On the Stochastic End-to-End Delay Analysis in Sink Trees Under Independent and Dependent Arrivals

Paul Nikolaus and Jens Schmitt

Distributed Computer Systems (DISCO) Lab, TU Kaiserslautern,
Kaiserslautern, Germany
{nikolaus,jschmitt}@cs.uni-kl.de

Abstract. Sink trees are a frequent topology in many networked systems; typical examples are multipoint-to-point label switched paths in Multiprotocol Label Switching networks or wireless sensor networks with sensor nodes reporting to a base station. In this paper, we compute end-to-end delay bounds using a stochastic network calculus approach for a flow traversing a sink tree.
For n servers with one flow of interest and n cross-flows, we derive solutions for a general class of arrivals with moment-generating function bounds. Comparing algorithms known from the literature, our results show that, e.g., pay multiplexing only once has to consider less stochastic dependencies in the analysis.
In numerical experiments, we observe that the reduced dependencies to consider, and therefore less applications of Hölder's inequality, lead to a significant improvement of delay bounds with fractional Brownian motion as a traffic model. Finally, we also consider a sink tree with dependent cross-flows and evaluate the impact on the delay bounds.

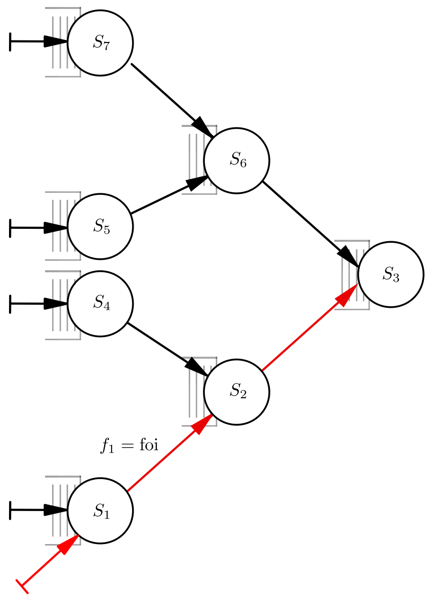
Keywords: Network calculus · Sink trees · Moment-generating functions · Hölder's inequality · Fractional Brownian motion.

1 Introduction

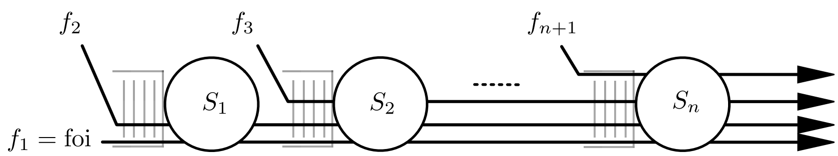
1.1 Background

The stochastic network calculus (SNC) offers a versatile uniform framework to compute probabilistic performance bounds in networked systems. The most prominent goal is to control tail probabilities for the end-to-end (e2e) delay, i.e., probabilities for rare events shall be bounded, e.g., $\mathrm{P}(\text{e2e delay} > 10\text{ms}) \leq 10^{-6}$. Many modern systems are eager after such performance guarantees, as exemplified in application visions like, e.g., Tactile Internet [17], Industrial IoT [8], or Internet at the speed-of-light [43].

Over almost three decades, the development of SNC has progressed with the pioneering work by [11, 15, 45], and important contributions in [9, 13, 18, 23, 28] to name a few, see also [14, 19] for a guide and some perspectives. Two flavors



(a) Sink tree.



(b) Sink tree tandem after reduction with n servers and $n+1$ flows.

Fig. 1. Sink tree reduction.

of SNC have evolved: arrival and service processes characterized either by tail bounds [13, 15, 23] or by moment-generating function (MGF) bounds [12, 18]. While tail bounds offer a wider modeling scope, they have been shown to result in more conservative bounds in scenarios where independence between stochastic processes can be assumed [37]. In this paper, we focus on the SNC with MGF bounds.

1.2 E2E Analysis in SNC – State of Affairs

When performing an SNC analysis given a network of servers with a set of flows routed over subsets of theses servers, the first step is to reduce the network to the tandem of servers which are traversed by a certain flow of interest (foi), see also Figure 1. To that end, arrival bounds for each cross-flow with which the foi shares a subset of servers need to be computed at the point when it joins the foi. Conceptually, this simple, but important step just requires the computation

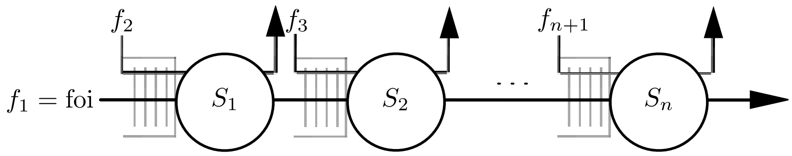


Fig. 2. Tandem in [18].

of output bounds for the cross-flows and has recently been improved by the use of Lyapunov's inequality in [34].

While, in this paper, we assume this reduction from network to tandem has already been performed, an important observation has to be made: Even under the assumption that all flows are independent when entering the network, some of them become dependent when traversing a shared server. These stochastic dependencies between cross-flows severely aggravate the e2e analysis for the foi. In the SNC based on MGF bounds, one can resort to use Hölder's Inequality to deal with dependent scenarios. However, the inequality is known to be very conservative and, if invoked too often in the e2e analysis, loosens the performance bounds considerably. This was shown in a simple tandem of servers traversed by many parallel flows in [32]. In that paper, it was demonstrated that two different ways of doing the e2e delay analysis, known from the deterministic network calculus (DNC) as separated flow analysis (SFA) and pay multiplexing only once (PMOO), result in very different bounds. In these simple tandems, PMOO completely avoids the usage of Hölder's inequality if all flows are originally independent whereas SFA requires the inequality's invocation at each server; consequently, PMOO clearly outperforms SFA in the quality of the bounds (and also in run times). This shows the importance of a careful e2e analysis and has been investigated extensively in DNC literature, see, e.g., [5, 7, 41]. Yet, to the best of our knowledge, it has not been investigated much in the literature on SNC with MGF bounds.

In existing work on the SNC based on MGF bounds [18], a tandem as in Figure 2 was analyzed with all flows being independent and a nice linear scaling of the e2e delay bound was shown. Yet, such a tandem, if being the result of a network reduction, is likely to be crossed by dependent flows based on their previous entanglement. Apart from the work mentioned above [32], there is also some work to deal with stochastic dependencies in the analysis [6,16,33]. Overall, it must be concluded that the e2e analysis in SNC still faces many open problems.

1.3 Motivation and Contribution

In this paper, we take the next step in tackling the SNC e2e delay analysis by providing results for a particular topology: sink trees. Sink trees are interesting for a number of application scenarios:

- Classically, in Multiprotocol Label Switching (MPLS) networks there is the the option to set up multipoint-to-point label-switched paths between several ingress edge routers and one egress edge router [39], thus creating a sink tree.
- Multi-hop wireless sensor networks with a central base station collecting data from sensor nodes induce a sink tree topology and have been investigated using network calculus methods previously, see, e.g., [25,40]. More generally, any data collection by a central point results in a sink tree and if time-critical decisions are made based on that data, performance guarantees are desirable, see, e.g., [47].
- Network-on-Chip architectures frequently employ tree topologies and have been analyzed using network calculus methods previously, see, e.g., [21,36].
- Switched Ethernets set up spanning trees to avoid cycles in frame forwarding, hence, again sink trees emerge as a natural choice to support resource allocation in such installations [22].
- Sink trees are also related to so-called fat trees in supercomputing [26]; in fact, fat trees have also been proposed in data center interconnects and have been subject to SNC-based analysis in [44,46], recently.

Hence, from an application perspective, it is clearly interesting to provide an e2e delay analysis for sink trees. While we constrain the topology for the SNC e2e analysis to sink trees, we want to remain as flexible as possible with respect to arrival and service models. In particular, we intentionally do not restrict to linear MGF bounds as provided by the so-called (σ, ρ)-bounds from [12], but derive the e2e delay bounds for general MGF bounds on arrivals and service. For instance, this includes a traffic model based on fractional Brownian motion (fBm). FBm has been shown to be useful for Internet traffic modeling [20,35], because it can capture the typical long-range dependence, which is why we also use it in our numerical experiments. On the other hand, it is a non-trivial traffic type for SNC to deal with and we do not provide stationary (time-independent) delay bounds (for Hurst parameter $H > 0.5$), but transient (time-dependent) delay bounds only. Having said that, it is interesting to note that some applications are actually more interested in transient bounds and corresponding developments have been reported in [2,4,10,29].

1.4 Outline

In Section 2, we provide the necessary SNC background and notations used throughout the paper. Section 3 presents the derivations for the e2e delay analysis of sink tree tandems under independent and dependent cross-flows using different algorithms (SFA and PMOO). In Section 4, numerical evaluations of different aspects are provided: influence of the time horizon on the transient delay bounds, effects of traffic parameters and sink tree depths, comparisons between different analysis algorithms and the independent and dependent scenarios. Section 5 concludes the paper.

2 SNC Background and Notation

We use the MGF-based SNC in order to bound the probability that the delay exceeds a given value $T \geq 0$. The MGF bound on a probability is established by applying Chernoff's bound [31]

$$\mathrm{P}(X > a) \leq e^{-\theta a}\,\mathrm{E}\big[e^{\theta X}\big]\,, \quad \theta > 0.$$

Definition 1 (Arrival Process). *We define an* arrival flow *by the stochastic process A with discrete time space $\mathbb{N}_0$ and continuous state space $\mathbb{R}_0^+$ as*

$$A(s,t) \coloneqq \sum_{i=s+1}^{t} a_i, \tag{1}$$

with $a_i \geq 0$ as the traffic increment process in time slot i.

Network calculus provides an elegant system-theoretic analysis by employing min-plus algebra.

Definition 2 (Convolution and Deconvolution in Min-Plus Algebra [1]). *Let $x(s,t)$ and $y(s,t)$ be real-valued, bivariate functions. The* min-plus convolution *of x and y is defined as*

$$x \otimes y\,(s,t) \coloneqq \inf_{s \leq u \leq t} \{x(s,u) + y(u,t)\}\,.$$

The min-plus deconvolution *of x and y is defined as*

$$x \oslash y\,(s,t) \coloneqq \sup_{0 \leq u \leq s} \{x(u,t) - y(u,s)\}\,.$$

The characteristics of the service process are captured by the notion of a dynamic S-server.

Definition 3 (Dynamic S-Server [12]). *Assume a service element has an arrival flow A as its input and the respective output is denoted by D. Let $S(s,t)$, $0 \leq s \leq t$, be a stochastic process that is nonnegative and increasing in t. The service element is a* dynamic S-server *iff for all $t \geq 0$ it holds that*

$$D(0,t) \geq A \otimes S\,(0,t) = \inf_{0 \leq s \leq t} \{A(0,s) + S(s,t)\}\,.$$

Definition 4 (Work-Conserving Server [12] [18]). *For any $t \geq 0$ let $\tau \coloneqq \sup\{s \in [0,t] : D(0,s) = A(0,s)\}$ be the beginning of the last backlogged period before t. Assume again the service $S(s,t)$, $0 \leq s \leq t$, to be a stochastic process that is nonnegative and increasing in t with $S(\tau,\tau) = 0$. A server is said to be* work-conserving *if for any fixed sample path the server is non-idling in $(\tau, t]$ and uses the entire available service, i.e., $D(0,t) = D(0,\tau) + S(\tau,t)$.*

The analysis is based on a per-flow perspective. That is, we consider a certain flow, the so-called *flow of interest* (foi). Throughout this paper, for the sake of simplicity, we assume the servers' scheduling to be arbitrary multiplexing [41].

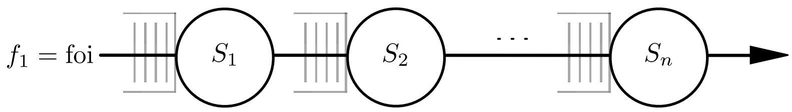


Fig. 3. Tandem of n servers.

Proposition 1 (Leftover Service under Arbitrary Multiplexing [18]). *Consider two arrivals flows f_1 and f_2 at a work-conserving dynamic S-server with service process S. Then, the corresponding arrival A_1 sees under arbitrary multiplexing the* leftover service

$$S_{\text{l.o.}}(s,t) = [S(s,t) - A_2(s,t)]^+ .$$

Definition 5 (Virtual Delay). *The* virtual delay *at time $t \geq 0$ is defined as*

$$d(t) \coloneqq \inf\{\tau \geq 0 : A(0,t) \leq D(0,t+\tau)\} .$$

It can briefly be described as the time it takes for the cumulated departures to "catch up with" the cumulated arrivals.

Theorem 1 (Output and Delay Bound [12] [18]). *Consider an arrival process $A(s,t)$ with dynamic S-server $S(s,t)$.*

The departure process D is upper bounded for any $0 \leq s \leq t$ according to

$$D(s,t) \leq A \oslash S\,(s,t). \tag{2}$$

The delay at $t \geq 0$ is upper bounded by

$$d(t) \leq \inf\{\tau \geq 0 : A \oslash S\,(t+\tau,t) \leq 0\} .$$

We focus on the stochastic analogue of Theorem 1 for moment-generating functions:

Theorem 2 (MGF Delay Bound [18] [3]). *For the assumptions as in Theorem 1, we obtain:*

The violation probability of a given stochastic delay bound $T \geq 0$ at time $t \geq 0$ is bounded by

$$\mathrm{P}(d(t) > T) \leq \mathrm{E}\left[e^{\theta(A \oslash S\,(t+T,t))}\right], \quad \forall \theta > 0. \tag{3}$$

In order to obtain the tightest possible result, the bound in Equation (3) should be optimized in θ.

The next theorem shows how network calculus leverages min-plus algebra to derive end-to-end results.

Theorem 3 (End-to-End Service [18]). *Consider a flow f crossing a tandem of n work-conserving servers with service processes $S_i, i = 1, \ldots, n$ as in Figure 3. Then, the overall service offered to f can be described by the* end-to-end service

$$S_{\text{e2e}}(s,t) = \bigotimes_{i=1}^{n} S_i(s,t) \coloneqq S_1 \otimes S_2 \otimes \cdots \otimes S_n\,(s,t).$$

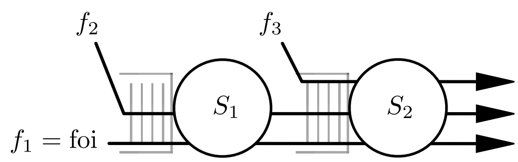


Fig. 4. Sink tree with 3 flows and 2 servers.

In the following definition, we introduce (σ, ρ)-constraints [12] as they are often used to compute time-independent, stationary bounds under stability.

Definition 6 **((σ, ρ)-Bound [12]).** *An arrival flow is $(\sigma_A(\theta), \rho_A(\theta))$-bounded for some $\theta > 0$, if for all $0 \leq s \leq t$*

$$\mathrm{E}\left[e^{\theta A(s,t)}\right] \leq e^{\theta \rho_A(\theta)(t-s)+\theta \sigma_A(\theta)}.$$

Theorem 4 (Generalized Hölder Inequality [30]). *Let $X_1, \dots, X_n \geq 0$ be random variables such that $\mathrm{E}[X_i^{p_i}] < \infty$. Then we have*

$$\mathrm{E}\left[\prod_{i=1}^{n} X_i\right] \leq \prod_{i=1}^{n} \mathrm{E}[X_i^{p_i}]^{\frac{1}{p_i}}$$

with $\sum_{i=1}^{n} \frac{1}{p_i} = 1$ and $p_i > 1$.

3 Sink Tree End-to-End Delay Bound

In this section, we provide stochastic delay bounds for sink trees for the separated flow analysis (SFA) and pay multiplexing only once (PMOO) e2e analysis algorithms as known from DNC [42]. All topologies in this paper assume the servers to be work-conserving and independent of the arrivals. We start the analysis with independent cross-flows, but forego this assumption at the end of the section.

3.1 Two-Server Sink Tree

We start the sink-tree analysis with the two-server case (Figure 4) as an illustrative example, since it already enables us to point at some key differences between SFA and PMOO. We extend the results to general sink trees in the following subsection.

Separated Flow Analysis (SFA) Here, we compute the leftover service at each server (assuming arbitrary multiplexing) until we convolve all service processes in a final step.

For the two-server sink tree in Figure 4, SFA yields the end-to-end service

$$S_{\mathrm{e2e}} = [S_1 - A_2]^+ \otimes [S_2 - (A_3 + (A_2 \oslash S_1))]^+ . \tag{4}$$

Observe that the arrival process A_2 appears twice. For the analysis, this means that we need to invoke Hölder's inequality to upper bound the MGF of dependent processes. It follows for the delay bound, that

$$\begin{aligned}
&\mathrm{P}(d(t) > T) \\
&\overset{(3)}{\leq} \mathrm{E}\left[e^{\theta(A_1 \oslash S_{\mathrm{e2e}}\,(t+T,t))}\right] \\
&\overset{(4)}{=} \mathrm{E}\left[e^{\theta\left(A_1 \oslash [S_1 - A_2]^+ \otimes [S_2 - (A_3 + (A_2 \oslash S_1))]^+\,(t+T,t)\right)}\right] \\
&\ \vdots \\
&\leq \sum_{s_0=0}^{t} \mathrm{E}\left[e^{\theta A_1(s_0,t)}\right] \\
&\quad \cdot \left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\left[e^{p_1 \theta A_2(s_0,s_1)}\right]^{\frac{1}{p_1}} \mathrm{E}\left[e^{e^{-p_1\theta S_1(s_0,s_1)}}\right]^{\frac{1}{p_1}} \mathrm{E}\left[e^{p_2 \theta A_3(s_1,t+T)}\right]^{\frac{1}{p_2}} \right. \\
&\quad \left. \cdot \left(\sum_{s_2=0}^{s_1} \mathrm{E}\left[e^{p_2 \theta A_2(s_2,t+T)}\right] \mathrm{E}\left[e^{-p_2\theta S_1(s_2,s_1)}\right] \right)^{\frac{1}{p_2}} \mathrm{E}\left[e^{-p_2 \theta S_2(s_1,t+T)}\right]^{\frac{1}{p_2}} \right),
\end{aligned}$$

where $1/p_1 + 1/p_2 = 1$.

Pay Multiplexing Only Once (PMOO) In contrast to SFA, in PMOO, we first convolve and then subtract. However, we only obtain a rigorous bound if we convolve servers that share the same set of cross-flows. Therefore, one has to first subtract all flows that are not in this intersection of cross-flows. For sink trees, there is still a unique outcome when applying the PMOO algorithm, since there is no overlapping interference. The analysis can become much more complex when considering general topologies [42].

It is known in deterministic network calculus, that neither of the analyses is strictly better than the other [41], though for many topologies PMOO yields a better delay bound [42].

For the two-server sink tree, PMOO yields the end-to-end service

$$S_{\mathrm{e2e}} = \left[\left([S_2 - A_3]^+ \otimes S_1\right) - A_2\right]^+. \tag{5}$$

In contrast to SFA, A_2 appears only once.

$$\begin{aligned}
&\mathrm{P}(d(t) > T) \\
&\overset{(3)}{\leq} \mathrm{E}\left[e^{\theta(A_1 \oslash S_{\mathrm{e2e}}\,(t+T,t))}\right] \\
&\overset{(5)}{=} \mathrm{E}\left[e^{\theta\left(A_1 \oslash \left[\left(S_1 \otimes [S_2 - A_3]^+\right) - A_2\right]^+\,(t+T,t)\right)}\right]
\end{aligned}$$

$$\leq \sum_{s_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(s_0,t)}\Big]\,\mathrm{E}\Big[e^{-\theta\big[\big(S_1\otimes[S_2-A_3]^+\big)-A_2\big]^+(s_0,t+T)}\Big]$$

$$\leq \sum_{s_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(s_0,t)}\Big]\,\mathrm{E}\Big[e^{\theta A_2(s_0,t+T)}\Big]\,\mathrm{E}\Big[e^{-\theta\big(\big(S_1\otimes[S_2-A_3]^+\big)\big)(s_0,t+T)}\Big]$$

$$\leq \sum_{s_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(s_0,t)}\Big]\,\mathrm{E}\Big[e^{\theta A_2(s_0,t+T)}\Big]\left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\Big[e^{-\theta S_1(s_1,t+T)}\Big]\cdot \mathrm{E}\Big[e^{-\theta[S_2-A_3]^+(s_0,s_1)}\Big]\right)$$

$$\leq \sum_{s_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(s_0,t)}\Big]\,\mathrm{E}\Big[e^{\theta A_2(s_0,t+T)}\Big]\left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\Big[e^{-\theta S_1(s_1,t+T)}\Big]\,\mathrm{E}\Big[e^{\theta A_3(s_0,s_1)}\Big]\right.$$

$$\left.\cdot\,\mathrm{E}\Big[e^{-\theta S_2(s_0,s_1)}\Big]\right).$$

Even though we consider only a two-server sink tree, we can already observe the key difference between SFA and PMOO, as only the SFA has to apply Hölder's inequality. We see in the following subsection, that this insight is even more evident in the general sink tree.

3.2 The General Case

In this subsection, we generalize the two-server sink tree to the sink tree with n servers, as in Figure 1b. The proof follows lines similar to the one of Proposition 3 and is therefore omitted.

Proposition 2 (Delay Bound with SFA). *With the SFA, the end-to-end service for $n+1$ arrival flows and n servers in a sink tree is*

$$\begin{aligned} S_{\mathrm{e2e}} = {} & [S_1 - A_2]^+ \otimes [S_2 - (A_3 + (A_2 \oslash S_1))]^+ \\ & \cdots \otimes \Big[S_n - \Big(A_n + (A_{n-1}\oslash S_{n-1}) + \cdots + \Big((A_1 \oslash S_1)\oslash [S_2 - A_2]^+\Big)\oslash \\ & \cdots \oslash [S_{n-1} - (A_2 + \cdots + A_{n-1})]^+\Big)\Big]^+ . \end{aligned} \tag{6}$$

This yields for the delay bound

$$\begin{aligned} & \mathrm{P}(d(t) > T) \\ \overset{(6)}{\leq} & \sum_{s_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(s,t)}\Big]\left(\sum_{s_1=s_0}^{t+T}\cdots\sum_{s_{n-1}=s_{n-2}}^{t+T} \mathrm{E}\Big[e^{p_1\theta A_2(s_0,s_1)}\Big]^{\frac{1}{p_1}}\,\mathrm{E}\Big[e^{-p_1\theta S_1(s_0,s_1)}\Big]^{\frac{1}{p_1}}\right. \\ & \cdots \mathrm{E}\Big[e^{p_n\theta\big(A_{n+1}+(A_n\oslash S_{n-1})+\cdots+\big(\big((A_2\oslash S_1)\oslash[S_2-A_3]^+\big)\oslash\cdots\oslash[S_{n-1}-(A_3+\cdots+A_n)]^+\big)\big)(s_{n-1},t+T)}\Big]^{\frac{1}{p_n}} \\ & \left.\cdot\,\mathrm{E}\Big[e^{-p_n\theta S_n(s_{n-1},t+T)}\Big]^{\frac{1}{p_n}}\right) \end{aligned}$$

with $\sum_{i=1}^{n} \frac{1}{p_i} = 1$.

The PMOO, on the other hand, does not have to take into account the dependencies between cross-flows that share servers.

Proposition 3 (Delay Bound with PMOO). *With the PMOO, the end-to-end service for $n+1$ arrival flows and n servers in a sink tree is*

$$S_{\mathrm{e2e}} = \left[\left(\left[\left([S_n - A_{n+1}]^+ \otimes S_{n-1}\right) - A_n\right]^+ \otimes \cdots \otimes S_1\right) - A_2\right]^+ . \tag{7}$$

This yields for the delay bound

$$\begin{aligned}
&\mathrm{P}(d(t) > T) \\
&\overset{(7)}{\leq} \sum_{s_0=0}^{t} \mathrm{E}\left[e^{\theta A_1(s_0,t)}\right] \mathrm{E}\left[e^{\theta A_2(s_0,t+T)}\right] \left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\left[e^{-\theta S_1(s_1,t+T)}\right] \mathrm{E}\left[e^{\theta A_3(s_0,s_1)}\right] \right. \\
&\quad \cdot \left(\sum_{s_2=s_0}^{s_1} \mathrm{E}\left[e^{-\theta S_2(s_2,s_1)}\right] \mathrm{E}\left[e^{\theta A_4(s_0,s_2)}\right] \cdots \left(\sum_{s_k=s_0}^{s_{k-1}} \mathrm{E}\left[e^{-\theta S_k(s_k,s_{k-1})}\right] \mathrm{E}\left[e^{\theta A_{k+2}(s_0,s_k)}\right] \right.\right. \\
&\quad \left.\left.\left. \cdots \left(\sum_{s_{n-1}=s_0}^{s_{n-2}} \mathrm{E}\left[e^{-\theta S_{n-1}(s_{n-1},s_{n-2})}\right] \mathrm{E}\left[e^{\theta A_{n+1}(s_0,s_{n-1})}\right] \mathrm{E}\left[e^{-\theta S_n(s_0,s_{n-1})}\right] \right)\right)\right)\right) .
\end{aligned}$$

Proof. See Appendix A.1.

3.3 Delay Bounds with PMOO under Dependent Cross-Flows

So far, the analysis only considered originally independent arrival flows. Now, if we assume the cross-flow arrivals to be dependent, even with the PMOO, we have to apply Hölder's inequality. Such dependencies may be due to resource sharing between cross-flows before they hit the foi, or simply because the original data sources are already dependent, as, e.g., in an environmental sensor network where the range of sensor nodes is overlapping and, thus, an observed physical phenomenon is reported by several neighboring nodes at the same time.

Proposition 4 (Delay Bound with PMOO and Dependent Cross-Flows). *If all n cross-flows are dependent, the PMOO yields*

$$\begin{aligned}
&\mathrm{P}(d(t) > T) \\
&\leq \sum_{s_0=0}^{t} \mathrm{E}\left[e^{\theta A_1(s,t)}\right] \left(\mathrm{E}\left[e^{p_1 \theta A_2(s,t+T)}\right]\right)^{\frac{1}{p_1}} \\
&\quad \cdot \left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\left[e^{-p_2 \theta S_1(s_1,t+T)}\right] \left(\mathrm{E}\left[e^{p_2 p_3 \theta A_3(s_0,s_1)}\right]\right)^{\frac{1}{p_3}} \right.
\end{aligned}$$

$$\cdots \left(\sum_{s_{k-1}=s_0}^{s_{k-2}} \mathrm{E}\left[e^{-p_2 p_4 \cdots p_{2k-2} \theta S_{k-1}(s_{k-1}, s_{k-2})} \right] \left(\mathrm{E}\left[e^{p_2 p_4 \cdots p_{2k-2} p_{2k-1} \theta A_{k+1}(s_0, s_{k-1})} \right] \right)^{\frac{1}{p_{2k-1}}} \right.$$

$$\cdots \left(\sum_{s_{n-2}=s_0}^{s_{n-3}} \mathrm{E}\left[e^{-p_2 p_4 \cdots p_{2n-4} \theta S_{n-2}(s_{n-2}, s_{n-3})} \right] \right.$$

$$\cdot \left(\mathrm{E}\left[e^{p_2 p_4 \cdots p_{2n-4} p_{2n-3} \theta A_n(s_0, s_{n-1})} \right] \right)^{\frac{1}{p_{2n-3}}}$$

$$\cdot \left(\sum_{s_{n-1}=s_0}^{s_{n-2}} \mathrm{E}\left[e^{-p_2 p_4 \cdots p_{2n-2} \theta S_{n-1}(s_{n-1}, s_{n-2})} \right] \mathrm{E}\left[e^{p_2 p_4 \cdots p_{2n-2} \theta A_{n+1}(s_0, s_{n-1})} \right] \right.$$

$$\left. \left. \left. \left. \cdot\, \mathrm{E}\left[e^{-p_2 p_4 \cdots p_{2n-2} \theta S_n(s_0, s_{n-1})} \right] \right)^{\frac{1}{p_{2n-2}}} \cdots \right)^{\frac{1}{p_{2k-2}}} \right)^{\frac{1}{p_4}} \right)^{\frac{1}{p_2}},$$

where $\frac{1}{p_1} + \frac{1}{p_2} = 1, \; \ldots, \; \frac{1}{p_{2n-1}} + \frac{1}{p_{2n-2}} = 1.$

4 Numerical Evaluation

In this section, we evaluate and compare the delay bounds in sink trees for different algorithms and parameters. At the beginning, all flows are assumed to be independent. At first, we investigate the impact of the transient time horizon t on the bound and how it relates to the assumed fractional Brownian motion traffic model. Then, we compare the SFA with the PMOO, before taking a look at the sensitivity of the model with respect to the fBm parameters. Furthermore, we consider the scaling behavior when increasing the tree depth. In the last experiment, we relax the independence assumption and consider the case of dependent cross-flows.

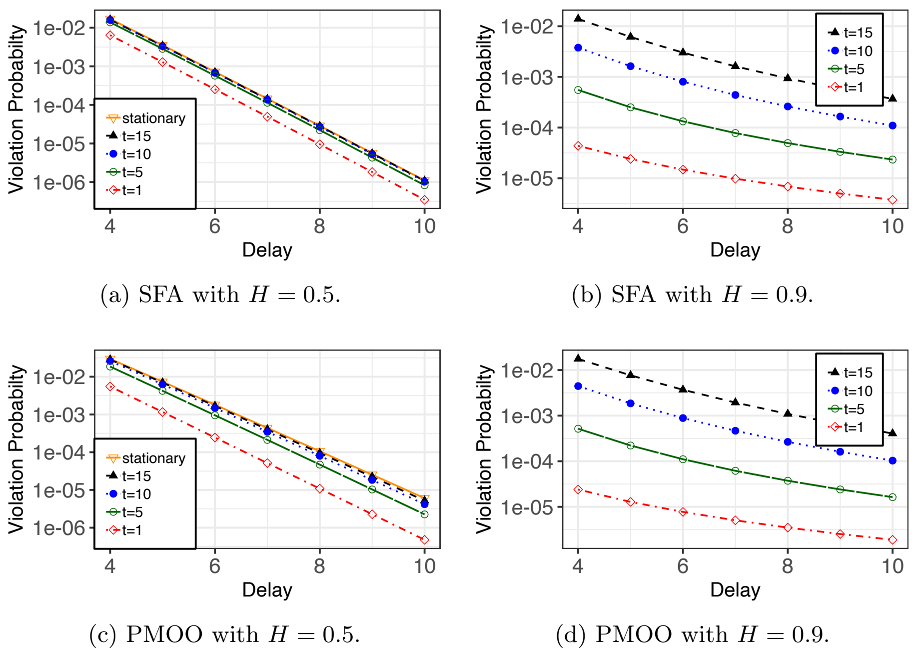
We assume the arrivals to be of the form (as in [27, 35, 38])

$$A(0,t) \coloneqq \lambda t + \sigma Z(t), \tag{8}$$

where $Z(t)$ is a normalized fractional Brownian motion, λ is the mean arrival rate and $\sigma^2 > 0$ is the variance of $A(0,1)$. The fBm $Z(t)$ is governed by the Hurst parameter $H \in (0,1)$ that, in turn, is independent of the other parameters [35]. This arrival process, also called fBm traffic model [27], has been shown to have the following MGF [24] for $t \geq 0$:

$$\mathrm{E}\left[e^{\theta A(0,t)} \right] = e^{\lambda \theta t + \frac{\theta^2 \sigma^2}{2} t^{2H}}. \tag{9}$$

For $H \in (0.5, 1)$, fBm exhibits a property called long-range dependence (LRD). If not mentioned otherwise, throughout the experiments, we choose $\lambda = 1, \sigma^2 = 1$, and $H = 0.7$.



(a) SFA with $H = 0.5$. (b) SFA with $H = 0.9$.

(c) PMOO with $H = 0.5$. (d) PMOO with $H = 0.9$.

Fig. 5. Delay violation probability for two servers and different t.

Given the continuous nature of fBm, the arrivals in Equation (8) are a continuous-time process

$$A(s,t) = \int_s^t a(x)\mathrm{d}x, \text{ where } 0 \leq s \leq t,$$

that has to be discretized in order to be applicable to our discrete-time arrival model (cf. Equation (1)).

We proceed as in [13]. Let $\tau > 0$ be a discretization parameter and $t \geq 0$. Then, assuming a dynamic S-server, it can be shown for the delay bound that

$$\mathrm{P}(d(t) > T) \leq \sum_{j=0}^{\left\lfloor \frac{t}{\tau} \right\rfloor} \mathrm{E}\left[e^{\theta A(t-(j+1)\tau, t)}\right] \mathrm{E}\left[e^{-\theta S(t-j\tau, t+T)}\right].$$

The rest follows along similar lines as in the discrete-time case.

If not explicitly specified, by default, the cross-flows are assumed to be independent and t is equal to 20. The server rate (we assume homogeneous sink trees) is denoted by $c > 0$. Further, all results are obtained by numerically optimizing θ and the Hölder parameters p_i.

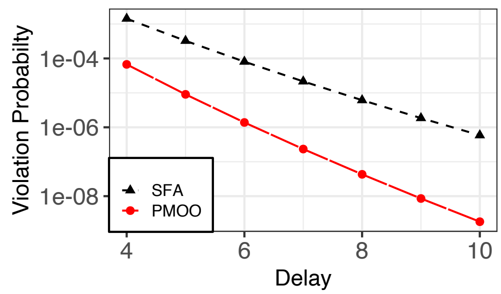


Fig. 6. Comparison between delay violation probabilities using SFA and PMOO.

4.1 Impact of a Finite Time Horizon

We compare the delay bounds for different time horizons t, applying the bounds for SFA (Proposition 2) and PMOO (Proposition 3), respectively. The results are depicted in Figure 5.

We observe that the delay bounds do not change significantly for larger t when the Hurst parameter $H = 0.5$ (Figure 5a and 5c). Since for this particular H, the fBm traffic model is (σ, ρ)-bounded (Definition 6), we can also derive stationary bounds that hold for all t. However, for $H = 0.9$ (Figure 5b and 5d), when the fBm traffic model exhibits a long-range dependence, the delay bounds vary strongly for different t. This indicates that, if one is aiming at transient bounds, results obtained from a stationary analysis may be too conservative.

4.2 Comparison between SFA and PMOO

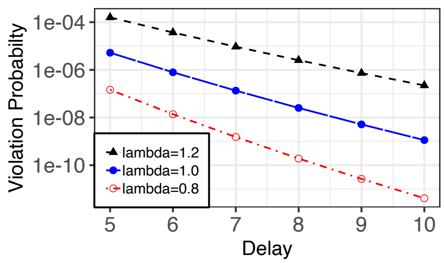
For a sink tree with two servers, we compare the delay bounds using SFA and PMOO. To that end, we consider a three-server sink tree with server rate $c = 6.0$.

The results in Figure 6 indicate a significant gap in the delay bounds. While the difference in the violation probability is about two orders of magnitude, in the delay space, the PMOO bound exhibits an improvement of roughly 30%. This is caused by the additional application of Hölder's inequality, that is only necessary in the SFA. Hence, in the following experiments, we only use PMOO.

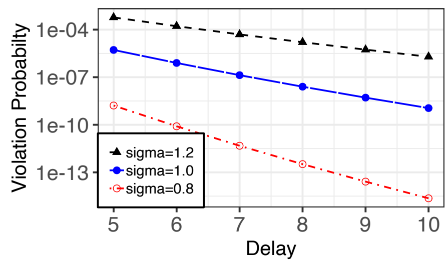
4.3 Parameter Sensitivity of Fractional Brownian Motion

In this subsection, we investigate the impact of the fBm traffic model parameters on the delay bounds. Therefore, for a three-server sink tree, we fixed the server rates to $c = 9.0$ and varied the parameters separately by 0.2. The results are shown in Figure 7.

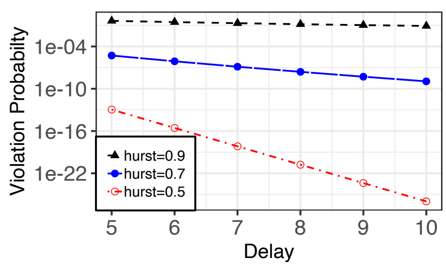
We see that, while all parameters clearly influence the outcome, the parameter sensitivity significantly differs. As expected, it is evident that, at the same load, the Hurst parameter H can be decisive whether the system suffers from long queues ($H = 0.9$), or hardly sees any queueing effects ($H = 0.5$) (Figure 7c).



(a) Mean arrival rate λ



(b) Standard deviation σ



(c) Hurst parameter H

Fig. 7. Parameter sensitivity of fractional Brownian motion on the delay bounds.

4.4 Scaling Effects of PMOO

In this experiment, we focus on how the delay violation probability scales with the number of servers for a delay of $T = 4$. Further, we keep the utilization at the last server (since it is the server with the heaviest load in a homogeneous sink tree, $\frac{(n+1)\lambda}{c}$) constant, i.e. we scale its capacity with the number of flows.

The results in Figure 8 show that the delay bounds improve with the number of servers. This improvement is due to statistical multiplexing effects as the number of flows grows.

4.5 Comparison Between Independent and Dependent Cross-Flows

So far, all experiments considered the cross-flows to be independent. In this last experiment, we now omit the independence assumption, i.e., we apply Hölder's inequality to the MGF of the cross-flows. The delay bounds for a sink tree of three servers with server rate $c = 9.0$ are depicted in Figure 9.

As expected, the impact of dependence (and therefore Hölder's inequality) is strong. The delay violation probability is about 9 orders of magnitude higher compared to the independent case. This indicates the importance of treating and, if possible, avoiding the invocation of Hölder's inequality.

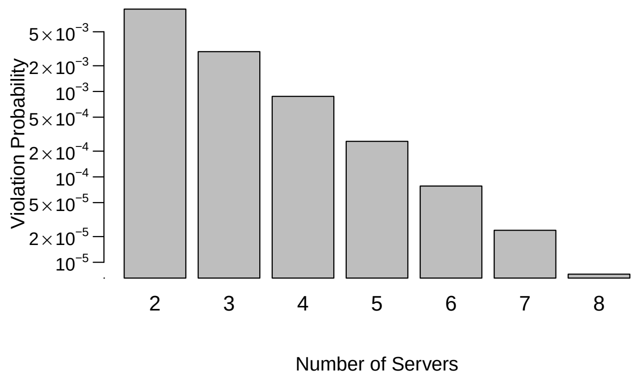


Fig. 8. Delay violation probability of the sink tree for different lengths and constant utilization at the last server.

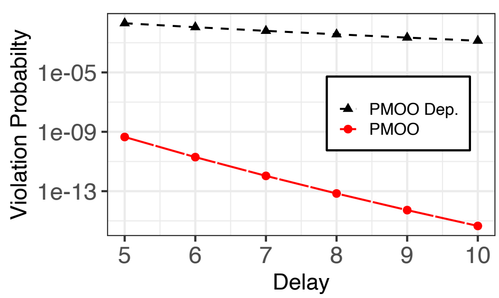


Fig. 9. Comparison between delay violation probabilities for independent and dependent cross-flows using PMOO.

5 Conclusion

In this paper, we have derived end-to-end delay bounds for a flow traversing a sink tree using a stochastic network calculus approach. It has been shown that pay multiplexing only once has to consider less stochastic dependencies, and therefore applies less Hölder inequalities in the analysis. Further, our numerical experiments with a fractional Brownian motion traffic model indicate that each application of Hölder's inequality significantly worsens the delay bound.

Overall, the e2e analysis still imposes many open problems in the stochastic network calculus. The most striking one is clearly how to take stochastic dependence into account. One possible approach could be to leverage negative dependence as in [33].

A Appendix

A.1 Proof of Proposition 3

Proof. We prove the theorem via induction. The base case $n = 2$ is already treated in Subsection 3.1.

Assume now that the induction hypothesis (IH) is true for some $n \in \mathbb{N}$. We denote the end-to-end service of tandems of length n by S_{e2e}^{n}. Observe that extending the sink tree basically means that we prolong all flows and add one flow that only traverses the last hop. Therefore, we apply the induction hypothesis on the last server n servers $S_2, \ldots, S_{n+1}$ and receive S_{e2e}^{n}. Afterwards, we basically apply the base case, as the network is reduced to the network consisting of S_1 and S_{e2e}^{n}. This gives

$$\begin{aligned} S_{\mathrm{e2e}}^{n+1} &= [(S_{\mathrm{e2e}}^{n} \otimes S_1) - A_2]^+ \\ &\overset{\text{(IH)}}{=} \left[\left(\left[\left(\left[\left([S_{n+1} - A_{n+2}]^+ \otimes S_n\right) - A_{n+1}\right]^+ \otimes \cdots \otimes S_2\right) - A_3\right]^+ \otimes S_1\right) - A_2\right]^+. \end{aligned}$$

For the delay bound, it follows that

$$\begin{aligned} &\mathrm{P}(d(t) > T) \\ &\overset{(3)}{\leq} \mathrm{E}\left[e^{\theta(A_1 \oslash S_{\mathrm{e2e}}(t+T,t))}\right] \\ &\leq \sum_{s_0=0}^{t} \mathrm{E}\left[e^{\theta A_1(s_0,t)}\right] \\ &\quad \cdot \mathrm{E}\left[e^{-\theta\left[\left(\left[\left(\left[\left([S_{n+1}-A_{n+2}]^+ \otimes S_n\right)-A_{n+1}\right]^+ \otimes \cdots \otimes S_2\right)-A_3\right]^+ \otimes S_1\right)-A_2\right]^+(s_0,t+T)}\right] \\ &\overset{\text{(IH)}}{\leq} \sum_{s_0=0}^{t} \mathrm{E}\left[e^{\theta A_1(s_0,t)}\right] \mathrm{E}\left[e^{\theta A_2(s_0,t+T)}\right] \left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\left[e^{-\theta S_1(s_1,t+T)}\right] \mathrm{E}\left[e^{\theta A_3(s_0,s_1)}\right]\right. \\ &\quad \cdot \left(\sum_{s_2=s_0}^{s_1} \mathrm{E}\left[e^{-\theta S_2(s_2,s_1)}\right] \mathrm{E}\left[e^{\theta A_4(s_0,s_2)}\right] \cdots \left(\sum_{s_{n-1}=s_0}^{s_{n-2}} \mathrm{E}\left[e^{-\theta S_{n-1}(s_{n-1},s_{n-2})}\right] \mathrm{E}\left[e^{\theta A_{n+1}(s_0,s_{n-1})}\right]\right.\right. \\ &\quad \left.\left.\left. \cdot \mathrm{E}\left[e^{-\theta\left([S_{n+1}-A_{n+2}]^+ \otimes S_n\right)(s_0,s_{n-1})}\right]\right)\right)\right) \\ &\leq \sum_{s_0=0}^{t} \mathrm{E}\left[e^{\theta A_1(s_0,t)}\right] \mathrm{E}\left[e^{\theta A_2(s_0,t+T)}\right] \left(\sum_{s_1=s_0}^{t+T} \mathrm{E}\left[e^{-\theta S_1(s_1,t+T)}\right] \mathrm{E}\left[e^{\theta A_3(s_0,s_1)}\right]\right. \\ &\quad \cdot \left(\sum_{s_2=s_0}^{s_1} \mathrm{E}\left[e^{-\theta S_2(s_2,s_1)}\right] \mathrm{E}\left[e^{\theta A_4(s_0,s_2)}\right] \cdots \left(\sum_{s_{n-1}=s_0}^{s_{n-2}} \mathrm{E}\left[e^{-\theta S_{n-1}(s_{n-1},s_{n-2})}\right] \mathrm{E}\left[e^{\theta A_{n+1}(s_0,s_{n-1})}\right]\right.\right. \end{aligned}$$

$$\cdot \left(\sum_{s_n = s_0}^{s_{n-1}} \mathrm{E}\left[e^{-\theta S_n(s_n, s_{n-1})} \right] \mathrm{E}\left[e^{\theta A_{n+2}(s_0, s_n)} \right] \mathrm{E}\left[e^{-\theta S_{n+1}(s_0, s_n)} \right] \right) \Bigg) \Bigg) \Bigg) .$$

This finishes the proof.

References

1. Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.P.: Synchronization and linearity: an algebra for discrete event systems. John Wiley & Sons Ltd (1992)
2. Beck, M.: Towards the analysis of transient phases with stochastic network calculus. In: IEEE 17th International Network Strategy and Planning Symposium (Networks'16) (2016)
3. Beck, M.A.: Advances in Theory and Applicability of Stochastic Network Calculus. Ph.D. thesis, TU Kaiserslautern (2016)
4. Becker, N., Fidler, M.: A non-stationary service curve model for performance analysis of transient phases. In: 2015 27th International Teletraffic Congress. pp. 116–124. IEEE (2015)
5. Bondorf, S., Nikolaus, P., Schmitt, J.B.: Quality and cost of deterministic network calculus – design and evaluation of an accurate and fast analysis. Proceedings of the ACM on Measurement and Analysis of Computing Systems (POMACS) **1**(1), 34 (2017)
6. Bouillard, A., Comte, C., de Panafieu, É., Mathieu, F.: Of kernels and queues: when network calculus meets analytic combinatorics. In: 2018 30th International Teletraffic Congress (ITC 30). vol. 2, pp. 49–54. IEEE (2018)
7. Bouillard, A., Thierry, É.: Tight performance bounds in the worst-case analysis of feed-forward networks. Discrete Event Dynamic Systems **26**(3), 383–411 (2016)
8. Boyes, H., Hallaq, B., Cunningham, J., Watson, T.: The industrial internet of things (iiot): An analysis framework. Computers in Industry **101**, 1–12 (2018)
9. Burchard, A., Liebeherr, J., Ciucu, F.: On superlinear scaling of network delays. IEEE/ACM Transactions on Networking **19**(4), 1043–1056 (2010)
10. Champati, J.P., Al-Zubaidy, H., Gross, J.: Transient delay bounds for multi-hop wireless networks. CoRR (2018)
11. Chang, C.S.: Stability, queue length, and delay of deterministic and stochastic queueing networks. IEEE Transactions on Automatic Control **39**(5), 913–931 (1994)
12. Chang, C.S.: Performance guarantees in communication networks. Springer, London (2000)
13. Ciucu, F., Burchard, A., Liebeherr, J.: Scaling properties of statistical end-to-end bounds in the network calculus. IEEE Transactions on Information Theory **52**(6), 2300–2312 (2006)
14. Ciucu, F., Schmitt, J.: Perspectives on network calculus – no free lunch, but still good value. In: Proc. ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM'12). pp. 311–322. New York, NY, USA (2012)
15. Cruz, R.L.: Quality of service management in integrated services networks. In: Proc. Semi-Annual Research Review, CWC, UCSD (1996)
16. Dong, F., Wu, K., Srinivasan, V.: Copula analysis for statistical network calculus. In: Proc. IEEE INFOCOM'15. pp. 1535–1543 (2015)

17. Fettweis, G.P.: The tactile internet: Applications and challenges. IEEE Vehicular Technology Magazine **9**(1), 64–70 (2014)
18. Fidler, M.: An end-to-end probabilistic network calculus with moment generating functions. In: Proc. IEEE IWQoS'06. pp. 261–270 (2006)
19. Fidler, M., Rizk, A.: A guide to the stochastic network calculus. IEEE Communications Surveys & Tutorials **17**(1), 92–105 (2015)
20. Fonseca, N.L., Mayor, G.S., Neto, C.A.: On the equivalent bandwidth of self-similar sources. ACM Transactions on Modeling and Computer Simulation (TOMACS) **10**(2), 104–124 (2000)
21. Jafari, F., Lu, Z., Jantsch, A., Yaghmaee, M.H.: Buffer optimization in network-on-chip through flow regulation. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **29**(12), 1973–1986 (2010)
22. Jasperneite, J., Neumann, P., Theis, M., Watson, K.: Deterministic real-time communication with switched ethernet. In: 4th IEEE International Workshop on Factory Communication Systems. pp. 11–18. IEEE (2002)
23. Jiang, Y., Liu, Y.: Stochastic network calculus, vol. 1. Springer (2008)
24. Kelly, F.P.: Notes on Effective Bandwidths. In: Kelly, F.P., Zachary, S., Ziedins, I. (eds.) Stochastic Networks: Theory and Applications, Royal Statistical Society Lecture Notes Series, vol. 4, pp. 141–168. Oxford University Press: Oxford (1996)
25. Koubaa, A., Alves, M., Tovar, E.: Modeling and worst-case dimensioning of cluster-tree wireless sensor networks. In: 2006 27th IEEE International Real-Time Systems Symposium (RTSS'06). pp. 412–421. IEEE (2006)
26. Leiserson, C.E.: Fat-trees: universal networks for hardware-efficient supercomputing. IEEE transactions on Computers **100**(10), 892–901 (1985)
27. Li, C., Burchard, A., Liebeherr, J.: A network calculus with effective bandwidth. IEEE/ACM Transactions on Networking **15**(6), 1442–1453 (2007)
28. Liebeherr, J., Burchard, A., Ciucu, F.: Delay bounds in communication networks with heavy-tailed and self-similar traffic. IEEE Transactions on Information Theory **58**(2), 1010–1024 (2012)
29. Mellia, M., Stoica, I., Zhang, H.: Tcp model for short lived flows. IEEE communications letters **6**(2), 85–87 (2002)
30. Mitrinovic, D.S., Vasic, P.M.: Analytic inequalities, vol. 1. Springer (1970)
31. Nelson, R.: Probability, stochastic processes, and queueing theory: the mathematics of computer performance modeling. Springer (1995)
32. Nikolaus, P., Schmitt, J.: On Per-Flow Delay Bounds in Tandem Queues under (In)Dependent Arrivals. In: Proceedings of 16th IFIP Networking 2017 Conference (NETWORKING'17). IEEE (2017)
33. Nikolaus, P., Schmitt, J., Ciucu, F.: Dealing with dependence in stochastic network calculus – using independence as a bound. Tech. Rep. 394/19, TU Kaiserslautern, Department of Computer Science. `https://disco.cs.uni-kl.de/discofiles/publicationsfiles/NSC19-1_TR.pdf` (2019)
34. Nikolaus, P., Schmitt, J., Schütze, M.: h-mitigators: Improving your stochastic network calculus output bounds. Computer Communications **144**, 188–197 (2019)
35. Norros, I.: On the use of fractional Brownian motion in the theory of connectionless networks. IEEE Journal on selected Areas in Communications **13**(6), 953–962 (1995)
36. Qian, Z., Bogdan, P., Tsui, C.Y., Marculescu, R.: Performance evaluation of NoC-based multicore systems: From traffic analysis to noc latency modeling. ACM Transactions on Design Automation of Electronic Systems (TODAES) **21**(3), 52 (2016)

37. Rizk, A., Fidler, M.: Leveraging statistical multiplexing gains in single-and multi-hop networks. In: Proc. IEEE Nineteenth IEEE International Workshop on Quality of Service (IWQoS '11). pp. 1–9 (2011)
38. Rizk, A., Fidler, M.: Non-asymptotic end-to-end performance bounds for networks with long range dependent fbm cross traffic. Elsevier Computer Networks **56**(1), 127–141 (2012)
39. Rosen, E., Viswanathan, A., Callon, R.: Multiprotocol label switching architecture. RFC 3031, RFC Editor (2001)
40. Schmitt, J., Bondorf, S., Poe, W.Y.: The sensor network calculus as key to the design of wireless sensor networks with predictable performance. Journal of Sensor and Actuator Networks **6**(3) (2017)
41. Schmitt, J., Zdarsky, F.A., Fidler, M.: Delay bounds under arbitrary multiplexing: When network calculus leaves you in the lurch ... In: Proc. IEEE International Conference on Computer Communications (INFOCOM'08). pp. 1669–1677. Phoenix, AZ, USA (2008)
42. Schmitt, J., Zdarsky, F.A., Martinovic, I.: Improving performance bounds in feed-forward networks by paying multiplexing only once. In: Proc. GI/ITG Conference on Measurement, Modeling, and Evaluation of Computer and Communication Systems (MMB'08). pp. 1–15 (2008)
43. Singla, A., Chandrasekaran, B., Godfrey, P.B., Maggs, B.: The internet at the speed of light. In: Proc. ACM Workshop on Hot Topics in Networks'14. pp. 1–7. HotNets-XIII (2014)
44. Wang, H., Shen, H., Wieder, P., Yahyapour, R.: A data center interconnects calculus. In: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). pp. 1–10. IEEE (2018)
45. Yaron, O., Sidi, M.: Performance and stability of communication networks via robust exponential bounds. IEEE/ACM Transactions on Networking **1**(3), 372–385 (1993)
46. Zhu, T., Berger, D.S., Harchol-Balter, M.: SNC-meister: Admitting more tenants with tail latency SLOs. In: Proc. ACM Symposium on Cloud Computing (SoCC'16). pp. 374–387 (2016)
47. Zografos, K.G., Androutsopoulos, K.N., Vasilakis, G.M.: A real-time decision support system for roadway network incident response logistics. Transportation Research Part C: Emerging Technologies **10**(1), 1–18 (2002)