FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

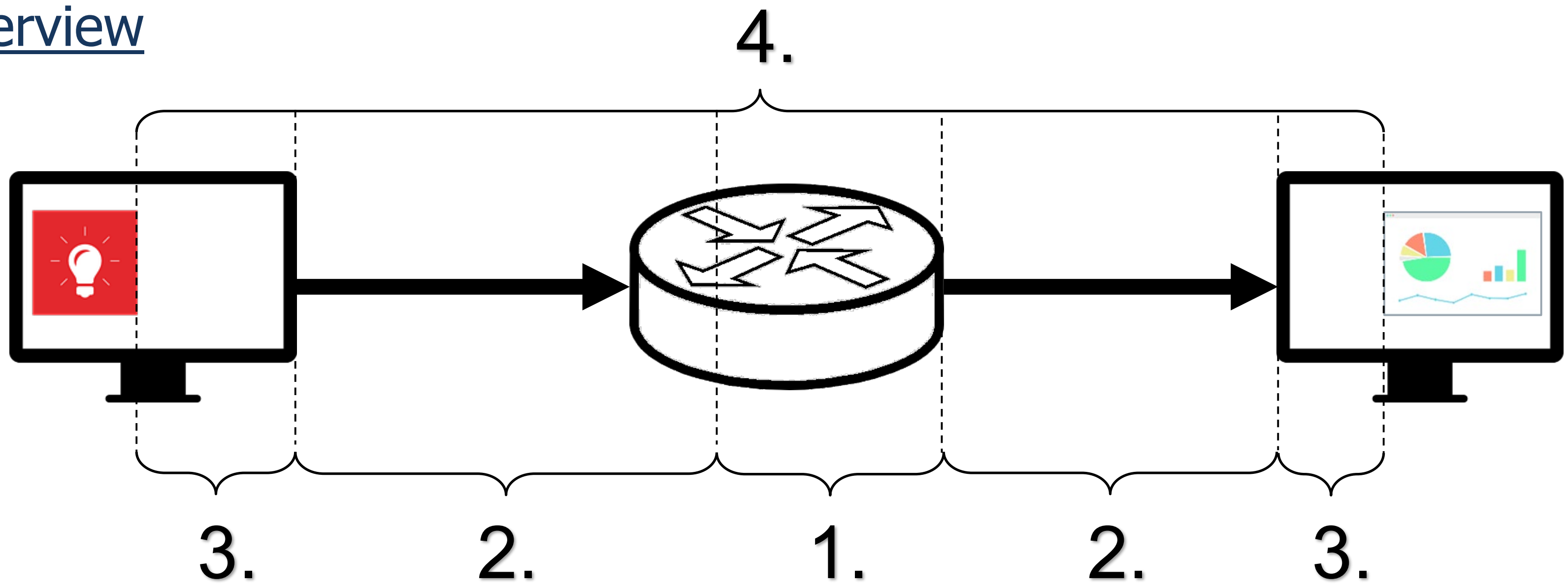
Lisa Maile
Computer Science 7, FAU Erlangen-Nürnberg

WoNeCa2020

Network Calculus Results for TSN: An Introduction

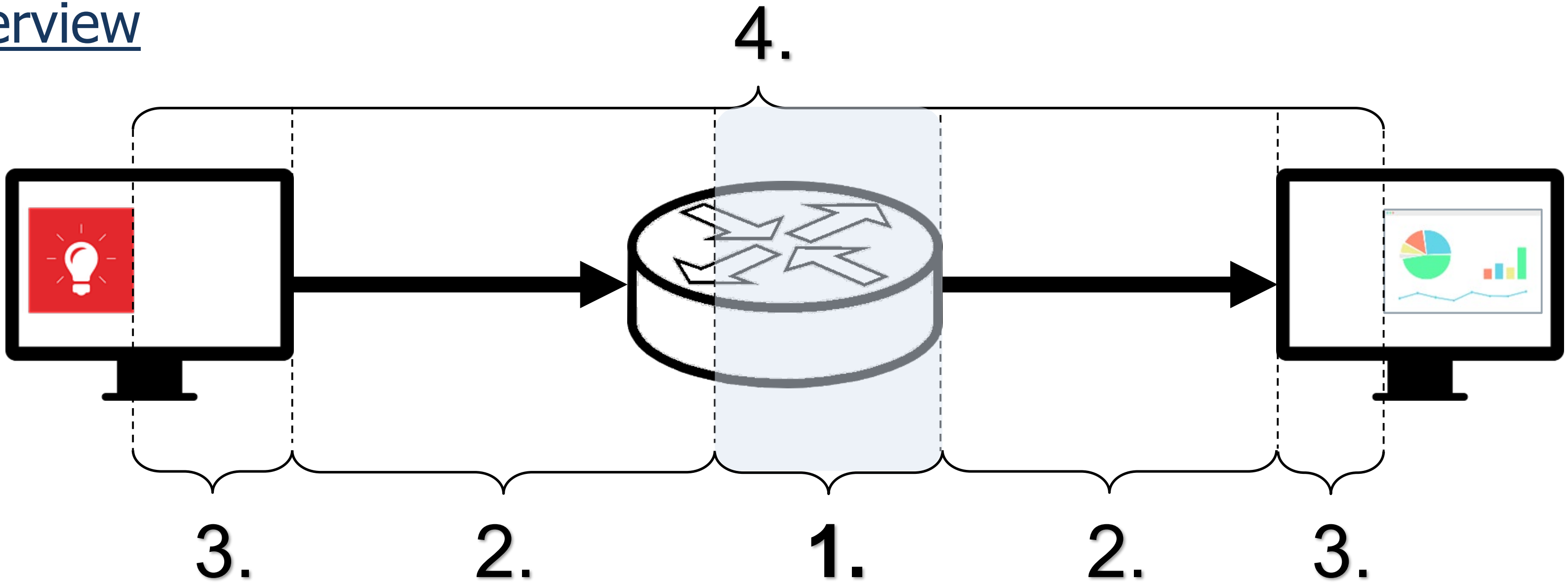
TIME-SENSITIVE URGENT NETWORKING

Overview



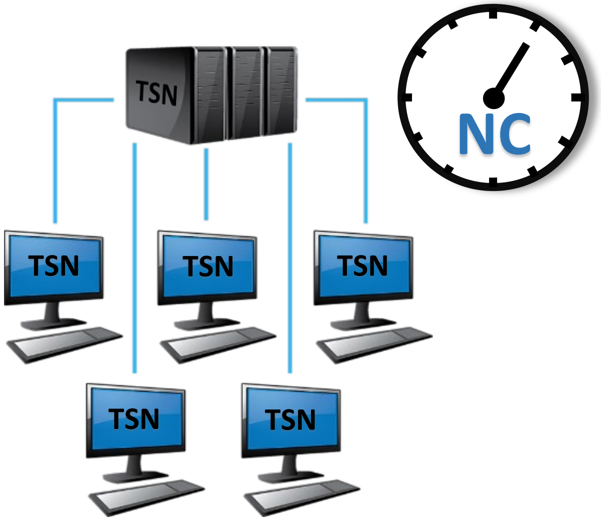
1. Queueing Delay - TSN schedulers, service and shaping curves
2. Hardware Delays - transmission, propagation, processing delays
3. Application Delay - arrival curve, interrupt latency
4. End-to-End Delay - overview, effect

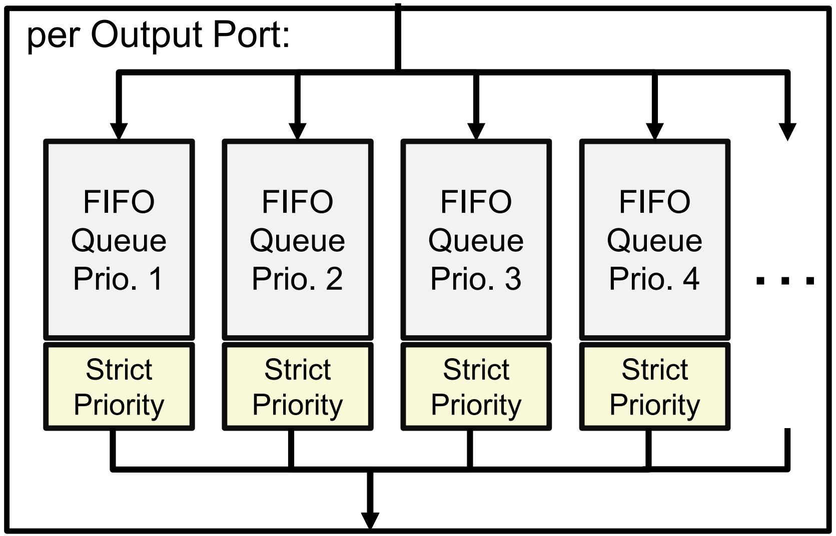
Overview



1. **Queueing Delay** - **TSN schedulers, service and shaping curves**
2. Hardware Delays - transmission, propagation, processing delays
3. Application Delay - arrival curve, interrupt latency
4. End-to-End Delay - overview, effect

Time-Sensitive Networking



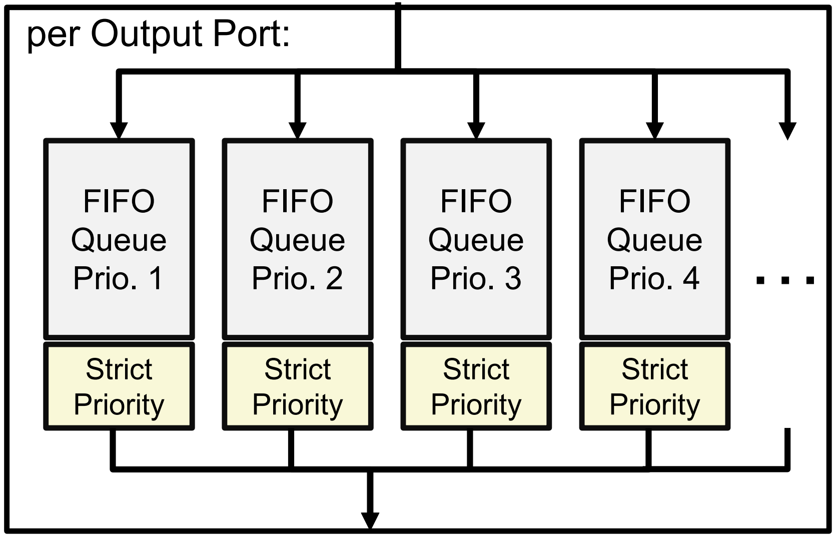


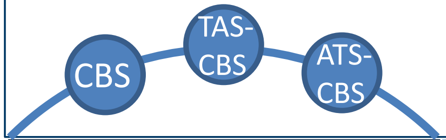
Time-Sensitive Networking

= real-time Ethernet Standards
(IEEE TSN Working Group)

TSN Scheduler

- Credit-Based Shaper (CBS)
- Asynchronous Traffic Shaping (ATS)
- Time-Aware Shaper (TAS, also often TT)
- Combinations





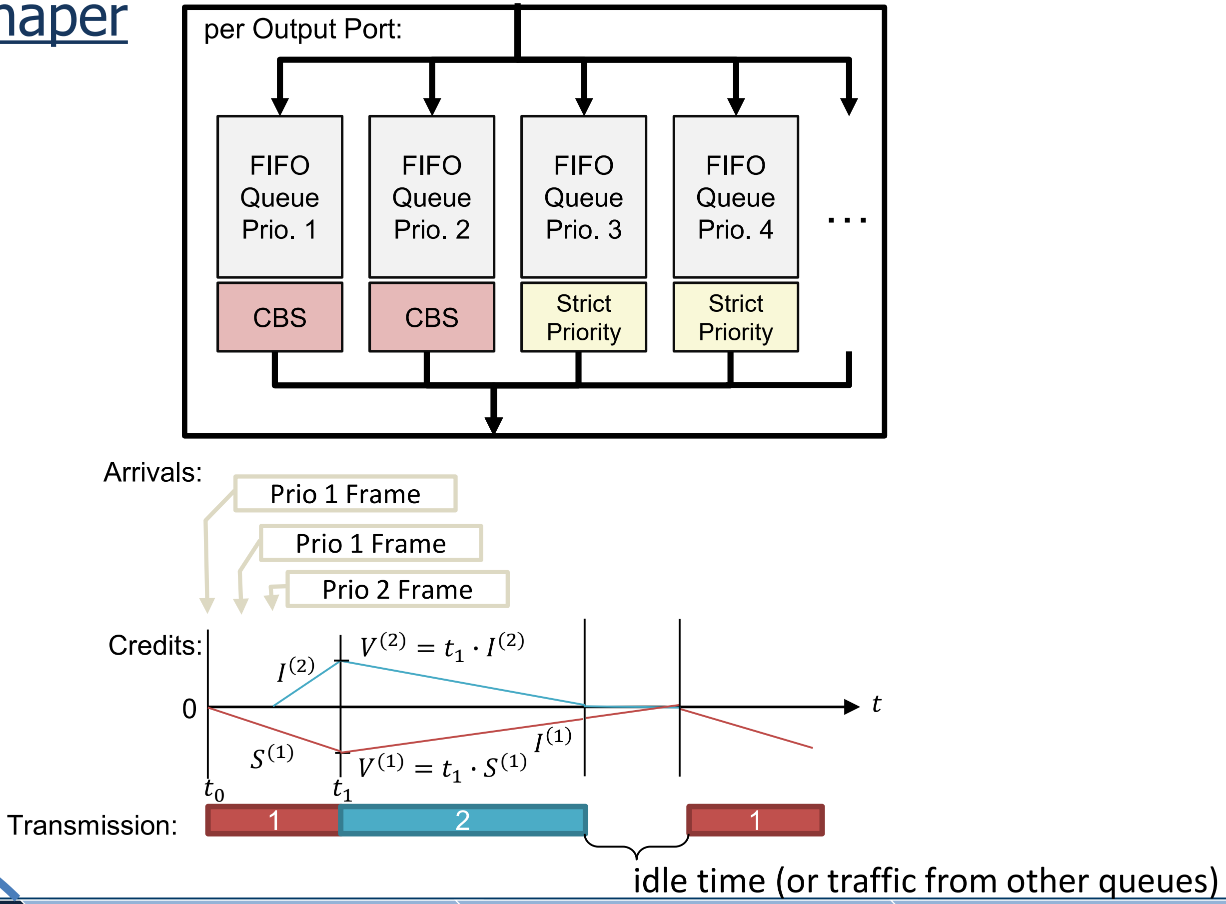
Survey

Source	Mechanism	Author	Year	Pre-emption	Work Basis	Impact of CDT	Arrival α	Min. Service β	Max. Service β^{max}	Shaper σ	Max. Output α^*	Delay	Backlog
[31]	CBS	R. Queck	July 2012	-	[48]	no	leaky-bucket	min. 2 CBS & x SP			CBS & SP	CBS & SP	
[27]	CBS	J. A. Ruiz De Azua *et al.*	Oct. 2014	-	[31]	no	detailed	min. & strict for 2 CBS & strict for SP	2 CBS	2 CBS			
[49]	CBS	Lin Zhao *et al.*	Nov. 2018	-	[27]	no	detailed	min. 3 CBS					
[41]	TAS	Luxi Zhao *et al.*	July 2018	no	[29] [42]	-	leaky-bucket	min. TT			TT	TT	
[50]	TAS-CBS	F. He *et al.*	May 2017	no	[27]	yes	leaky-bucket			2 CBS	CBS incl. shaper		
[29]	TAS-CBS	Luxi Zhao *et al.*	April 2018	yes&no	[27]	yes	leaky-bucket	min. 2 CBS			CBS	CBS	
[26]	TAS-CBS	H. Daigmorte *et al.*	June 2018	yes&no	[29] [27]	yes	detailed	min. & strict for x CBS		x CBS			
[28]	TAS-CBS	Luxi Zhao *et al.*	Dec. 2018	no	[26] [29] [27]	yes	leaky-bucket	min. x CBS		x CBS & link	CBS incl. shaper	CBS	
[37] & [35]	ATS-CBS	E. Mohammadpour *et al.*	Sep. 2018	-	[27]	yes	$=\alpha^*$	min. 2 CBS / min. x CBS [32]			CBS incl. link	CBS	CBS

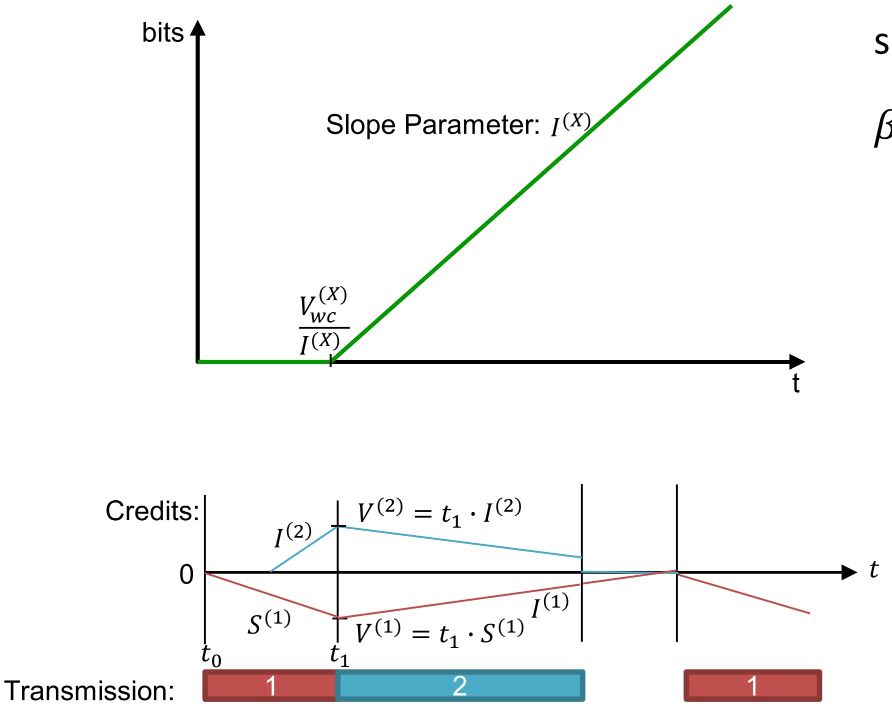
- common notation?

Maile, L. and Hielscher, K. and German, R.,
'Network Calculus Results for TSN: An Introduction', 2020
Information Communication Technologies Conference

Credit Based Shaper



Credit Based Shaper



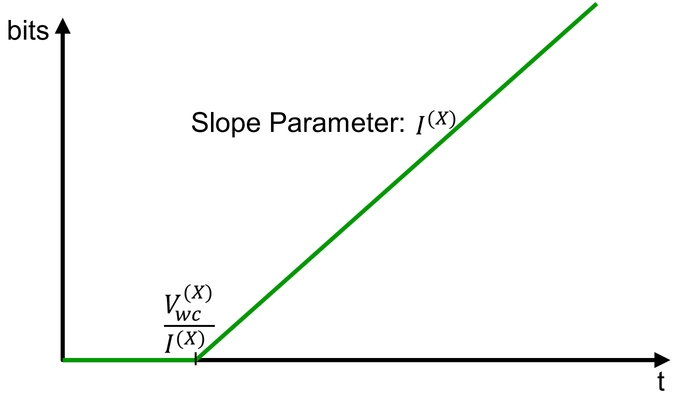
service curve for class X:

$$\beta_{CBS}^{X}(t) = I^{X} * \left(t - \frac{V_{wc}^{X}}{I^{X}}\right)$$

time to reach worst-case idle credit of queue

R. Queck [1]
J. A. R. De Azua and M. Boyer [2]
L. Zhao, P. H. M. Pop, Z. Zheng, H. Daigmorte, and M. Boyer [3]

Credit Based Shaper

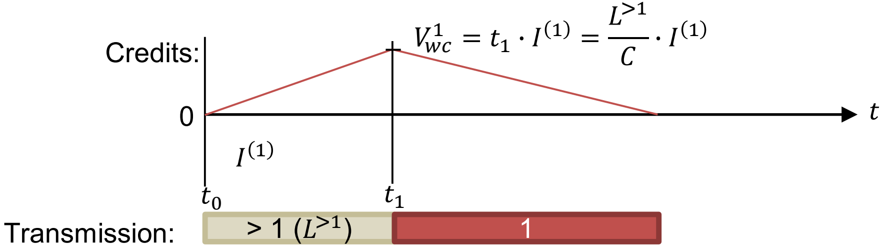


service curve for class X:

$$\beta^X_{CBS}(t) = I^X * \left(t - \frac{V^X_{wc}}{I^X}\right)$$

time to reach worst-case idle credit of queue

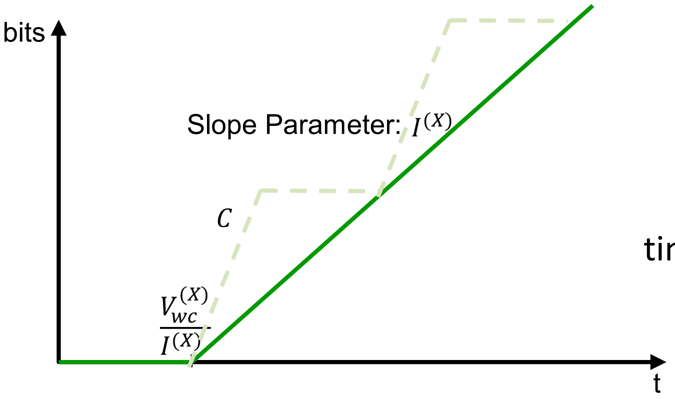
worst-case idle credit of queue 1 (V^1_{wc}):



R. Queck [1]
J. A. R. De Azua and M. Boyer [2]
L. Zhao, P. H. M. Pop, Z. Zheng, H. Daigmorte, and M. Boyer [3]

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

Credit Based Shaper

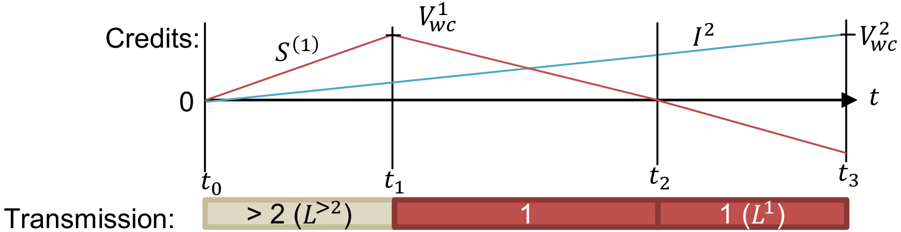


service curve for class X:

$$\beta_{CBS}^{X}(t) = I^{X} * \left(t - \frac{V_{wc}^{X}}{I^{X}}\right)$$

time to reach worst-case idle credit of queue

worst-case idle credit of queue 2 (V_{wc}^{2}):

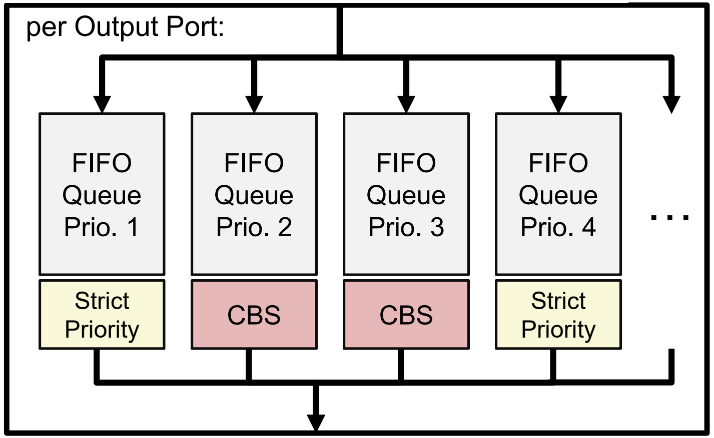


$$V_{wc}^{2} = t_3 \cdot I^{(2)} = \left(\frac{L^{>2}}{C} + \frac{L^{1}}{C} + \frac{V_{wc}^{1}}{-S^{(1)}}\right) \cdot I^{(2)}$$

R. Queck [1]
J. A. R. De Azua and M. Boyer [2]
L. Zhao, P. H. M. Pop, Z. Zheng, H. Daigmorte, and M. Boyer [3]
E. Mohammadpour, E. Stai, and J. L. Boudec [4, 5]

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

Control Data Traffic (CDT)



= higher prioritized queue

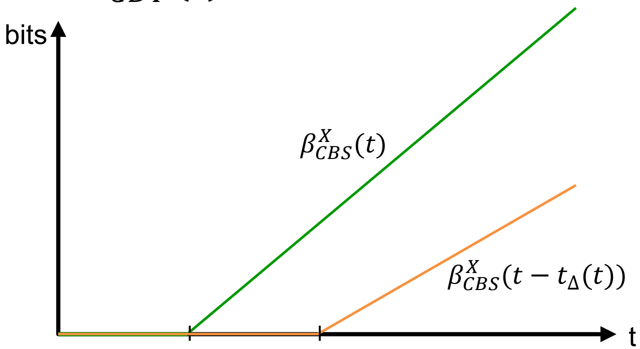
- CBS queues cannot send while CDT queue sends
- time that CDT sends until time t: $t_\Delta(t)$
- change CBS service curve to:

$$\beta^X_{CBS\_CDT}(t) = \beta^X_{CBS}(t - t_\Delta(t))$$

- with $t_\Delta(t)$ determined by the output of the CDT queue $\alpha^*_{CDT}(t)$ and the link rate C

$$t_\Delta(t) \le \frac{\alpha^*_{CDT}(t)}{C} = \frac{(\alpha_{CDT} \oslash \beta_{CDT})(t)}{C}$$

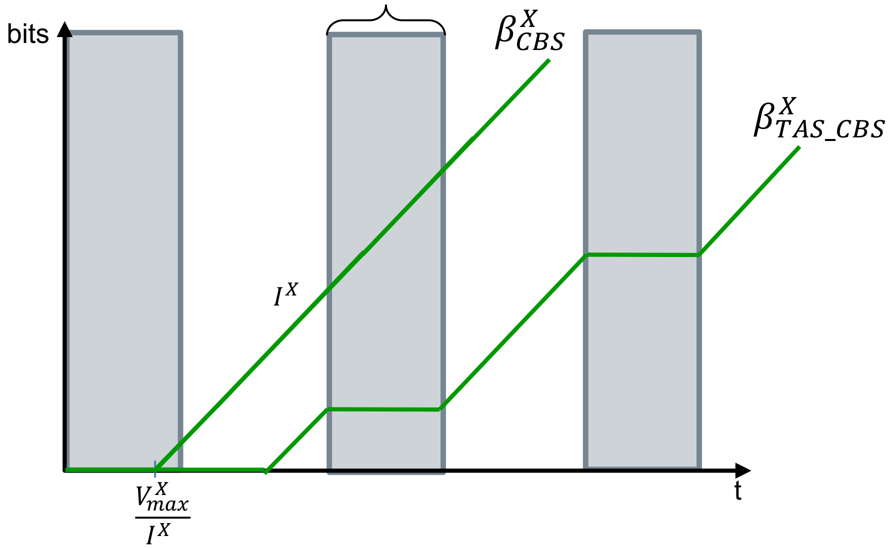
- with $\alpha_{CDT} = b + rt$ and $\beta_{CDT} = C \cdot \left[t - \frac{L^{>1}}{C}\right]$

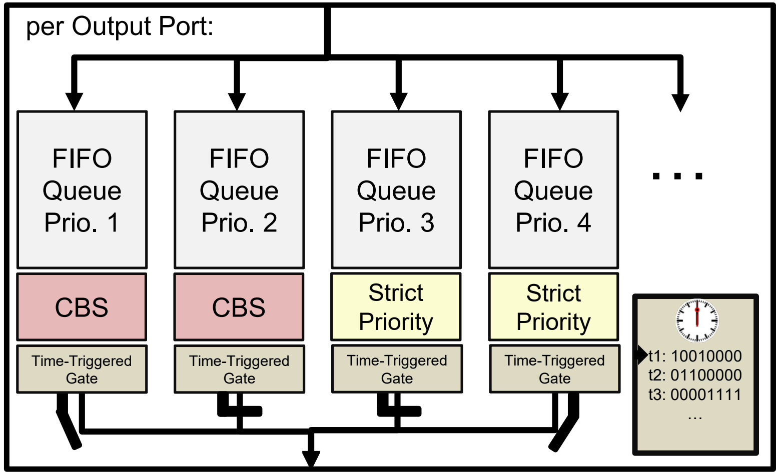


E. Mohammadpour, E. Stai, M. Mohiuddin, and J. Le Boudec [6]

Time Aware Shaper (TAS) and CBS

gate closed and guard band (depending on integration mode)



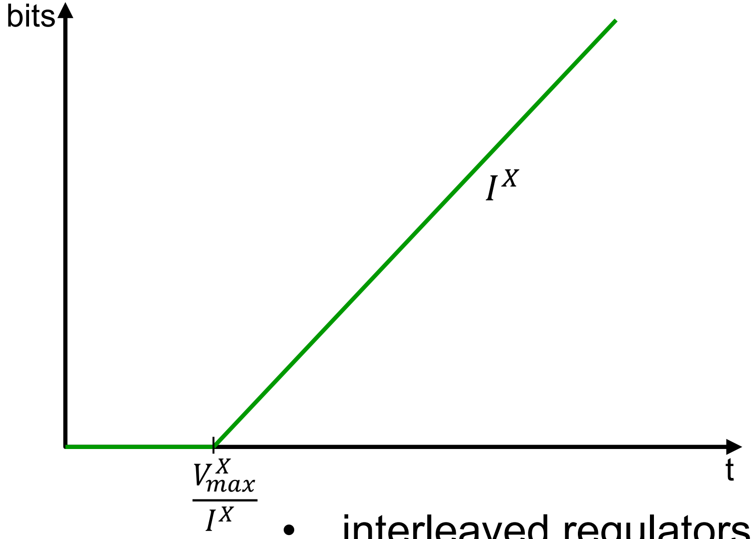


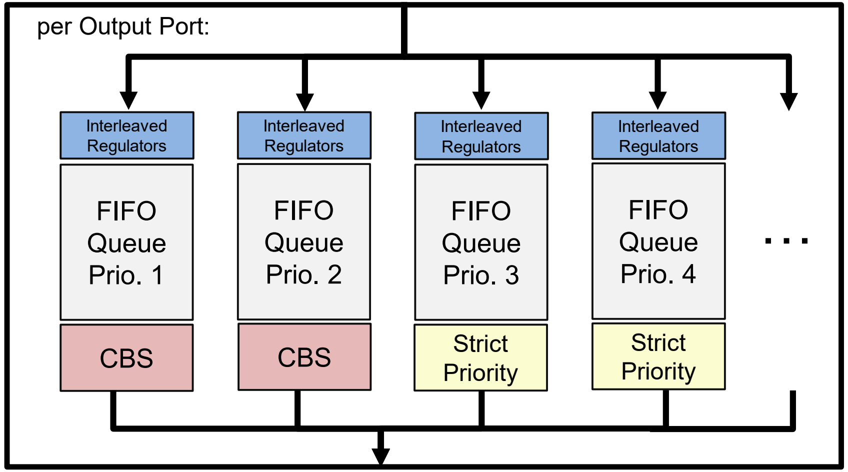
- CBS queues cannot send while gate is closed or during guard band $t_{GATE}(t)$
- change CBS service curve to:

$$\beta^X_{TAS\_CBS}(t) = \beta^X_{CBS}(t - t_{GATE}(\mathrm{t}))$$

H. Daigmorte, M. Boyer, and L. Zhao [7]
L. Zhao, P. Pop, Z. Zheng, and Q. Li [8]
L. Zhao, P. H. M. Pop, Z. Zheng, H. Daigmorte, and M. Boyer [9, 10]

Asynchronous Traffic Shaping (ATS) and CBS

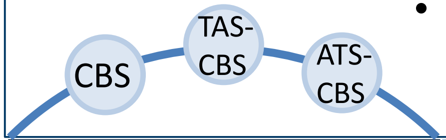




- interleaved regulators come for free
  (for the overall worst-case delay of all flows in a queue)
- service curve:

$$\beta^X_{\text{ATS\_}CBS}(t) = \beta^X_{CBS}(t)$$

E. Mohammadpour, E. Stai, M. Mohiuddin, and J. Le Boudec [11]



Further Curves

- also consider effect of link shaping

- strict service and shaping curves
  - for CBS and TAS-CBS
  - by assuming credit of queue at t=0 is either the min. or max. value
  - non-greedy shaping curve σ, output is upper bounded:

$$\alpha^* = \min\{\alpha \oslash \beta, \sigma\}$$

Mechanism	Author	Min. Service β	Max. Service β^{max}	Shaper σ	Max. Output α^*
CBS	R. Queck	min. 2 CBS & x SP			CBS & SP
CBS	J. A. Ruiz De Azua *et al.*	min. & strict for 2 CBS & strict for SP	2 CBS	2 CBS	
CBS	Lin Zhao *et al.*	min. 3 CBS			
TAS	Luxi Zhao *et al.*	min. TT			TT
TAS-CBS	F. He *et al.*			2 CBS	CBS incl. shaper
TAS-CBS	Luxi Zhao *et al.*	min. 2 CBS			CBS
TAS-CBS	H. Daigmorte *et al.*	min. & strict for x CBS		x CBS	
TAS-CBS	Luxi Zhao *et al.*	min. x CBS		x CBS & link	CBS incl. shaper
ATS-CBS	E. Mohammad-pour *et al.*	min. 2 CBS / min. x CBS [32]			CBS incl. link

Class- vs. Flow-Based

- all results are class-based
  - service curves are defined for FIFO systems, so no strictness required
  - so the residual (flow-based) service curve for flow j is:

$$\beta_j^\theta(t) = \left[\beta(t) - \sum_{\forall k \mid k \neq j} \alpha_k(t-\theta)\right]^+$$

  - θ can be optimized:

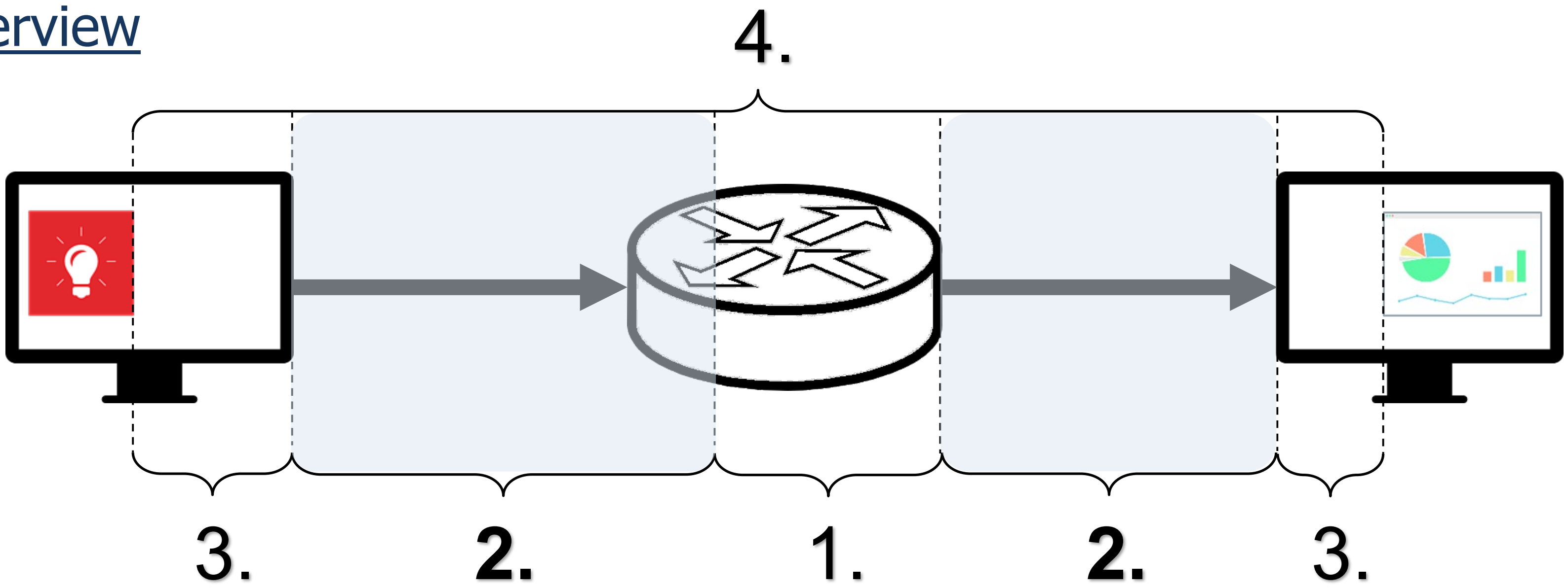
$$\alpha_j^* = \inf_{0 \leq \theta \leq t} \{\alpha_j \oslash \beta_j^\theta\}$$

- care: ATS can change the per-flow delay

➢ interesting future work

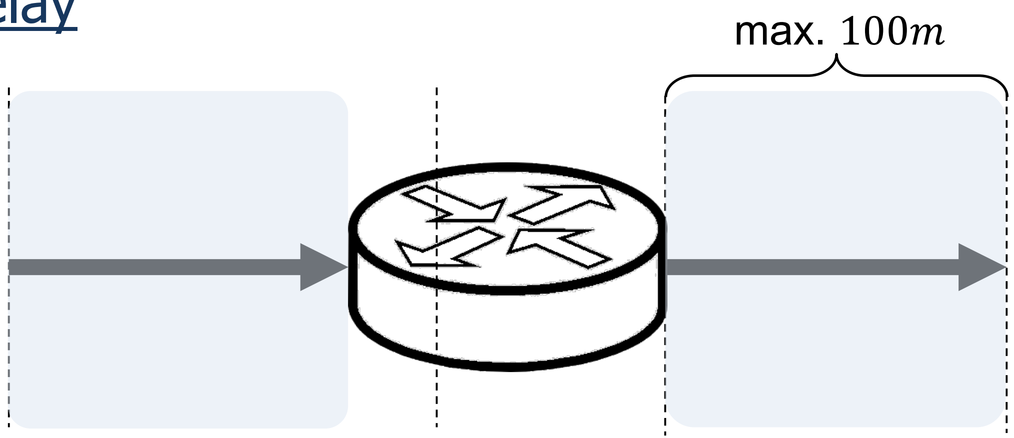
Le Boudec, J.Y., Thiran, P. [12]

Overview



1. Queueing Delay - TSN schedulers, service and shaping curves
2. **Hardware Delays** - **transmission, propagation, processing delays**
3. Application Delay - arrival curve, interrupt latency
4. End-to-End Delay - overview, effect

Propagation Delay

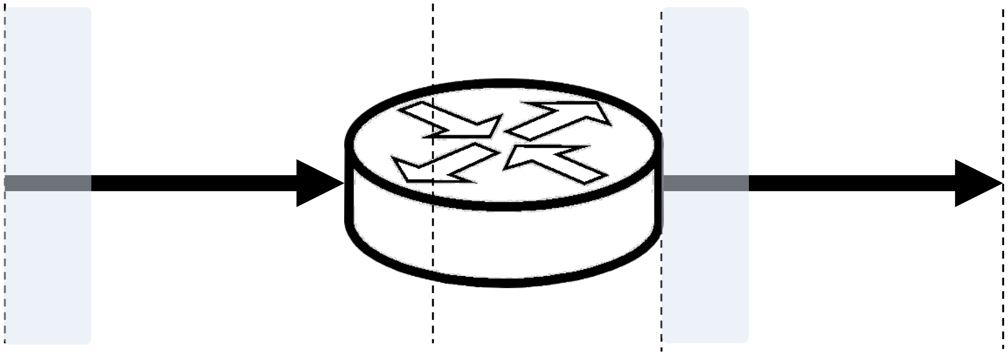


- delay bases on cable length s and propagation speed of medium v
  - Ethernet propagation speed $v = 2 \cdot 10^8 \frac{m}{s}$
  - max. Ethernet cable length $s = 100m$
  - worst-case propagation delay:

$$d_{pp} \leq 0.5\mu s$$

Lee, K.C., Lee, S. [13]

Transmission Delay



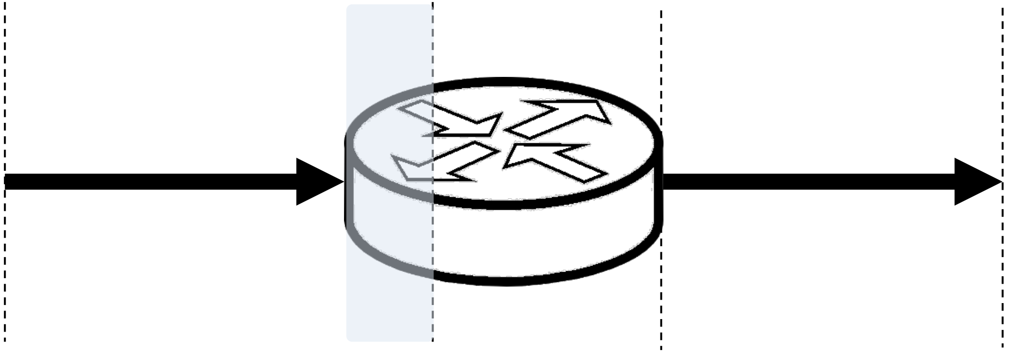
- after d_{pp} only one bit arrives
- also called: serialization, sending or reading delay
- transmission delay: whole serialized packet arrives
- in NC, this can be modeled with **packetizer**
- changes service curves to:

$$\beta'(t) = \beta(t) - l_{max}$$

for “store-and-forward”

for “cut-through” this is l_{header}

Processing Delay



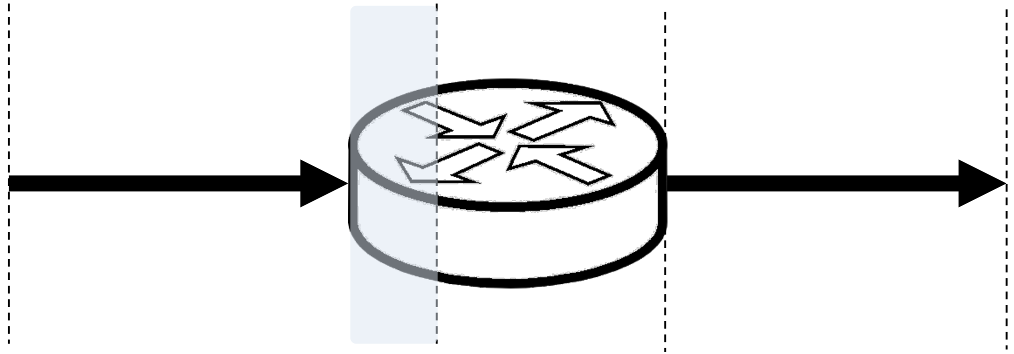
- few works deal with processing delays in forwarding devices
- for an estimation, [14] proposes to use:
  - number of instructions i
  - number of memory lookups m
  - average memory access time t
  - processor clock frequency f

(number of instructions, number of memory lookups, average memory access time): estimation from [14]

(processor clock frequency): TSN switch (TQ STKLS1028A)

Ramaswamy, R., Ning Weng, Wolf, T. [14]

Processing Delay

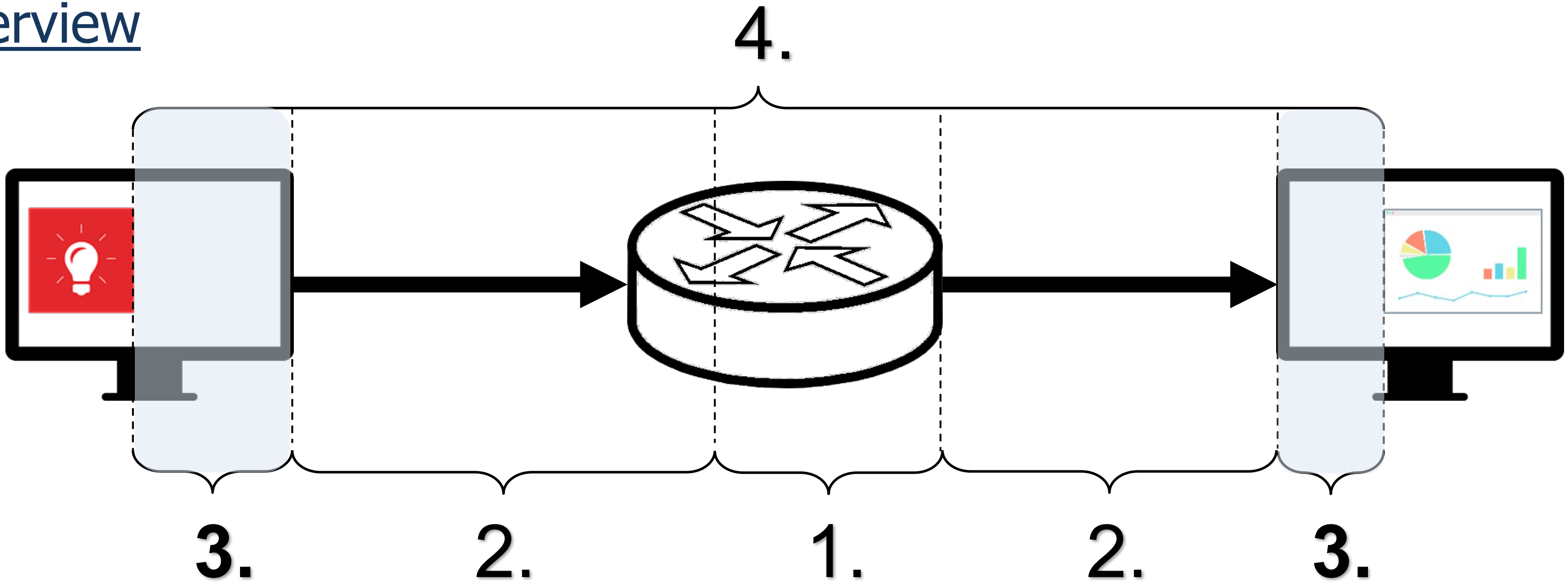


- few works deal with processing delays in forwarding devices
- for an estimation, [14] proposes to use:
  - number of instructions i $= 4693$
  - number of memory lookups m $= 947$
  - average memory access time t $= 4ns$
  - processor clock frequency f $= 1.3\ GHz$

$$d_{pc} = \frac{4693}{1.3GHz} + 947 \cdot 4ns = 7.4\mu s$$

Ramaswamy, R., Ning Weng, Wolf, T. [14]

Overview



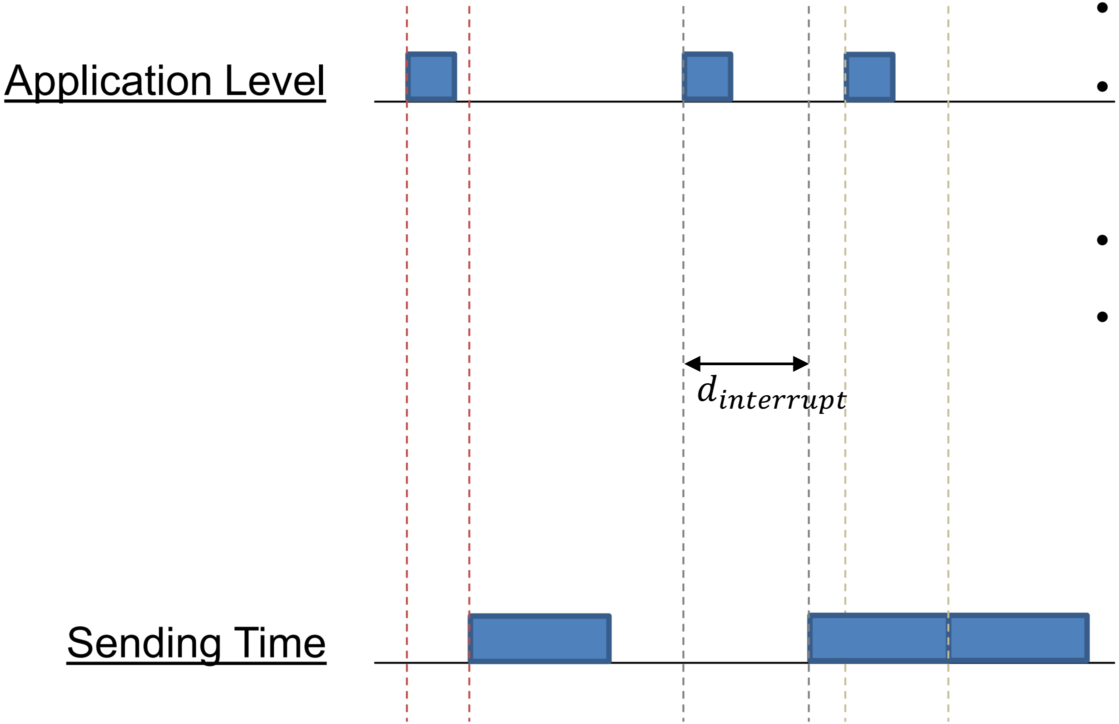
1. Queueing Delay
2. Hardware Delays
3. **Application Delay**
4. End-to-End Delay

- TSN schedulers, service and shaping curves
- transmission, propagation, processing delays
- **arrival curve, interrupt latency**
- overview, effect

Application Delay

- open research question: *How to create TSN applications?*
- we highlight possible scenarios and discuss their worst-case delays
- for guarantees, we need an upper bound on traffic that an application creates
  - arrival curve, e.g. b and r are required
  - can be derived e.g. by code analysis or observation/measurement of the application

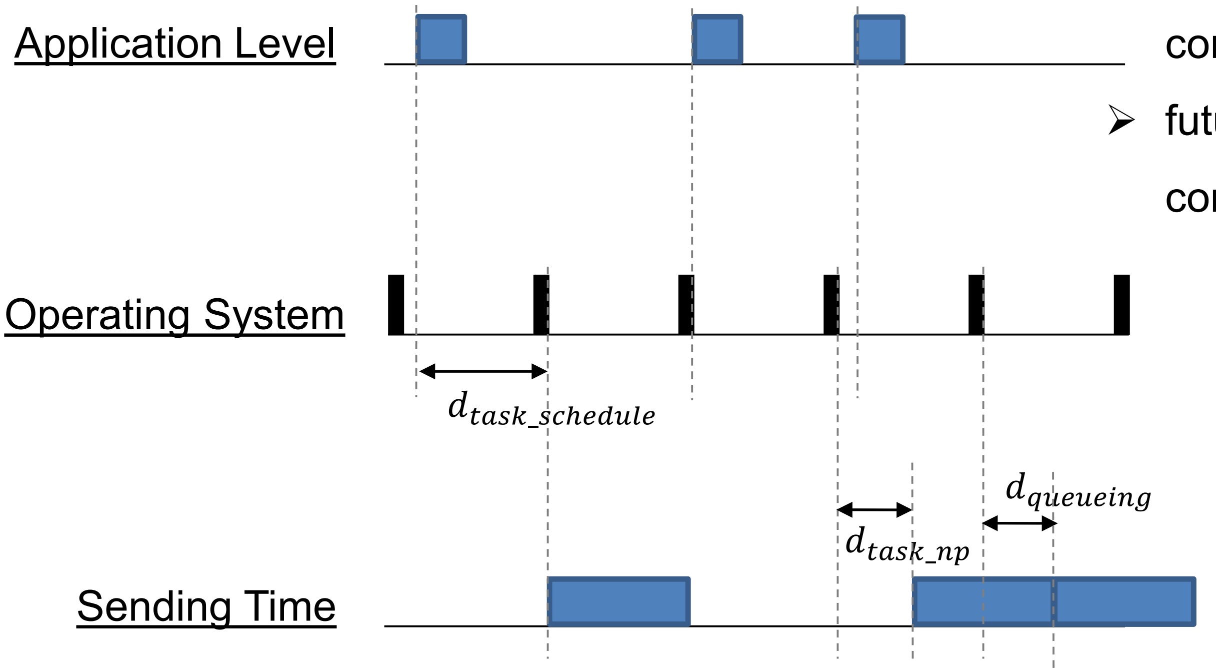
Unshaped



- $d_{interrupt}$ is highly variable
- bandwidth-intensive tasks:
  up to $140\mu s$ [15, Linux 2.6.9]
- worst-case is similar in RT-Linux
- file-intensive tasks:
  up to $11ms$ [15]

Liu, M., Liu, D., Wang, Y., Wang, M., Shao, Z. [15]

Time-Triggered Operating System (TTOS) [15]



- mainly for Time-Triggered communication
- ➢ future work: OS scheduling configuration

Kopetz, H. [16]

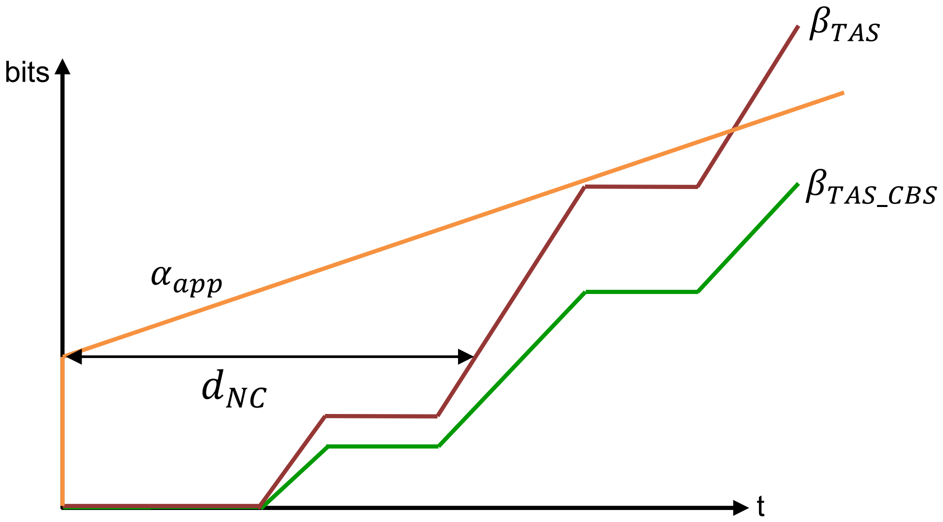
TSN scheduler & Class Measurement Interval (CMI)

- handle application to link delay as first hop

$$d_{send} = d_{interrupt} + d_{NC}$$

- future work:
  - service curve for ATS
  - service curve for CMI

    (how to implement this?)



Delay at Receiver

- receive packet (packetizer)

$$\frac{l_{max}}{C}$$

- trigger interrupt

$$d_{interrupt}$$

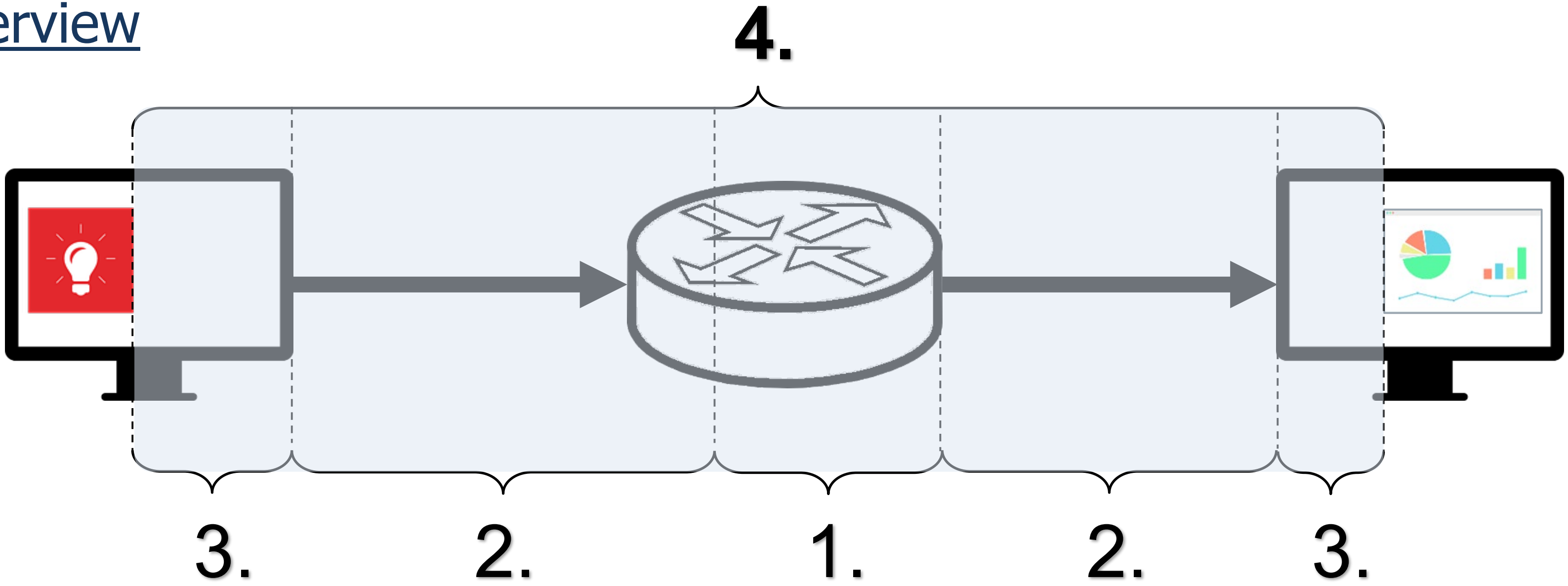
- process information

$$d_{pc}$$

- complete delay at receiver:

$$d_r = \frac{l_{max}}{C} + d_{interrupt} + d_{pc}$$

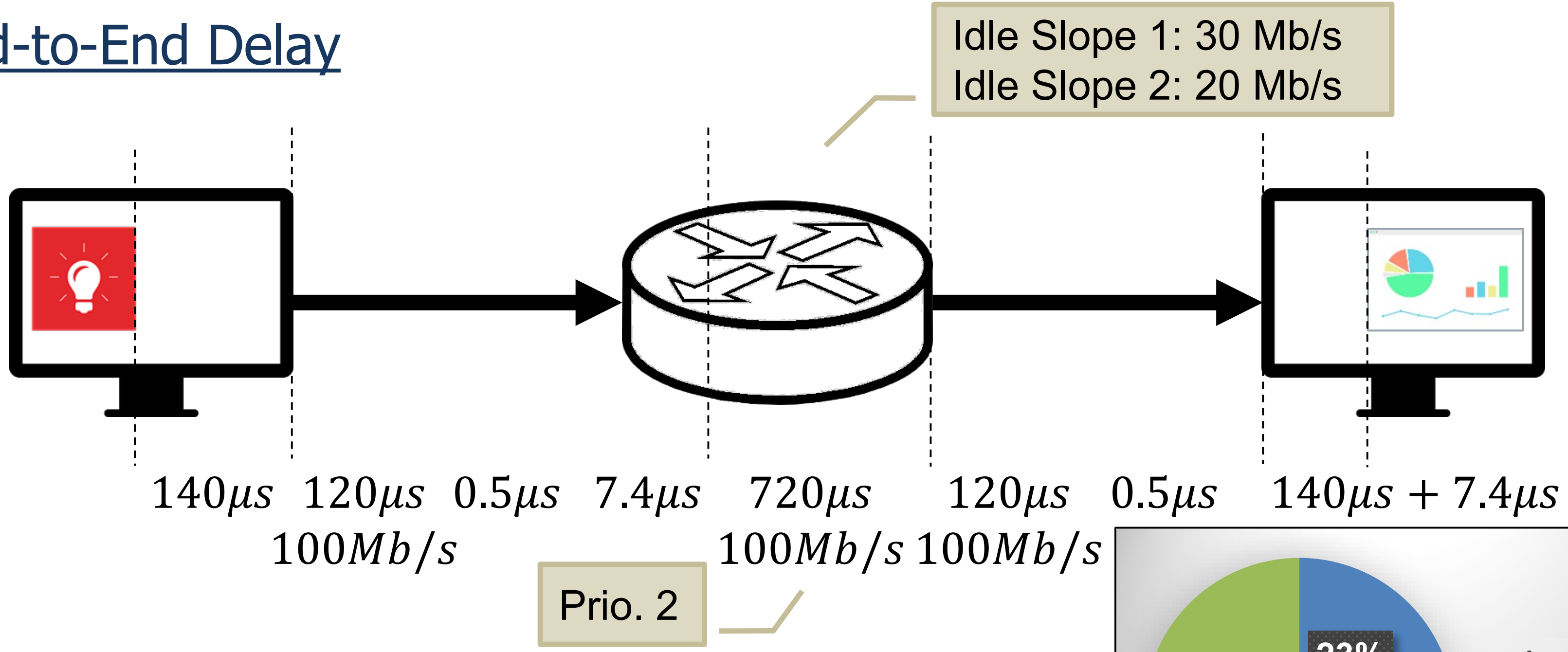
Overview

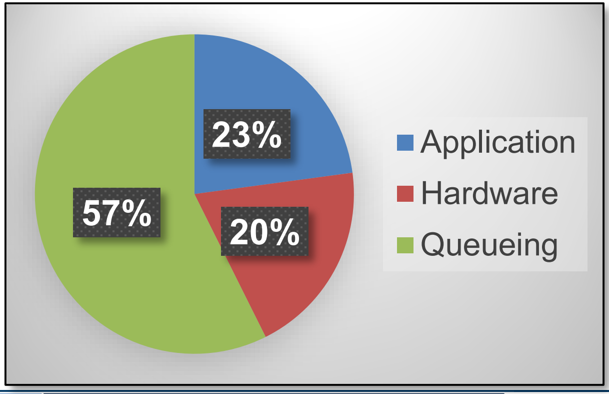


1. Queueing Delay
2. Hardware Delays
3. Application Delay
4. **End-to-End Delay**

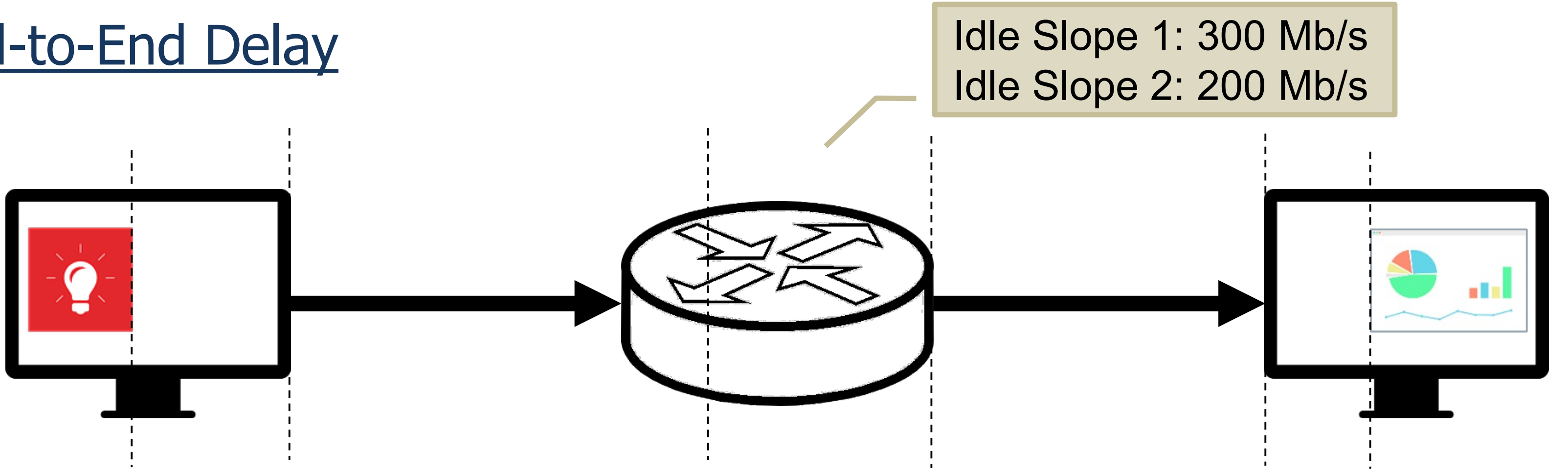
- TSN schedulers, service and shaping curves
- transmission, propagation, processing delays
- arrival curve, interrupt latency
- **overview, effect**

End-to-End Delay





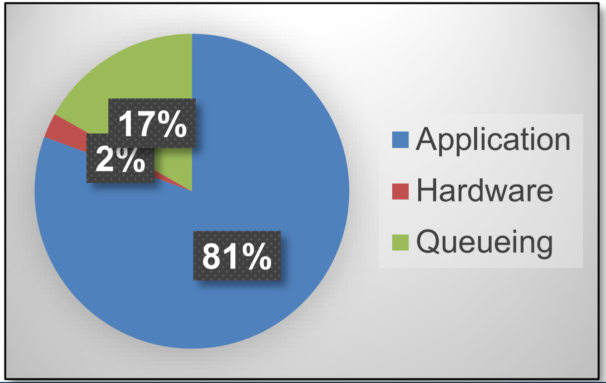
End-to-End Delay



$140\mu s$ $\mathbf{1.2\mu s}$ $0.5\mu s$ $7.4\mu s$ $\mathbf{60\mu s}$ $\mathbf{1.2\mu s}$ $0.5\mu s$ $140\mu s + 7.4\mu s$

$\mathbf{1Gb/s}$ $\mathbf{1Gb/s}$ $\mathbf{1Gb/s}$

- measured:
  - sending to receiving without queueing delay $\leq 3\mu s$
  - application to application without queueing delay $\leq 300\mu s$



[1] R. Queck, “Analysis of ethernet AVB for automotive networks using network calculus,” in IEEE International Conference on Vehicular Electronics and Safety (ICVES 2012). IEEE, Jul. 2012, pp. 61–67.

[2] J. A. R. De Azua and M. Boyer, “Complete modelling of AVB in network calculus framework,” in Proceedings of the 22nd International Conference on Real-Time Networks and Systems. Versaille, France: ACM Press, Oct. 2014, pp. 55–64.

[3] L. Zhao, P. H. M. Pop, Z. Zheng, H. Daigmorte, and M. Boyer, “Improving worst-case end-to-end delay analysis of multiple classes of avb traffic in tsn networks using network calculus,” 2019, [Preprint].

[4] E. Mohammadpour, E. Stai, and J. L. Boudec, “Improved credit bounds for the credit-based shaper in time-sensitive networking,” IEEE Networking Letters, vol. 1, no. 3, pp. 136–139, Sep. 2019.

[5] E. Mohammadpour, E. Stai, and J.-Y. Le Boudec, “Improved delay bound for a service curve element with known transmission rate,” IEEE Networking Letters, vol. 1, no. 4, pp. 156–159, 12 2019, conference Name: IEEE Networking Letters.

[6] E. Mohammadpour, E. Stai, M. Mohiuddin, and J. Le Boudec, “Latency and backlog bounds in time-sensitive networking with credit based shapers and asynchronous traffic shaping,” in 2018 30th International Teletraffic Congress (ITC 30), vol. 02, Sep. 2018.

[7] H. Daigmorte, M. Boyer, and L. Zhao, “Modelling in network calculus a TSN architecture mixing time-triggered, credit based shaper and best-effort queues,” Jun. 2018, [Preprint].

[8] L. Zhao, P. Pop, Z. Zheng, and Q. Li, “Timing analysis of AVB traffic in TSN networks using network calculus,” in Real-Time and Embedded Technology and Applications Symposium, Apr. 2018, pp. 25–36.

[9] L. Zhao, P. H. M. Pop, Z. Zheng, H. Daigmorte, and M. Boyer, “Improving worst-case end-to-end delay analysis of multiple classes of avb traffic in tsn networks using network calculus,” 2019, [Preprint].

[10] L. Zhao, P. Pop, Z. Zheng, H. Daigmorte and M. Boyer, "Latency Analysis of Multiple Classes of AVB Traffic in TSN with Standard Credit Behavior using Network Calculus," in IEEE Transactions on Industrial Electronics, doi: 10.1109/TIE.2020.3021638.

[11] E. Mohammadpour, E. Stai, M. Mohiuddin, and J. Le Boudec, “Latency and backlog bounds in time-sensitive networking with credit based shapers and asynchronous traffic shaping,” in 2018 30th International Teletraffic Congress (ITC 30), vol. 02, Sep. 2018.

[12] Le Boudec, J.Y., Thiran, P.: Network Calculus: A Theory of Deterministic Queuing Systems for the Internet. Springer-Verlag, Berlin, Heidelberg (2001)

[13] Lee, K.C., Lee, S.: Performance evaluation of switched ethernet for real-time industrial communications. Computer Standards & Interfaces 24(5), 411-423 (2002).

[14] Ramaswamy, R., Ning Weng, Wolf, T.: Characterizing network processing delay. In: IEEE Global Telecommunications Conference, 2004. GLOBECOM '04. vol. 3, pp. 1629-1634 Vol.3 (Nov 2004).

[15] Liu, M., Liu, D., Wang, Y., Wang, M., Shao, Z.: On improving realtime interrupt latencies of hybrid operating systems with two-level hardware interrupts. IEEE Transactions on Computers 60(7), 978-991 (July 2011).

[16] Kopetz, H.: Event-triggered versus time-triggered real-time systems. Springer Berlin Heidelberg (Operating Systems of the 90s and Beyond), pp. 86-101 (1991)