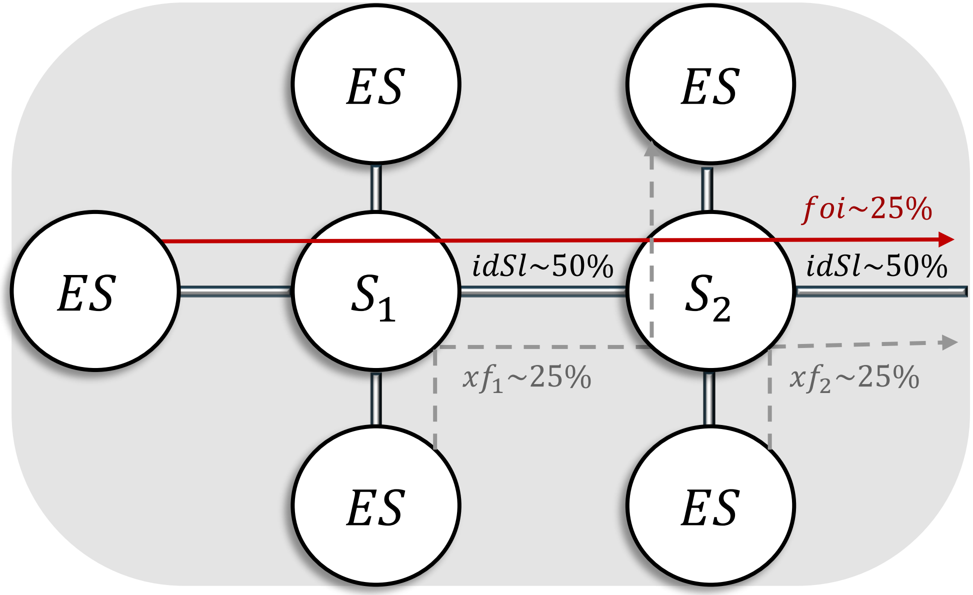




Unbounded Latency

Illustration of Proof with Recursion

Recursion Level 0:



Principle of Recursion:

recursion level 0
foi: path with **two** hops
cross-flows (xf_i): path with **two** hops

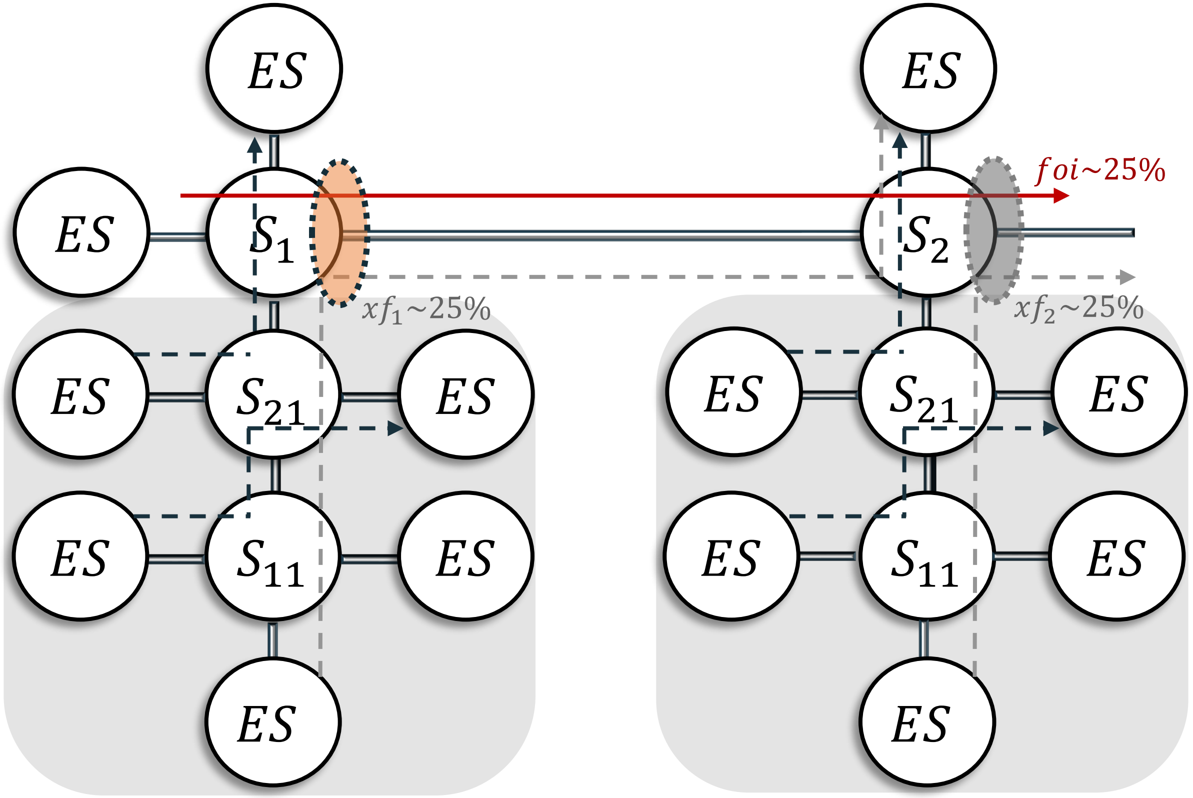
For each recursion level i:
crossflows of level i = foi in level i-1

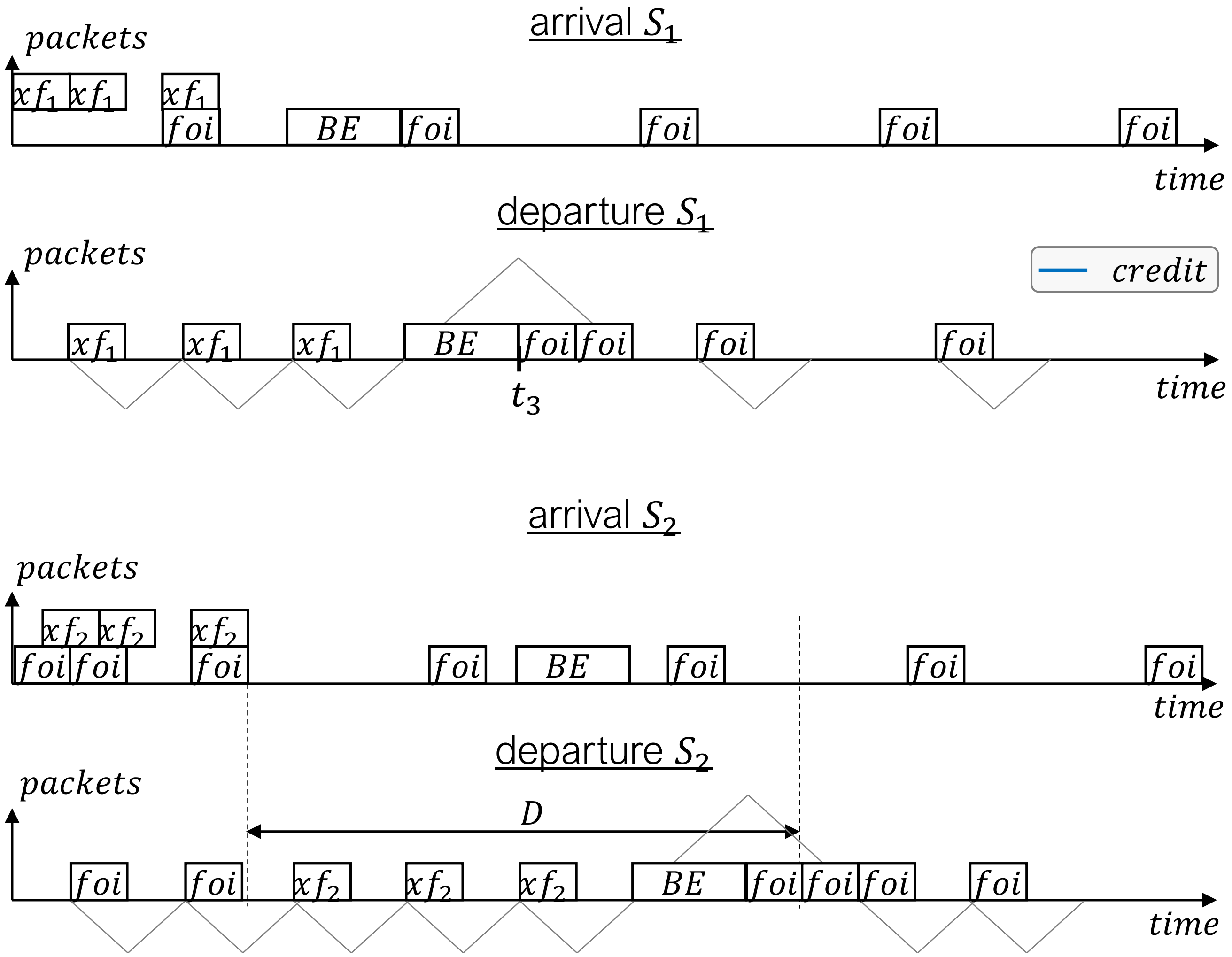
recursion level i → ∞
foi: path with **two** hops
cross-flows (xf_i): path with **four** hops

Unbounded Latency

Illustration of Proof with Recursion

Recursion Level 1:





Unbounded Latency

Summary of Proof

Summary of Proof:

- local delay D increases with each recursion level i → ∞ to infinity
- **Result:** even with **limited $idSl$, fixed CMI,** and **limited path length**, the worst case delay of CBS can increase **infinitely** → depends on the cross-flows (more traffic added to the network, but not locally at the bridge).
- **Note 1:** The actual delay can be safely calculated (e.g., with Network Calculus, eligible intervals, etc.) with global network knowledge. But no bound with only local information possible.
- **Note 2:** The proof is even simpler, if the CMI values are not fixed.

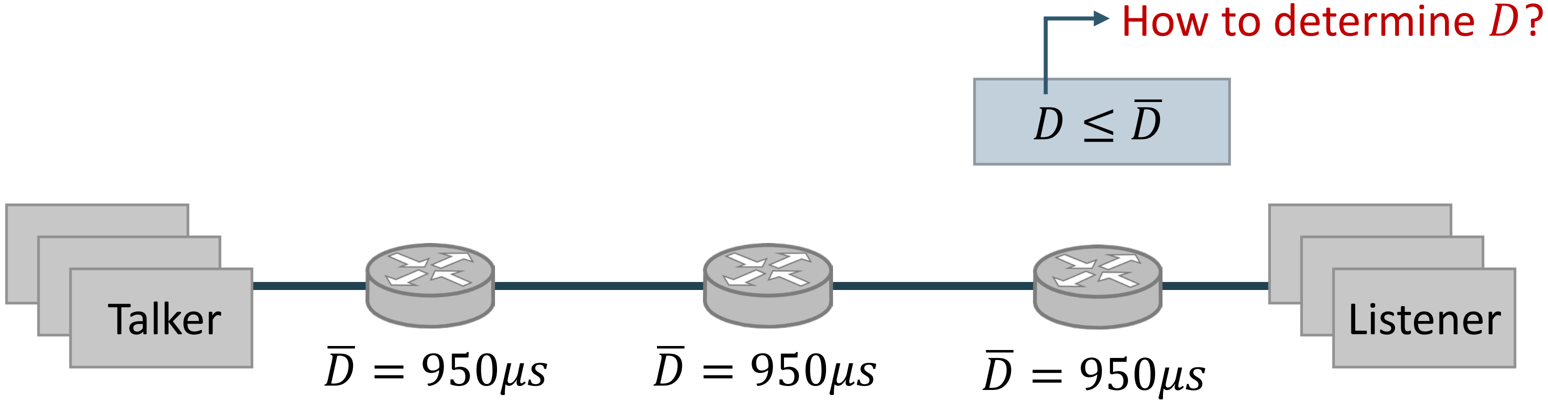
Solution: Reliable Reservation Protocol

Reliable Reservation Protocol

New Solution / Overview

Decentralized Admission Control

1) Talker advertises the availability of data → including TSpec and delay-requirement
2) Advertisements are distributed through the network → per-hop delay bounds $\bar{D}$ accumulated in protocol field, with $\bar{D}$ ~~from standard formulas~~ as configurable input
3) Listeners receive the advertisements, if interested in the stream: respond with subscription → check: accumulated latency $\leq$ end-to-end dela-requirement
4) Bridges check required bandwidth and reserve stream → check: actual/current worst case delay $D \leq$ delay bound $\bar{D}$
5) Transmission of data

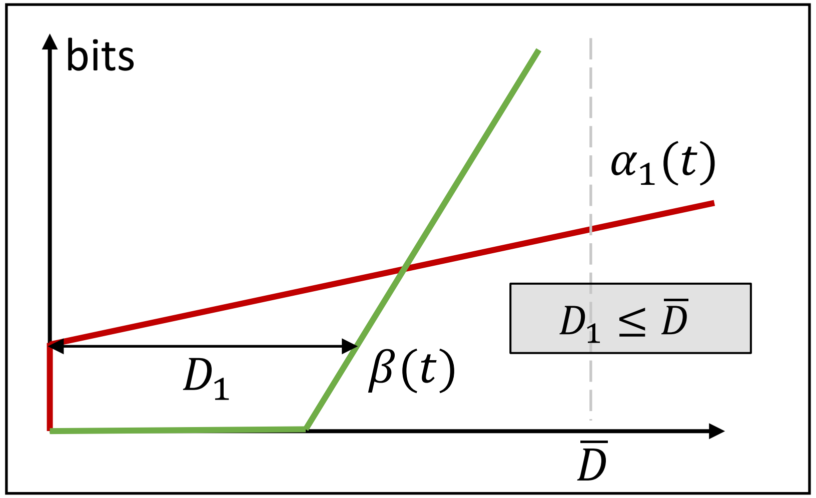


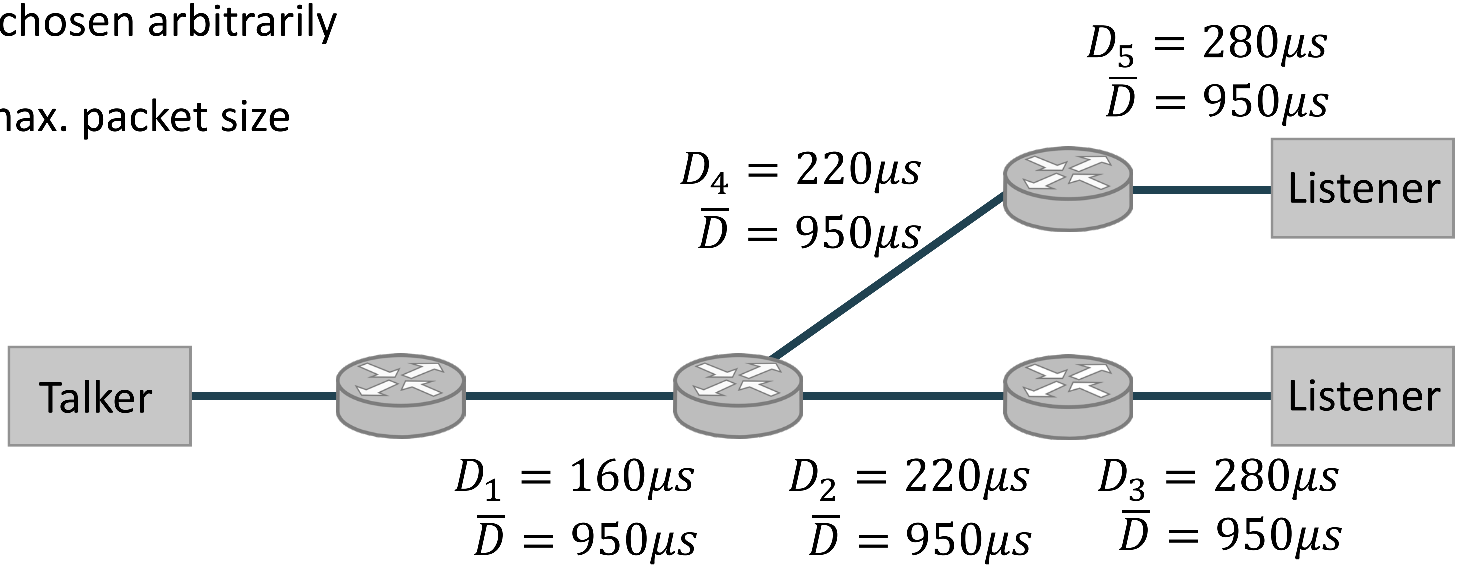
Reliable Reservation Protocol

Delay Analysis

How to determine the actual/current delay D?

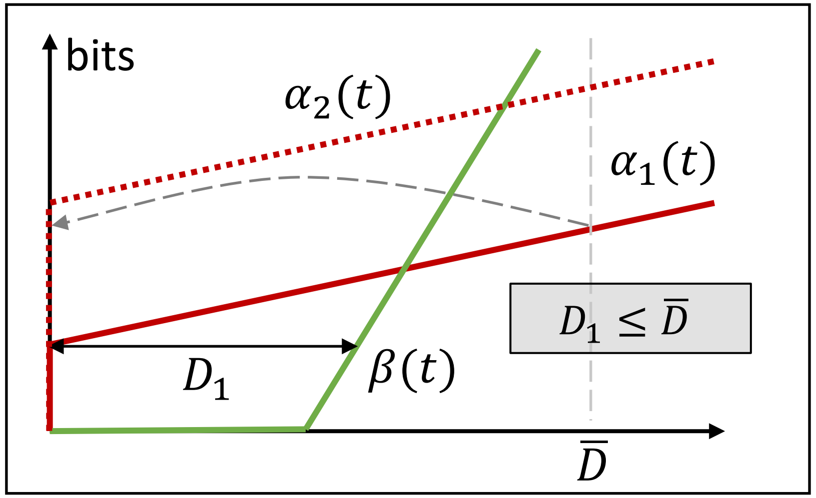
- using Network Calculus: $D \leq h(\alpha, \beta)$
  with h being the max. horizontal distance
- only using locally available information
- for the simplicity of the presentation, we omit hardware delays
- arrival curve $\alpha(t)$ from TSpec → CMI can be chosen arbitrarily
- service curve $\beta(t)$ from $idSl$, link rate, and max. packet size

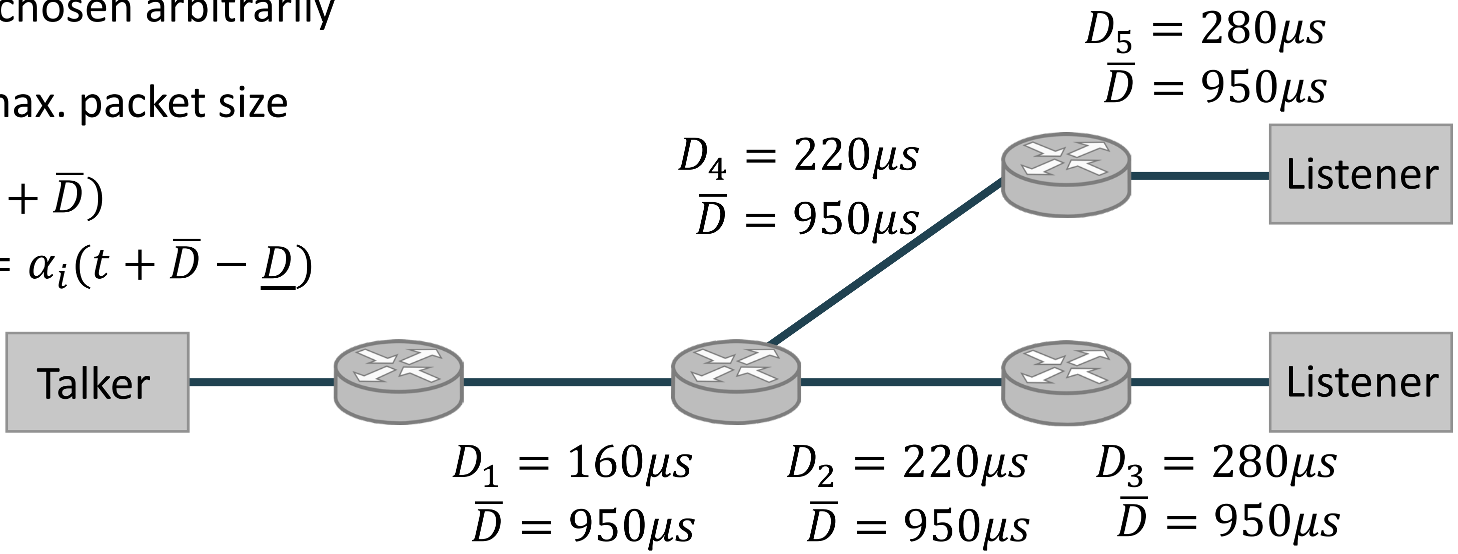




How to determine the actual/current delay D?

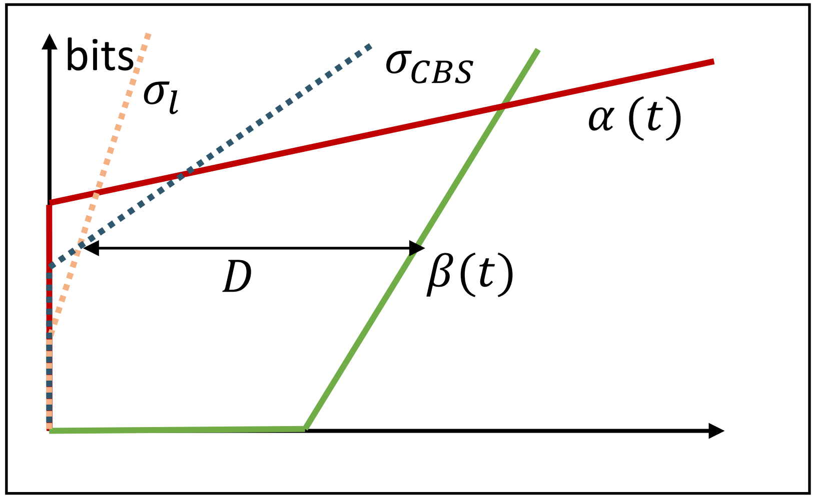
- using Network Calculus: $D \leq h(\alpha, \beta)$
  with h being the max. horizontal distance
- only using locally available information
- for the simplicity of the presentation, we omit hardware delays
- arrival curve $\alpha(t)$ from TSpec → CMI can be chosen arbitrarily
- service curve $\beta(t)$ from $idSl$, link rate, and max. packet size
- arrival output curve using $\overline{D}$: $\alpha_{i+1}(t) = \alpha_i(t + \overline{D})$
  or with min. Delay $\underline{D}$ (e.g., in RAP): $\alpha_{i+1}(t) = \alpha_i(t + \overline{D} - \underline{D})$





Reliable Reservation Protocol

Delay Analysis



How to determine the actual/current delay D?

- using Network Calculus: $D \leq h(\alpha, \beta)$
  with h being the max. horizontal distance
- only using locally available information
- for the simplicity of the presentation, we omit hardware delays
- arrival curve $\alpha(t)$ from TSpec → CMI can be chosen arbitrarily
- service curve $\beta(t)$ from $idSl$, link rate, and max. packet size
- arrival output curve using $\overline{D}$: $\alpha_{i+1}(t) = \alpha_i(t + \overline{D})$
  or with min. Delay $\underline{D}$ (e.g., in RAP): $\alpha_{i+1}(t) = \alpha_i(t + \overline{D} - \underline{D})$
- improved by shaping curves of the link σ_l

  and optionally by previous CBS queues σ_{CBS}

} Information available

→ Information needs to be added / distributed

Reliable Reservation Protocol

Summary

Worst Case Delay Bounds

- the worst case delay for the **current reservations** can be calculated locally in the bridges
  → efficient calculations using NC
- but: no **reservation-independent** upper bounds on the delay can be determined

Effect of Interference / Cross Flows

- by upper bounding the delay with $\overline{D}$, the effect of all cross-flows is also upper bounded
  → details do not need to be known

Delay Bounds $\overline{D}$

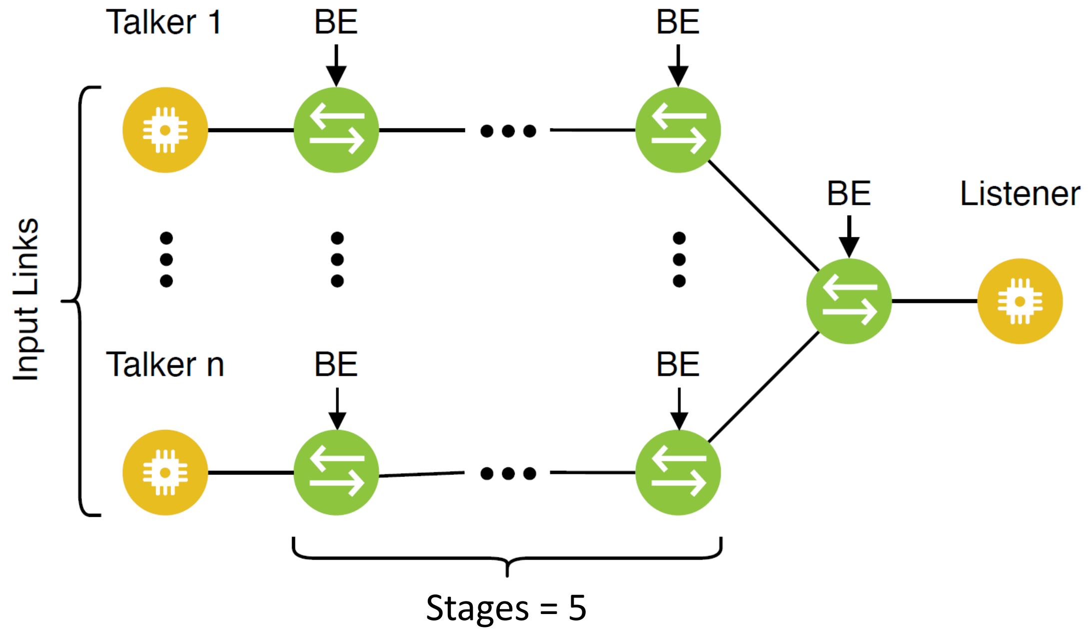
- values for $\overline{D}$ per queue need to be communicated to the devices
- reservations are safe with any values for $\overline{D}$, but the network efficiency can differ
- we need clever ways to find $\overline{D}$ (see for existing research slide 29)

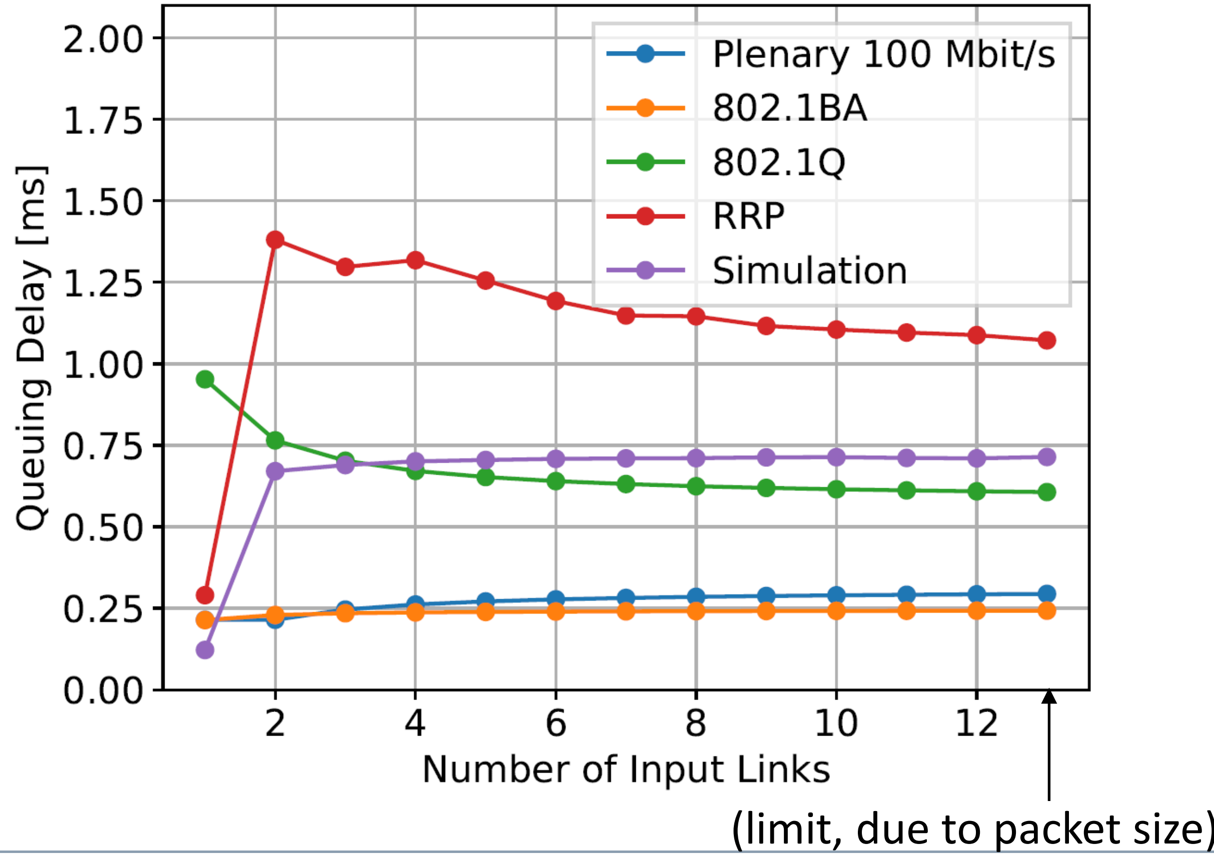
Evaluation

Validity of Worst Case Queuing Delay

Deadline Guarantees

- $\overline{D}$ from RRP versus the standard formulas and simulation results
- results illustrate that worst case is covered



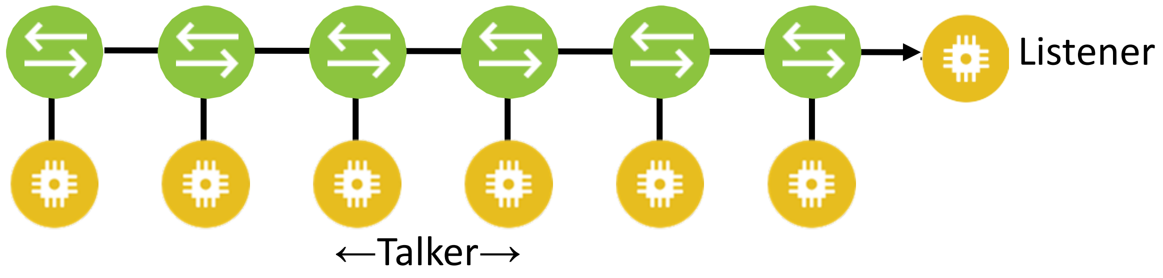


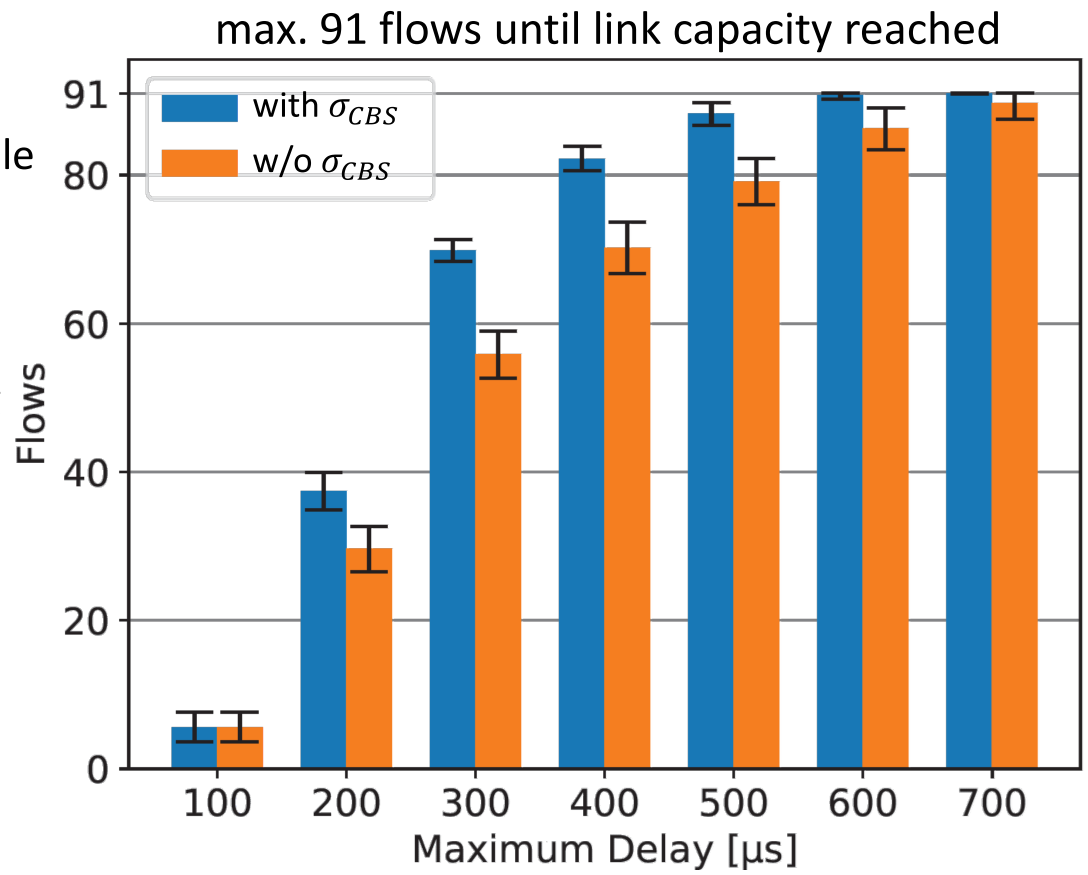
Evaluation

Improvement by Shaping Curve

CBS Shaping Curves Effect on Delay Guarantees

- 128 Byte packets, CMI = 125 µs, idleSlope of 75%, 1 Gbit/s links
- CBS shaping information of neighboring devices currently not available
- considering CBS shaping allows for more flow reservations
  → using RAP to distributing neighboring information is beneficial
- evaluation shows No. of Flow that can be reserved when the shaping of neighbors is locally considered (with σ_{CBS}) or not (w/o σ_{CBS})
- flows randomly distributed from different Talkers
  → error bars show standard deviation after 1000 repetitions





Conclusion

- TSN seeks to add delay guarantees to networks with decentralized control
- Currently, no worst case latency formula in the standards cover the actual worst case
- With the current procedure, bounding the worst case delay in CBS networks is not possible
- We introduced RRP - a new approach which can be used by the existing protocols
- CBS shaping information can increase the number of reservable flows
- We hope that our solutions help future standardization processes in TSN

References

General Topics

Presented Publication

L. Maile, D. Voitlein, A. Grigorjew, K.-S. J. Hielscher, and R. German, *“On the Validity of Credit-Based Shaper Delay Guarantees in Decentralized Reservation Protocols,”* in *Proceedings of the 31st International Conference on Real-Time Networks and Systems (RTNS ’23).* Dortmund, Germany. Association for Computing Machinery (New York, NY, USA), Jun. 2023, pp. 108–118. doi: 10.1145/3575757.3593644.

Related Publications

A. Grigorjew et al., *“Bounded Latency with Bridge-Local Stream Reservation and Strict Priority Queuing,”* in *11th International Conference on Network of the Future (NoF 2020),* Bordeaux, France: IEEE, Oct. 2020, pp. 55–63. doi: 10.1109/NoF50125.2020.9249224.

C. Boiger. *“Class A bridge latency calculations”.* Technical Report. 2010. *IEEE 802 November Plenary Meeting*. Online available: https://www.ieee802.org/1/ files/public/docs2010/ba-boiger-bridge-latency-calculations.pdf (2023-10-30).

L. Maile, K.-S. Hielscher, and R. German, *“Network Calculus Results for TSN: An Introduction,”* in *Information Communication Technologies Conference (ICTC 2020)*, Nanjing, China: IEEE, May 2020, pp. 131–140. doi: 10.1109/ICTC49638.2020.9123308.

References

Research on the Effects and Determination of Values for $\overline{D}$

A. Grigorjew, M. Seufert, N. Wehner, J. Hofmann, and T. Hoßfeld, *“ML-Assisted Latency Assignments in Time-Sensitive Networking,” IFIP/IEEE International Symposium on Integrated Network Management (IM 2021),* Bordeaux, France, 2021, pp. 116-124.

L. Maile, D. Voitlein, A. Grigorjew, K.-S. J. Hielscher, and R. German, *“Combining Static and Dynamic Traffic with Delay Guarantees in Time-Sensitive Networking,” International Conference on Performance Evaluation Methodologies and Tools (ValueTools 2023),* Heraklion, Greece. Springer LNICST Series, forthcoming, 2023.

Thank you!

More information?

Lisa Maile
lisa.maile@web.de
Friedrich-Alexander-Universität Erlangen-Nürnberg