http://disco.cs.uni-kl.de

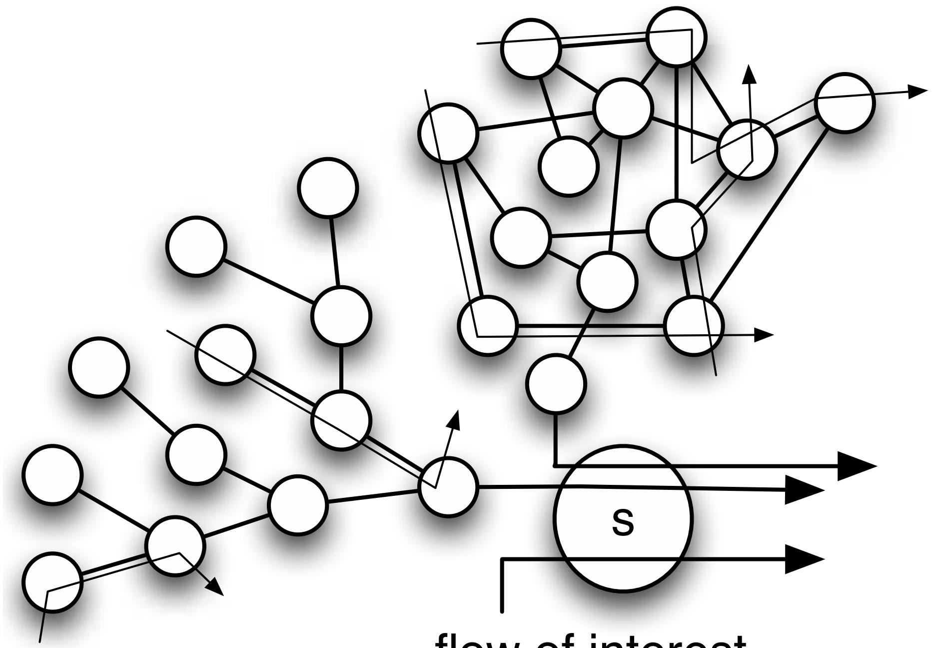
Bounding Flow Arrivals in Feed-Forward Networks

Steffen Bondorf

bondorf@cs.uni-kl.de

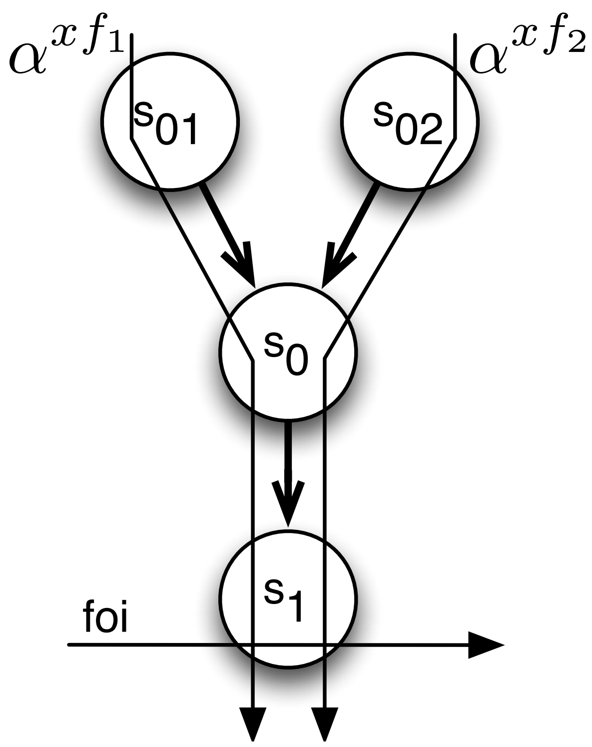
Bounding Flow Arrivals

- Where do we need these bounds?
  - At the locations of interference with the flow of interest.
- How do we derive them?
  - That's not that easy to answer.



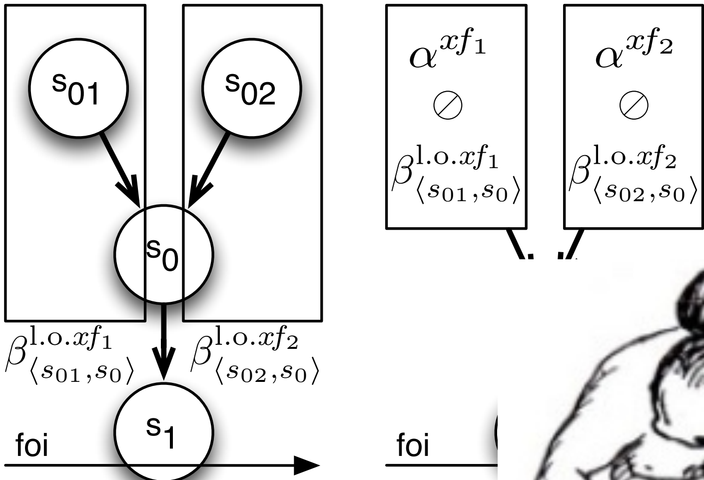
Cutting Down the Network

- The previous network is too complex to work with
- Cut down the network to the relevant part
  - Use output bounds, left-over service curves (arbitrary multiplexing), …
- Result:

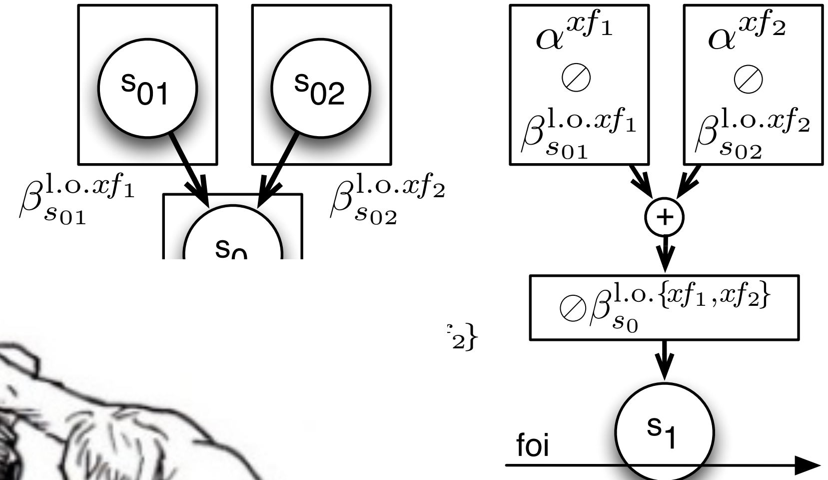


The Struggle: Segregation vs. Aggregation

Contender A:
Segregation of cross-flows



Contender B:
Aggregation of cross-traffic



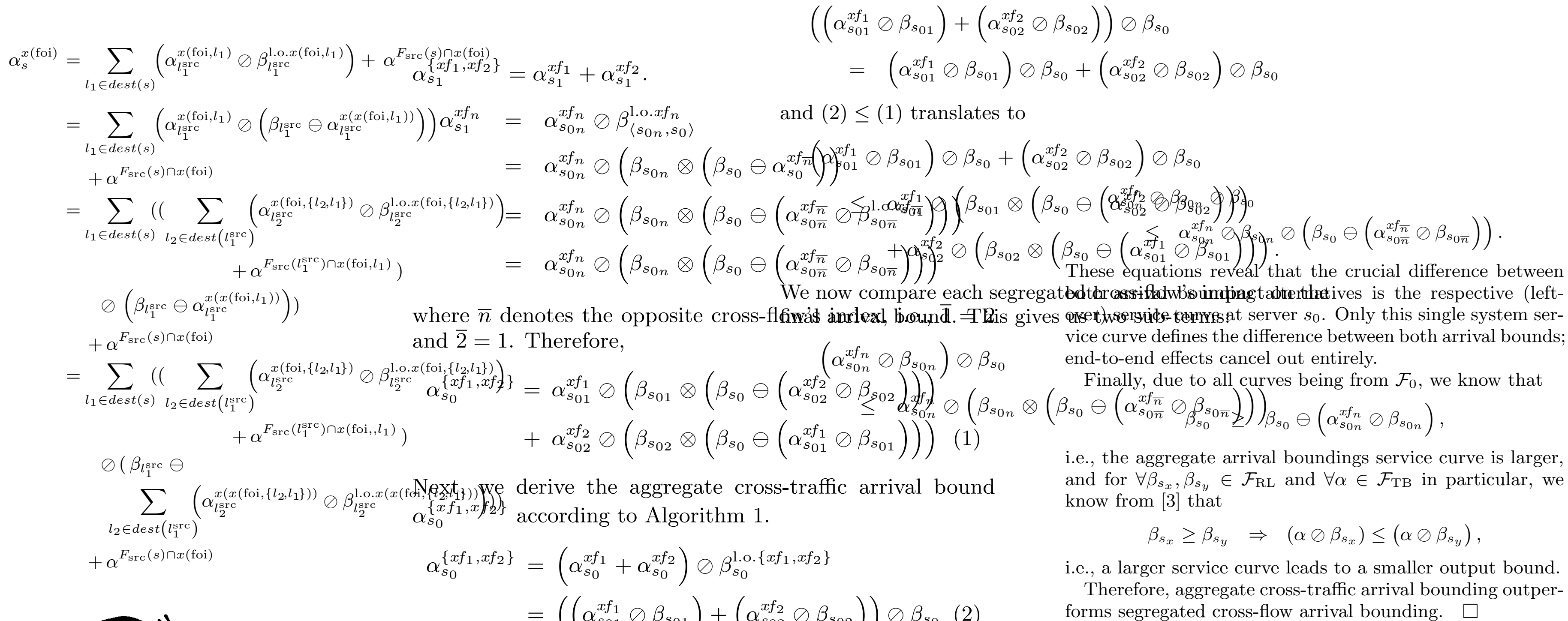
End-to-end analysis

Flow aggregation

Picture credits: http://fightland.vice.com/

Round 1: The Contenders Approach Each Other

- Pay Bursts Only Once
  - Subtraction before convolution
- Token buckets and rate latencies



where $\overline{n}$ denotes the opposite cross-flow's index, i.e., $\overline{1} = 2$ and $\overline{2} = 1$. Therefore,

Next, we derive the aggregate cross-traffic arrival bound $\alpha_{s_0}^{\{xf_1,xf_2\}}$ according to Algorithm 1.

$$\alpha_{s_0}^{\{xf_1,xf_2\}} = \left(\alpha_{s_0}^{xf_1} + \alpha_{s_0}^{xf_2}\right) \oslash \beta_{s_0}^{\text{l.o.}\{xf_1,xf_2\}} = \left(\left(\alpha_{s_{01}}^{xf_1} \oslash \beta_{s_{01}}\right) + \left(\alpha_{s_{02}}^{xf_2} \oslash \beta_{s_{02}}\right)\right) \oslash \beta_{s_0} \quad (2)$$

and (2) ≤ (1) translates to

These equations reveal that the crucial difference between both cross-flow bounding alternatives is the respective (left-over) service curve at server s_0. Only this single system service curve defines the difference between both arrival bounds; end-to-end effects cancel out entirely.

Finally, due to all curves being from $\mathcal{F}_0$, we know that

i.e., the aggregate arrival boundings service curve is larger, and for $\forall \beta_{s_x}, \beta_{s_y} \in \mathcal{F}_{\text{RL}}$ and $\forall \alpha \in \mathcal{F}_{\text{TB}}$ in particular, we know from [3] that

$$\beta_{s_x} \geq \beta_{s_y} \Rightarrow (\alpha \oslash \beta_{s_x}) \leq (\alpha \oslash \beta_{s_y}),$$

i.e., a larger service curve leads to a smaller output bound.

Therefore, aggregate cross-traffic arrival bounding outperforms segregated cross-flow arrival bounding. □

This round goes to aggregation [Valuetools2015]

Evaluation of Round 1

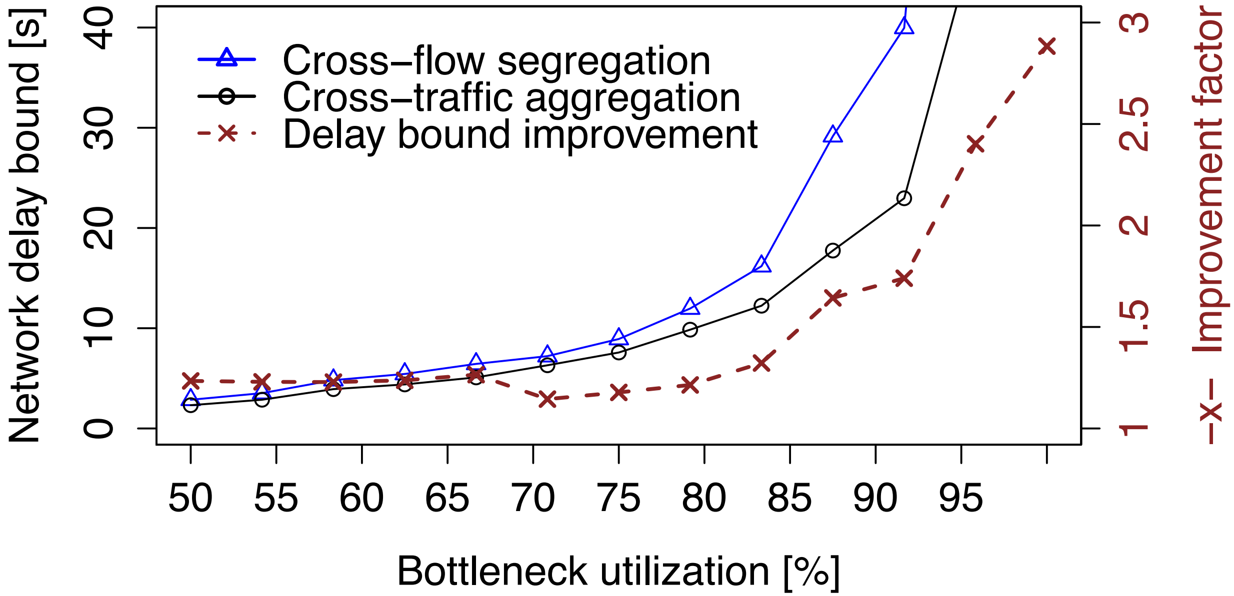
- Network Models
  - **Erdös-Rényi random graphs** using topology generator aSHIIP
    - flat and hierarchical
    - n=32 nodes and p=0.1 link probability resulting in: number of servers: flat: 114, hierarchical: 73
    - all servers are 100 Mbps links
  - Token-buckets with rate 1 Mbps and burst 1 Mb
  - Random source and sink, routed on shortest path
  - Tool: DiscoDNC v2 [Valuetools2014]

- Network delay bound D: maximum delay bound over all flows

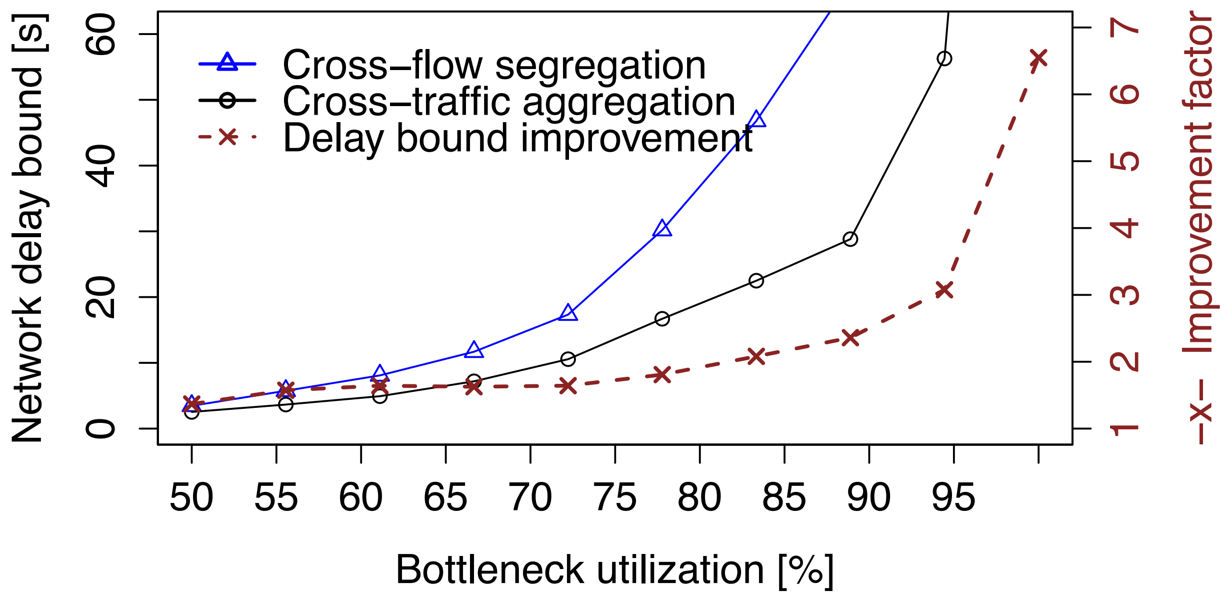
- Improvement factor: $\frac{D(\text{segregated xf-bounding})}{D(\text{aggregate xf-bounding})}$

Evaluation of Round 1

flat



hierarchical



Evaluation of Round 1

- Evaluation looks good
- Assumptions are limiting, yet crucial
  - Distributivity of deconvolution over addition [INFOCOM2015]

According to [3], Lemma 12, we can distribute the deconvolution of token-bucket arrivals with a rate-latency service curve over the aggregation. For (2), this means:

$$\left(\left(\alpha^{xf_1}_{s01} \oslash \beta_{s01}\right) + \left(\alpha^{xf_2}_{s02} \oslash \beta_{s02}\right)\right) \oslash \beta_{s0}$$
$$= \left(\alpha^{xf_1}_{s01} \oslash \beta_{s01}\right) \oslash \beta_{s0} + \left(\alpha^{xf_2}_{s02} \oslash \beta_{s02}\right) \oslash \beta_{s0}$$

Limitation to token buckets and rate latencies means:
It is not a technical KO!

Round 2: More General Curve Shapes

$$\left(\left(\alpha^{f_1} \oslash \beta\right) + \left(\alpha^{f_2} \oslash \beta\right)\right)(t) =$$

$$\sup\left\{ \sup_{i_\beta \in I_\beta} \left\{\alpha^{f_1}(t+i_\beta) - \beta(i_\beta)\right\}, \sup_{i_{\alpha^{f_1}} \in I_{\alpha^{f_1}}} \left\{\alpha^{f_1}(i_{\alpha^{f_1}}) - \beta(i_{\alpha^{f_1}} - t)\right\}\right\}$$

$$+ \sup\left\{ \sup_{i_\beta \in I_\beta} \left\{\alpha^{f_2}(t+i_\beta) - \beta(i_\beta)\right\}, \sup_{i_{\alpha^{f_2}} \in I_{\alpha^{f_2}}} \left\{\alpha^{f_2}(i_{\alpha^{f_2}}) - \beta(i_{\alpha^{f_2}} - t)\right\}\right\}$$

$$(\text{Condition } \beta = \beta_{R,T} \in \mathcal{F}_{\text{RL}})$$

$$= \sup\left\{\alpha^{f_1}(t+T), \sup_{i_{\alpha^{f_1}} \in I_{\alpha^{f_1}}} \left\{\alpha^{f_1}(i_{\alpha^{f_1}}) - \beta(i_{\alpha^{f_1}} - t)\right\}\right\}$$

$$+ \sup\left\{\alpha^{f_2}(t+T), \sup_{i_{\alpha^{f_2}} \in I_{\alpha^{f_2}}} \left\{\alpha^{f_2}(i_{\alpha^{f_2}}) - \beta(i_{\alpha^{f_2}} - t)\right\}\right\}$$

$$\left(\text{Condition } i_{\alpha^{f_i}} \leq i_{\alpha^{\mathbb{F}}}^{\max} \leq T\right)$$

$$= \sup\left\{\alpha^{f_1}(t+T), \sup_{i_{\alpha^{f_1}} \in I_{\alpha^{f_1}}} \left\{\alpha^{f_1}(i_{\alpha^{\mathbb{F}}}^{\max})\right\}\right\}$$

$$+ \sup\left\{\alpha^{f_2}(t+T), \sup_{i_{\alpha^{f_2}} \in I_{\alpha^{f_2}}} \left\{\alpha^{f_2}(i_{\alpha^{\mathbb{F}}}^{\max})\right\}\right\}$$

$$\left(\text{Condition } i_{\alpha^{\mathbb{F}}}^{\max} \leq T\right)$$

$$= \alpha^{f_1}(t+T) + \alpha^{f_2}(t+T)$$

$$(\text{Condition } \beta = \beta_{R,T} \in \mathcal{F}_{\text{RL}})$$

$$= \alpha^{f_1}(t+i_\beta) + \alpha^{f_2}(t+i_\beta)$$

$$= \left(\alpha^{f_1} + \alpha^{f_2}\right)(t+i_\beta)$$

$$= \left(\alpha^{f_1} + \alpha^{f_2}\right)(t+i_\beta) + 0$$

$$= \sup_{i_\beta}\left\{\left(\alpha^{f_1} + \alpha^{f_2}\right)(t+i_\beta) - \beta(i_\beta)\right\}$$

$$\left(\text{Condition } i_{\alpha^{\mathbb{F}}}^{\max} \leq i_\beta\right)$$

$$= \sup\left\{ \sup_{i_\beta \in I_\beta} \left\{\left(\alpha^{f_1} + \alpha^{f_2}\right)(t+i_\beta) - \beta(i_\beta)\right\}, \left(\alpha^{f_1} + \alpha^{f_2}\right)(i_{\alpha^{\mathbb{F}}}^{\max}) - \beta(i_{\alpha^{\mathbb{F}}}^{\max} - t)\right\}$$

The second round is on,
It’s getting more intense
(excerpts on the left).

Limits of the approach
are reached fast.
→ Segregation blocked

Future work:
→ Don’t clinch,
Tackle from a different angle ☺

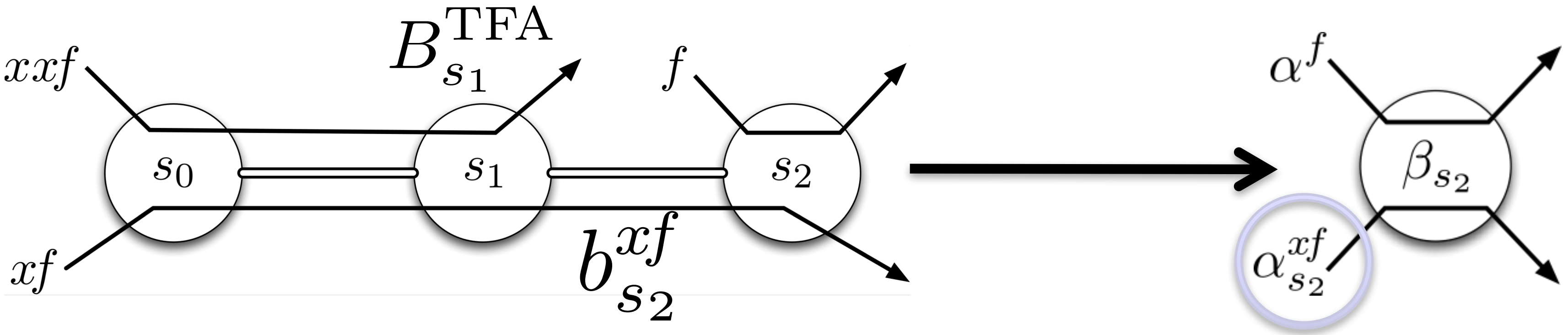
Round 3: Pay Multiplexing Only Once

- Change of tactics: Convolve before Subtraction
- Advantageous for end-to-end analysis

Result of round 3?

TBD, potentially the fight will continue …

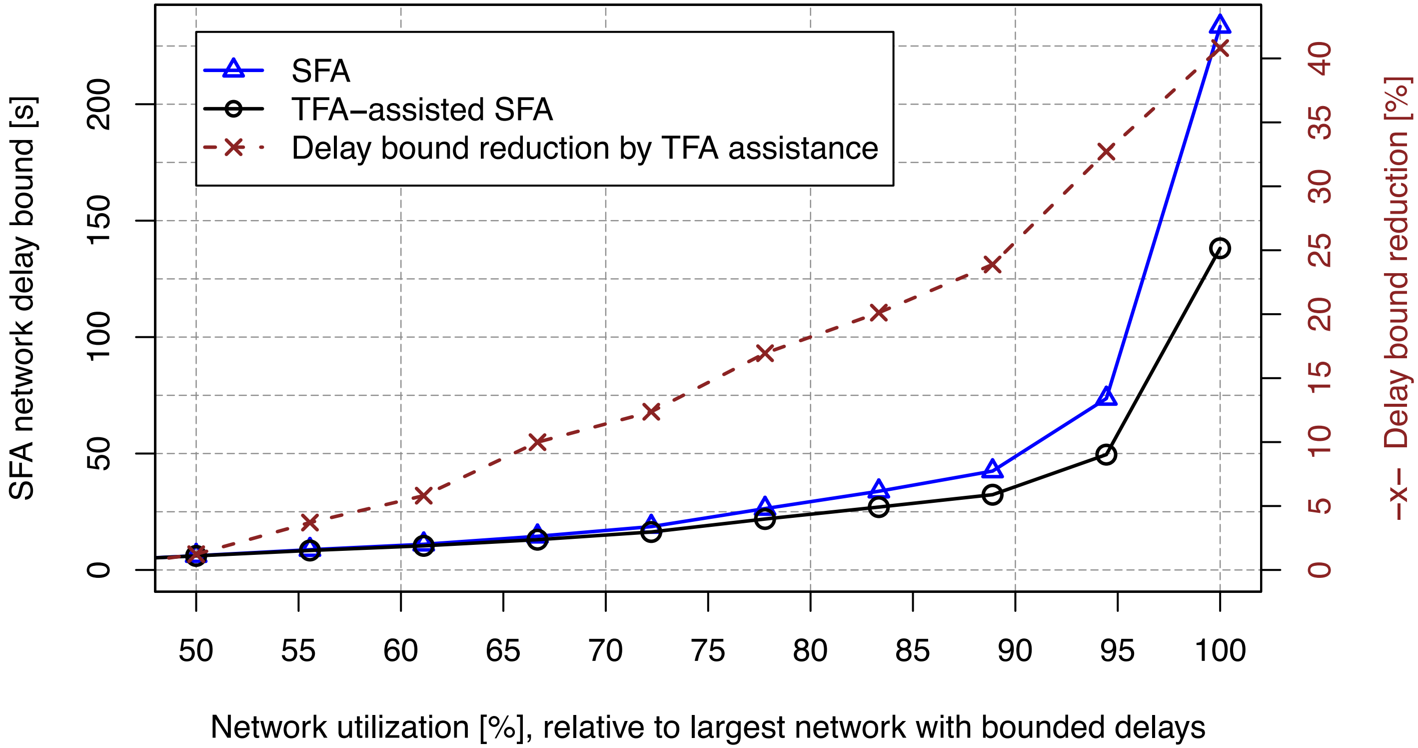
Another Boxing Ring: [MMB2016] (i.e., Yesterday)



- An unlikely place to aggregate: One hop “too early”
  Use the Total Flow Analysis (TFA) to get s_2’s backlog bound
  - It aggregates all the flows, not only the one that really interferes at s_2
  - Their backlog bound $B_{s_1}^{\mathrm{TFA}}$ can be *smaller than* the single flow’s output burstiness $b_{s_2}^{xf}$
  - Cap the burstiness of xf if it exceeds $B_{s_1}^{\mathrm{TFA}}$.
- Aggregation beats segregation … sometimes
  - This “sometimes“ happens if utilization is hight

Evaluation

- Erdös Rényi Graph
  - Hierarchy retrofitted for bottlenecks (same as before)
    - See the aSHIIP topology generator (Supélec.fr)
  - Increase amount and thus utilization



Conclusion

- Nothing is decided yet.

- Invest more effort to just do both?
- It does not scale well with the network size.

- Rule of thumb:

  Aggregate cross-traffic if you can,
  segregate cross-flows if you must.

Thank you for your attention

References

[Valuetools2014]
The DiscoDNC v2 – A Comprehensive Tool for Deterministic Network Calculus
Steffen Bondorf, Jens B. Schmitt
8th International Conference on Performance Evaluation Methodologies and Tools (Valuetools), 2014.

[Valuetools2015]
Calculating Accurate End-to-End Delay Bounds – You Better Know Your Cross-Traffic
Steffen Bondorf, Jens B. Schmitt
9th International Conference on Performance Evaluation Methodologies and Tools (Valuetools), 2015.

[INFOCOM2015]
Boosting Sensor Network Calculus by Thoroughly Bounding Cross-Traffic
Steffen Bondorf, Jens B. Schmitt
34th IEEE International Conference on Computer Communications (INFOCOM), 2015.

[MMB2016]
Improving Cross-Traffic Bounds in Feed-Forward Networks – There is a Job for Everyone
Steffen Bondorf, Jens B. Schmitt
18th International GI/ITG Conference on “Measurement, Modelling and Evaluation of Computing Systems” and “Dependability and Fault Tolerance” (MMB & DFT), 2016.