RUB

RUHR-UNIVERSITÄT BOCHUM

VIRTUAL CROSS-FLOW DETOURING IN THE DETERMINISTIC NETWORK CALCULUS

Steffen Bondorf (RUB) and Fabien Geyer (TUM | Airbus CRT)

Overview

- Bisecting the Title
  Virtual Cross-Flow Detouring in the Deterministic Network Calculus
  - Deterministic Network Calculus (DNC)
  - Importance of Bounding Cross-Flows
  - Virtually Changing Paths in the DNC Analysis

- Virtual Cross Flow Detouring
  - How and when does it work?
  - Numerical Evaluation

Motivation: Worst-Case Performance Analysis

Networks embedded into safety-critical systems need performance assurances
→ Certification

Know the worst-case performance during operation
→ Formal verification required

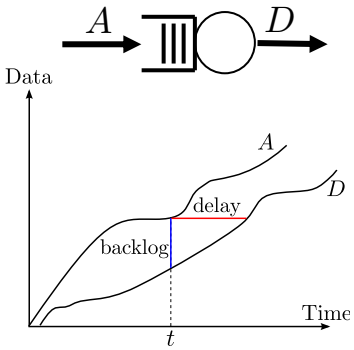
Analyze and rank different network configurations reliably

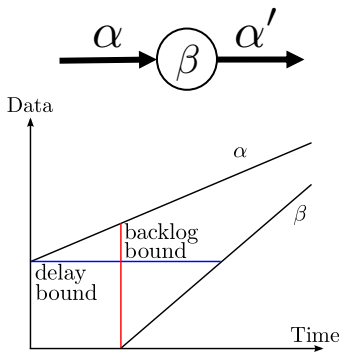
Prevent over-provisioned designs
→ Accuracy matters

Our choice: Deterministic Network Calculus (DNC)

Network Calculus Modeling

Worst-case bounds on system behavior: cumulative arrivals and service [Cruz91]





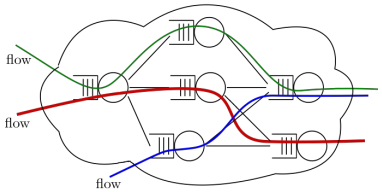
Arrival Curve α:

$$\alpha(s) \geq A(t) - A(t-s)\, \forall s \leq t$$

Strict Service Curve β:

A server is said to offer a strict service curve β if, during any backlogged period of duration u, the output of the system is at least equal to $\beta(u)$.

Network Calculus Analysis: (min,+)-Algebra



A set of (min,+)-algebraic operations [LeBoudec01]

Output bound

$$(\alpha \oslash \beta)(d) = \sup_{u \geq 0} \{\alpha(d+u) - \beta(u)\} =: \alpha'(d)$$

Left-over service curve

$$(\beta \ominus \alpha)(d) = \sup_{0 \leq u \leq d} \{(\beta - \alpha)(u)\} =: \beta^{\text{l.o.}}$$

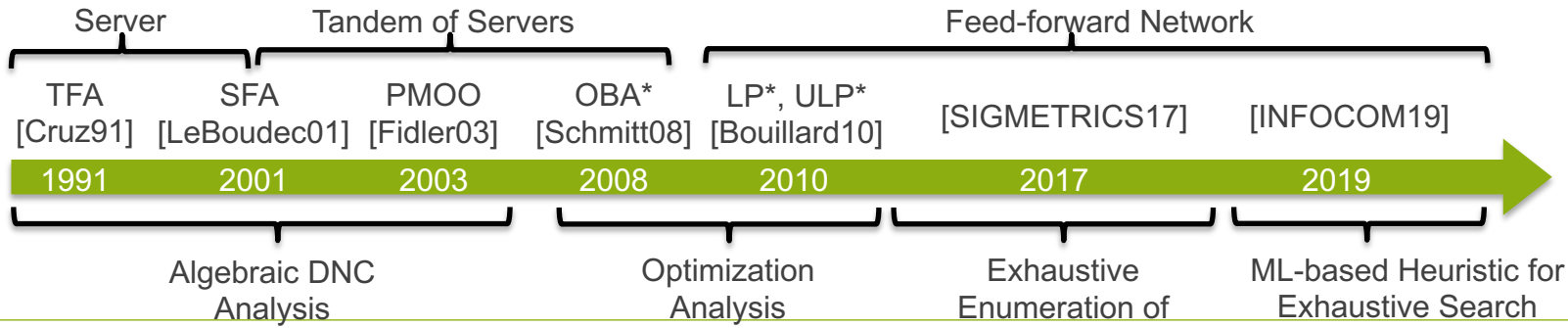
Aggregation of flows

$$(\alpha_1 + \alpha_2)(d) = \alpha_1(d) + \alpha_2(d)$$

Concatenation of servers (sequences/tandems)

$$(\beta_1 \otimes \beta_2)(d) = \inf_{0 \leq s \leq d} \{\beta_1(d-s) + \beta_2(s)\} = \beta_{\langle 1,2 \rangle}$$

A history of improvements to the analysis

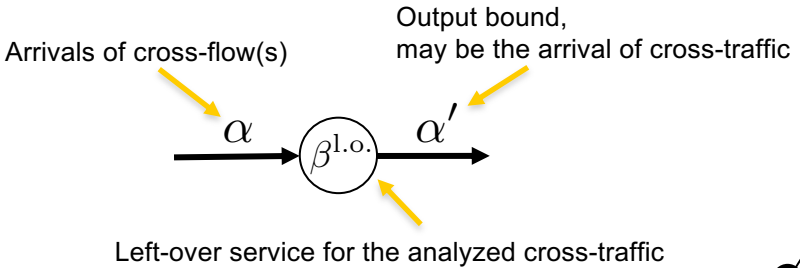


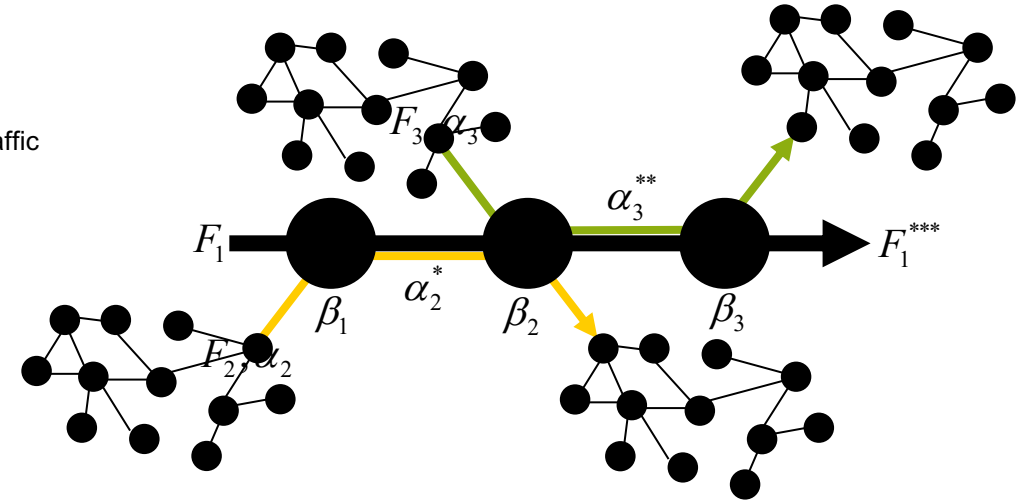
* becomes computationally infeasible

Latest Focus: Bounding Cross-traffic Arrivals [SIGMETRICS17]

Feed-forward networks are analyzed as a sequence of tandems.

Bounding the arrivals of cross-traffic arrival bounding is required at these tandems, computed as the output of a sub-network crossed before interfering with the analyzed flow.



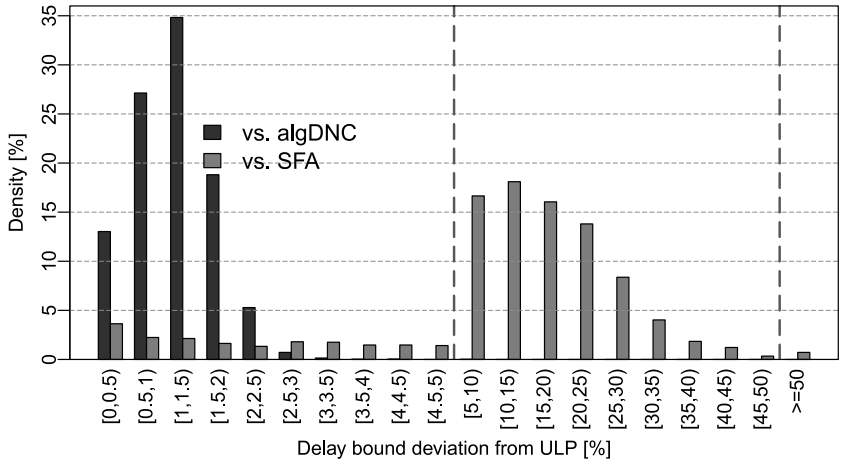


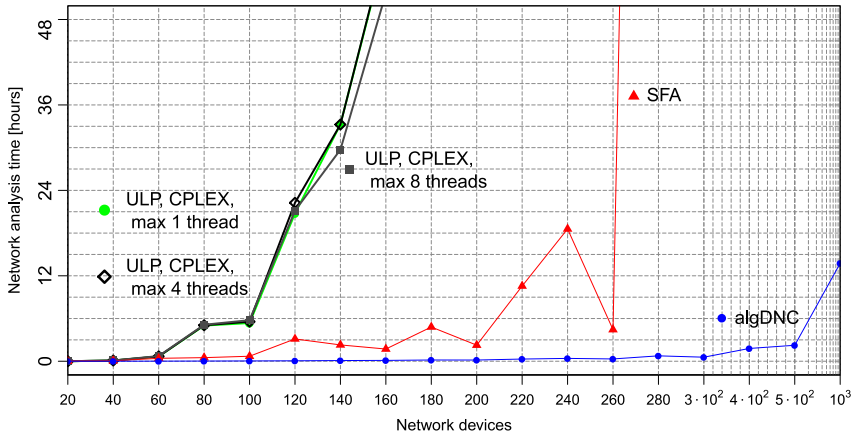
Bounding Cross-traffic Arrivals Done Right: Impact

Algebraic analysis is competitive with optimization, in quality and cost [SIGMETRICS17]

9 networks, 12376 flows

Median Delay Bound Deviation from optimization: 1.142%
99th percentile at 2.48%
Multiple orders of magnitude faster than optimization





RUHR UNIVERSITÄT BOCHUM RUB

However …
Algebraic DNC suffers from Mismatches between Modeling and Analysis Capabilities

The DNC analysis might not be able to fully consider modeled behavior.
Instead, it applies worst-case assumptions (seldom made explicit to the modeler).

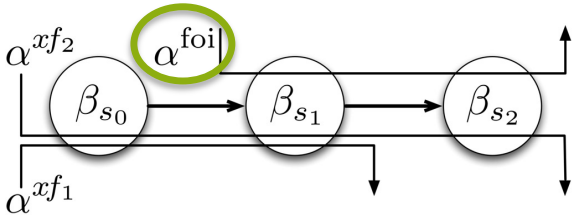
Our objective:

Find, quantify and mitigate such problems

Mismatch: Aggregate Flows, Separate Analysis (I) [ICC17]

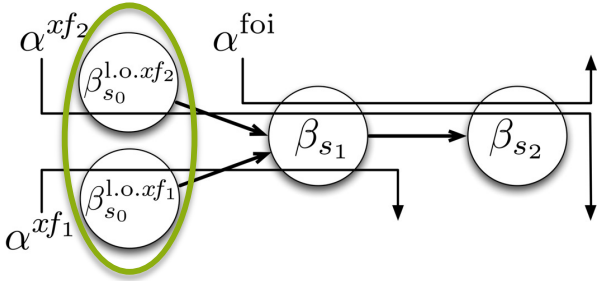
Simple Sample Scenario:

- Bound the end-to-end delay of the flow of interest (foi)
- Subject to cross-flows xf_1 and xf_2



Problem:

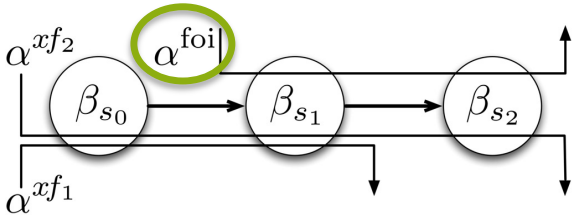
- Cross-flow entanglement on servers s_1 and s_2
- Enforces DNC to separately bound their arrivals at s_1
- DNC analysis proceeding:
  - Explicitly assignment priorities to establish the worst case for each of the two cross-flows
  - Simultaneously assume $xf_1 \prec xf_2$ and $xf_2 \prec xf_1$
- Mutually exclusive left-over β operations
- A realistic system cannot behave like this!
- Overly pessimistic analysis!



Mismatch: Aggregate Flows, Separate Analysis (II) [ICC17]

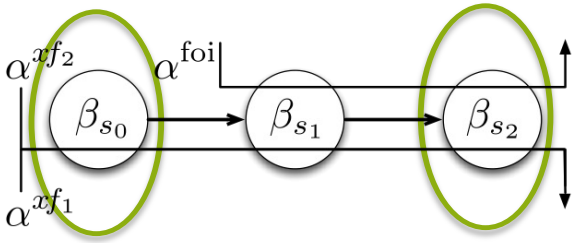
Simple Sample Scenario:

- Bound the end-to-end delay of the flow of interest (foi)
- Subject to cross-flows xf_1 and xf_2



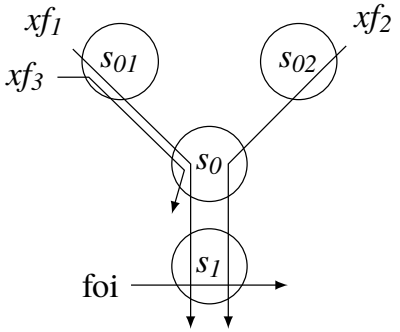
Mitigation: Virtual Flow Prolongation at the End

- Change cross-flow entanglement on servers s_1 and s_2
- Allows DNC to aggregately bound their arrivals at s_1
- DNC analysis proceeding:
  No explicitly assignment priorities required
  for the single cross-flow aggregate
- No mutually exclusive left-over β
- But: Additional interference at s_2
- **It's safe replace the original model**
- It's a tradeoff



It's changing flow paths,
yet only in the analysis making it virtual!

Virtual Cross-Flow Detouring



Simple Sample Scenario:

- Bound the end-to-end delay of the flow of interest (foi)
- Subject to cross-flows xf_1 and xf_2

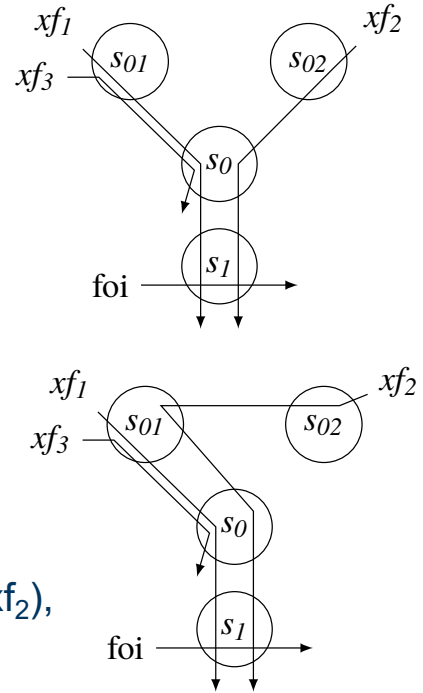
Problem:

- There is another flow, xf_3, that interferes with xf_1
- xf_1 and xf_3 both cross the server tandem s_{01} and s_0
- xf_1 and xf_2 aggregately interfere with the foi at s_1
- The recursive DNC analysis proceeding starts at the foi:
  - xf_1 and xf_2 are bounded in aggregate at s_0,
    i.e., s_0 must be analyzed in isolation
  - Then, xf_1 and xf_3 cannot be analyzed on the entire tandem s_{01} and s_0
  - Instead, the analysis assumes worst-case bustiness of xf_3 at s_{01} and at s_0
- Overly pessimistic analysis!

Virtual Cross-Flow Detouring

Simple Sample Scenario:

- Bound the end-to-end delay of the flow of interest (foi)
- Subject to cross-flows xf_1 and xf_2



Mitigation: Virtual Cross-Flow Detouring

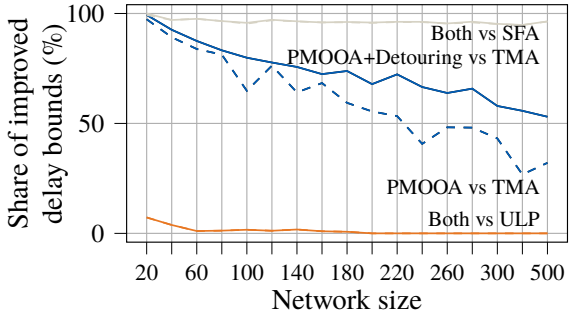
- Assume (in the analysis only!) xf_2 crosses s_{01}, too
- Entirely different interference pattern that matches analysis capabilities
  → DNC can compute a better bound than before
- But is this virtual model transformation really more pessimistic?
  - Added pessimism is very tightly coupled to the PMOO analysis!
  - It cannot make use of the potential positive changes [Schmitt08].
    The location of interference might reduce the load at s_0 (less bursty xf_2), yet, not in the PMOO analysis.
- A generalization of flow prolongation at the end
- Many potential detouring alternatives → a heuristic *PMOOA+Detouring* is in the paper

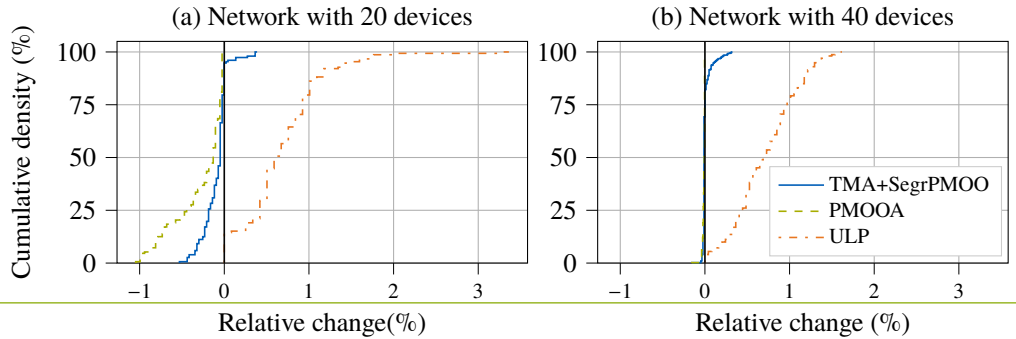
Numerical Evaluation

Same networks as before, main competitors:

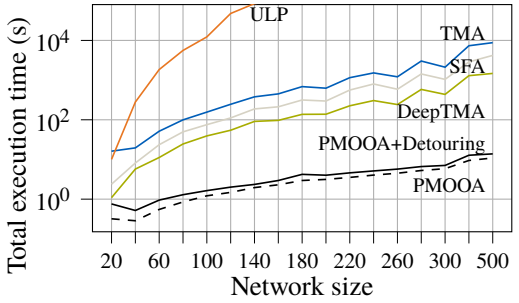
- PMOO Analysis without Detouring (PMOOA)
- exhaustive algebraic analysis (TMA)
  - TMA with bound tightening SegrPMOO addition
  - ML-augmented TMA (DeepTMA)
  - Optimization-based analysis (ULP)

Competitive or superior Delay Bounds vs. TMA





Superior runtime performance



Conclusion

- Determinsitic Network Calculus is an advanced tool for performance modeling and analysis but its generic feed-forward analysis still has some problems
- We uncovered and mitigated one of these problems with virtual cross-flow detouring
- We were able to design a simple heuristic that can compute comeptitive delay bounds at a fraction of the runtime of other analyses

References

[SIGMETRICS17] S. Bondorf, P. Nikolaus, and J. Schmitt. *Quality and Cost of Deterministic Network Calculus – Design and Evaluation of an Accurate and Fast Analysis*. In Proc. of ACM SIGMETRICS 2017; full paper in Proceedings of the ACM on Measurement and Analysis of Computing Systems (POMACS), volume 1, 2017.

[Bouillard10] A. Bouillard, L. Jouhet, and E. Thierry. *Tight Performance Bounds in the Worst- Case Analysis of Feed-Forward Networks*. In Proc. of IEEE INFOCOM, 2010.

[Cruz91] R. L. Cruz. *A Calculus for Network Delay, Part I: Network Elements in Isolation.* In IEEE Transactions on Information Theory, 1991, and R. L. Cruz. *A Calculus for Network Delay, Part II: Network Analysis.* In IEEE Transactions on Information Theory, 1991.

[Fidler03] M. Fidler. *Extending the Network Calculus Pay Bursts Only Once Principle to Aggregate Scheduling.* In Proc. of the International Workshop on Quality of Service in Multiservice IP Networks (QoS-IP), 2003.

[LeBoudec01] J.-Y. Le Boudec and P. Thiran. Network Calculus: A Theory of Deterministic Queuing Systems for the Internet. Springer, 2001.

[Schmitt08] J. Schmitt, F. A. Zdarsky, and M. Fidler. *Delay Bounds under Arbitrary Multiplexing: When Network Calculus Leaves You in the Lurch ...* In Proc. of IEEE INFOCOM, 2008.

[INFOCOM19] F. Geyer and S. Bondorf. *DeepTMA: Predicting Effective Contention Models for Network Calculus using Graph Neural Networks.* In Proc. of INFOCOM, 2019.

[ICC17] S. Bondorf. *Better Bounds by Worse Assumptions – Improving Network Calculus Accuracy by Adding Pessimism to the Network Model.* In Proc. of the IEEE International Conference on Communications (ICC), 2017.