Improving Output Bounds in the Stochastic Network Calculus Using Lyapunov's Inequality

Paul Nikolaus
Distributed Computer Systems (DISCO) Lab
TU Kaiserslautern, Germany
nikolaus@cs.uni-kl.de

Jens Schmitt
Distributed Computer Systems (DISCO) Lab
TU Kaiserslautern, Germany
jschmitt@cs.uni-kl.de

***Abstract*—Giving tight estimates for output bounds is key to an accurate network analysis using the stochastic network calculus (SNC) framework. In order to upper bound the delay for a flow of interest in the network, one typically has to calculate output bounds of cross-traffic flows several times. Thus, an improvement in the output bound calculation pays off considerably. In this paper, we propose a new output bound calculation in the SNC framework by making use of Lyapunov's inequality. We prove the bound's validity and also show why it is always at least as accurate as the state-of-the-art method. Numerical evaluations demonstrate that even in small heterogeneous two server topologies, our approach can improve a delay bounds' violation probability by a factor of 340. For a set of randomly generated parameters, the bound is still decreased by a factor of 1.33 on average. Furthermore, our approach can be easily integrated in existing end-to-end analyses.**

I. Introduction

A. Motivation

Providing delay bounds in packet-switched networks is a timeless challenge with recent sample applications as, e.g., Internet at the speed of light [1], Tactile Internet [2], Internet of Things [3], or the envisioned cyber-physical systems [4], which often face real-time requirements.

The Network Calculus (NC) holds the promise to enable tight end-to-end delay analysis in such advanced applications building on a modular and uniform mathematical framework based on min-plus algebra [5]. Starting from the 1990s with two papers by Cruz [6], [7], NC demonstrated its benefits providing tight bounds for deterministic worst-case end-to-end delay bounds. In the following, the Deterministic Network Calculus (DNC) was further elaborated and mathematically cast into a min-plus algebra setting [8], [9]. More recently, NC was generalized into a stochastic setting providing probabilistic worst-case bounds: the Stochastic Network Calculus (SNC) framework [8], [10]–[13]. SNC's main features can be summarized as providing a very general scheduling abstraction (the service curve) and the ability to enable system-wide end-to-end analysis (the concatenation theorem) [13].

SNC results can be categorized into different branches such as tail-bound based [10], [12], [14], moment generating functions (MGF) based [8], [11], and martingale based [15] approaches. Recent work evidences its applicability to modern

ISBN 978-3-903176-08-9

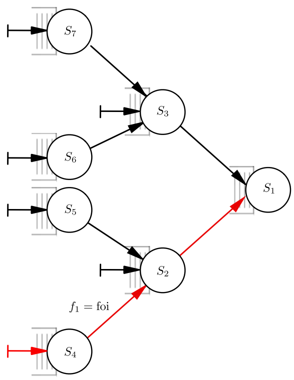


Fig. 1. Full binary sink tree with seven nodes.

problems, e.g., in the analysis of parallel systems (using the fork-join pattern) or multi-tenancy [16]–[18].

Typically a DNC/SNC network analysis proceeds along the following steps:

1) Reducing the network to a tandem of servers traversed by the flow of interest (foi) by invoking the output bound calculation to characterize cross-traffic flows at the servers where they join the foi.
2) Reducing the tandem of servers traversed by the flow of interest (foi) to a single server representing the whole system.
3) Calculating the delay bound of the foi at the single server representing the whole system.

Most of the existing NC literature has mainly focused on steps 2) and 3). In DNC, step 1) has seen some advanced treatment recently [19], but in SNC it has been largely neglected in the sense that no work beyond the standard output bound calculation was invested. In contrast to this, we focus on step 1) and, in particular, try to improve the SNC output bound calculation in this paper. As the output bound calculation has to be invoked numerous times in step 1), we believe its accuracy to be key in larger network analyses. For example: Assume a full binary tree of height h where each node represents a server and each of these servers has an arrival flow that is transmitted to the sink; let the foi be starting from one of the leaf nodes (see also Figure 1), then the number of output bound calculations is $2^h - h - 1$, whereas we only need to invoke the delay bound calculation once (in step 3)). Thus, any improvement in the output bound calculation pays off tremendously in larger network analyses.

Yet, how can we improve upon the standard SNC output bound calculation? The tail bound and MGF SNC analyses have the application of the so-called Union bound or Boole's inequality in common. In a series of publications, [15], [20]–[22], the authors emphasized its poor performance and suggested an appealing martingale-based approach. It provides tight single hop lower and upper bounds on the delay for different scheduling disciplines. Yet, to the best of our knowledge, so far there is no concatenation result in the martingale-based SNC and thus step 2) from above cannot be performed and, thus, an elegant end-to-end analysis remains elusive. Hence, we decided to remain within the standard SNC framework and, yet, try to counteract the inherent problems of the Union bound.

B. Main Contribution

In this paper, we present a modification of the MGF-based SNC that mitigates the Union bound's effect in the output bound calculation. It consists of the application of Lyapunov's inequality just before the invocation of the Union bound and does not impose any additional assumptions. It is thus minimally invasive and all existing results and procedures of the SNC are literally still applicable while, as we see below, it improves the performance bounds. In fact, we prove this new bound to be always at least as good as the state-of-the-art method and show that in a very simple heterogeneous two server setting, it can improve the delay bound already by a factor of up to 340.

It comes, however, at the price of an additional parameter per invocation of Lyapunov's inequality. Thus, we trade higher computational effort in the optimization of these parameters for improved bounds. However, as we also show this effort is moderate if the optimization is done carefully.

C. Outline

The rest of the paper is structured as follows: In Section II, we introduce the necessary notations for SNC and its main results as we need them in this paper. In Section III, we present our new output bound calculation and prove its validity. A numerical evaluation is given in Section IV: we compare output bounds for a single server and delay bounds for a two server setting as well as a fat tree topology with the current state of the art method. In Section V, we prove that Lyapunov's inequality cannot be applied directly to delay bounds. Section VI concludes the paper.

II. SNC Background and Notation

In this section, we introduce some of the basic terms and concepts in SNC.

We use the MGF-based SNC in order to calculate per-flow delay bounds. To be precise, we bound the probability that the delay exceeds a given value, typically denoted by T. The connection between probability bounds and MGFs is established by Chernoff's bound

$$\mathrm{P}(X > a) \leq e^{-\theta a}\,\mathrm{E}\big[e^{\theta X}\big]\,, \quad \theta > 0, \tag{1}$$

with $\mathrm{E}\big[e^{\theta X}\big]$ as the moment-generating function (MGF) of a random variable X. We define an *arrival flow* by the stochastic process A with discrete time space $\mathbb{N}$ and continuous state space $\mathbb{R}_0^+$ as $A(s,t) \coloneqq \sum_{i=s+1}^{t} a\,(i)$, with $a(i)$ as the traffic increment process in time slot i. Network calculus provides an elegant system-theoretic analysis by employing min-plus algebra:

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

$$\begin{aligned}(x \otimes y)\ (s,t) &\coloneqq \min_{s \leq i \leq t} \{x\,(s,i) + y(i,t)\}\,,\\ (x \oslash y)\ (s,t) &\coloneqq \max_{0 \leq i \leq s} \{x(i,t) - y(i,s)\}\,.\end{aligned} \tag{2}$$

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

Definition 2 (Dynamic S-Server)**.** Assume a service element has an arrival flow A as its input and the respective output is denoted by A'. 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:

$$A'(0,t) \geq (A \otimes S)\ (0,t) = \min_{0 \leq i \leq t} \{A(0,i) + S(i,t)\}\,.$$

The analysis in this paper is based on a per-flow perspective. I.e., 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 [23]. That is, if flow f_2 is prioritized over flow f_1, the leftover service for the corresponding arrival A_1 is $S_{\text{l.o.}} = [A_2 - S]^+$. Furthermore, we require the server to be work-conserving.

In the following definition, we introduce (σ, ρ)-constraints [8] as they enable us to give stationary bounds under stability conditions.

Definition 3 ((σ, ρ)-Bound)**.** An arrival flow is $(\sigma_A(\theta), \rho_A(\theta))$-bounded for some $\theta > 0$, if its MGF exists and for all $0 \leq s \leq t$

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

A dynamic S-server is $(\sigma_S(\theta), \rho_S(\theta))$-bounded for some $\theta > 0$, if its MGF exists and for all $0 \leq s \leq t$

$$\mathrm{E}\Big[e^{-\theta S(s,t)}\Big] \leq e^{\theta(\rho_S(\theta)(t-s)+\sigma_S(\theta))}.$$

Definition 4 (Virtual Delay)**.** The *virtual delay* at time $t \geq 0$ is defined as

$$d(t) \coloneqq \inf\{s \geq 0 : A(0,t) \leq A'(0,t+s)\}\,.$$

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

Theorem 5 (Output and Delay Bound). *[11] [24] Consider an arrival process $A(s,t)$ with dynamic S-server $S(s,t)$.*

The departure process A' is upper bounded for any $0 \leq s \leq t$ according to

$$A'(s,t) \leq (A \oslash S)\ (s,t).$$

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

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

We focus on the analogue of Theorem 5 for moment generating functions:

Theorem 6 (Output and Delay MGF-Bound). *[11] [24] For the assumptions as in Theorem 5, we obtain:*

The MGF of the departure process A' is upper bounded for any $0 \leq s \leq t$ according to

$$\mathrm{E}\Big[e^{\theta A'(s,t)}\Big] \leq \mathrm{E}\Big[e^{\theta (A \oslash S)\ (s,t)}\Big]. \tag{3}$$

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

$$\mathrm{P}(d(t) > T) \leq \mathrm{E}\Big[e^{\theta (A \oslash S)\ (t+T,t)}\Big]. \tag{4}$$

III. New Output Bound Calculation

In this section, we derive our new approach to compute the MGF-output bound. Furthermore, we apply this idea to $(\sigma(\theta), \rho(\theta))$-bounded arrivals and service.

A. Insertion of Lyapunov's Inequality

The most intuitive way to bound (3) is to continue with

$$\mathrm{E}\Big[e^{\theta (A \oslash S)\ (s,t)}\Big] \overset{(2)}{=} \mathrm{E}\Big[e^{\theta \max_{0 \leq i \leq s}\{A(i,t)-S(i,s)\}}\Big] \leq \sum_{i=0}^{s} \mathrm{E}\Big[e^{\theta (A(i,t)-S(i,s))}\Big], \tag{5}$$

where the max is always less than or equal to the sum since we have only non-negative terms. Inequality (5) is similar to the application of the Union bound[1],

$$\mathrm{P}\Big(\max_{i=1,\ldots,n} X_i > a\Big) \leq \sum_{i=1}^{n} \mathrm{P}(X_i > a). \tag{6}$$

It has been shown to often perform poorly, in particular for correlated increments. The authors of [25] suggested instead a martingale-based approach that allows for significantly more accurate delay bounds. To the best of our knowledge, however, achieving a concatenation property to enable an end-to-end analysis remains an elusive goal in the martingale-based approach.

[1]For probability bounds such as the backlog or the delay, it is even equivalent to the Union bound, as

$$\mathrm{P}\Big(\max_{i=1,\ldots,n} X_i > a\Big) \overset{(6)}{\leq} \sum_{i=1}^{n} \mathrm{P}(X_i > a) \overset{(1)}{\leq} e^{-\theta a} \sum_{i=1}^{n} \mathrm{E}\Big[e^{\theta X_i}\Big]$$

$$\Leftrightarrow \mathrm{P}\Big(\max_{i=1,\ldots,n} X_i > a\Big) \overset{(1)}{\leq} e^{-\theta a}\,\mathrm{E}\Big[\max_{i=1,\ldots,n} e^{\theta X_i}\Big] \overset{(5)}{\leq} e^{-\theta a} \sum_{i=1}^{n} \mathrm{E}\Big[e^{\theta X_i}\Big]$$

Therefore, we call inequality (5) in the following "quasi-Union bound."

In this paper, we use Lyapunov's inequality to mitigate the Union bound's effect. Yet, as we see in Subsection III-B, existing end-to-end analyses are still applicable.

Proposition 7 (Lyapunov Inequality). *Let $X \geq 0$ be in $\mathcal{L}^l$ with $l \geq 1$. Then it holds that*

$$\mathrm{E}[X] \leq \big(\mathrm{E}\big[X^l\big]\big)^{\frac{1}{l}}. \tag{7}$$

Remark 8. Proposition 7 is a special case of *Jensen's inequality* [26]:

$$h(\mathrm{E}[X]) \leq \mathrm{E}[h(X)], \tag{8}$$

where h is a differentiable convex function on $\mathbb{R}$. The fact that X must be in $\mathcal{L}^l$ has a negligible effect since $l = 1$ should always be feasible, i.e., $\mathrm{E}[X]$ exists. As the random variables of our interest have existing MGF bounds, this should be a very mild assumption.

Since $l = 1$ is feasible for $X \in \mathcal{L}^1$, (7) can be rewritten as

$$\mathrm{E}[X] = \inf_{l \geq 1} \Big\{\big(\mathrm{E}\big[X^l\big]\big)^{\frac{1}{l}}\Big\}. \tag{9}$$

Using (9) one step before the quasi-Union bound's invocation (5) leads to

$$\begin{aligned}\mathrm{E}\Big[e^{\theta A'(s,t)}\Big] &\leq \mathrm{E}\Big[e^{\max_{0 \leq i \leq s}\{A(i,t)-S(i,s)\}}\Big] \\ &\overset{(9)}{=} \inf_{l \geq 1} \left\{\left(\mathrm{E}\Big[e^{l\theta \max_{0 \leq i \leq s}\{A(i,t)-S(i,s)\}}\Big]\right)^{\frac{1}{l}}\right\} \\ &\overset{(5)}{\leq} \inf_{l \geq 1} \left\{\left(\sum_{i=0}^{s} \mathrm{E}\Big[e^{l\theta (A(i,t)-S(i,s))}\Big]\right)^{\frac{1}{l}}\right\}.\end{aligned} \tag{10}$$

This new bound is obviously always at least as accurate as the quasi-Union bound (5), since $l = 1$ is feasible. The reason why this can improve over previous estimation lies in the subadditivity of the root function. It yields the following relation:

$$\begin{aligned}&\inf_{l \geq 1} \left\{\left(\sum_{i=0}^{s} \mathrm{E}\Big[e^{l\theta (A(i,t)-S(i,s))}\Big]\right)^{\frac{1}{l}}\right\} \\ &\leq \inf_{l \geq 1} \left\{\sum_{i=0}^{s} \left(\mathrm{E}\Big[e^{l\theta (A(i,t)-S(i,s))}\Big]\right)^{\frac{1}{l}}\right\}.\end{aligned}$$

The infimum on the right hand side is achieved at $l = 1$, which proves again that Lyapunov's inequality cannot worsen the bound's tightness. Yet, the subadditivity also implies that the insertion of Lyapunov's inequality can mitigate the effect of the quasi-Union bound (5), since we take the root outside of the sum. As our numerical evaluation in Section IV shows, in some cases a significant improvement for the output bound is achieved despite this method's minimal invasiveness.

B. Application to (σ, ρ)-Bounds

The bounds in (5) and (10) give an estimate for the min-plus operators in Theorem 6, but are computationally infeasible for larger networks. Since the number of sums in these calculations typically scales linearly with the number of invoked min-plus operators, one usually seeks for stationary closed-form

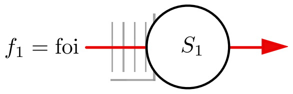


Fig. 2. One server topology.

solutions. Using (σ,ρ)-bounds (Definition 3) conveniently solves this problem by letting these sums converge, as the next proposition together with its corresponding remark show.

Proposition 9. *[24] Consider a $(\sigma_A(\theta),\rho_A(\theta))$-bounded arrival process $A(s,t)$ with $(\sigma_S(\theta),\rho_S(\theta))$-bounded dynamic S-server $S(s,t)$. If the stability condition $\rho_A(\theta) < -\rho_S(\theta)$ holds, then the output A' is $(\sigma_{A'}(\theta),\rho_{A'}(\theta))$-bounded with*

$$\begin{aligned}\sigma_{A'}(\theta) =& \sigma_A(\theta)+\sigma_S(\theta) - \frac{1}{\theta}\log\left(1-e^{\theta(\rho_A(\theta)+\rho_S(\theta))}\right),\\ \rho_{A'}(\theta) =& \rho_A(\theta).\end{aligned}$$

Remark 10. The computational advantage can be observed as follows:
The quasi-Union bound yields $\mathrm{E}\left[e^{\theta A'(s,t)}\right] \overset{(5)}{\leq} \sum_{i=0}^{s}\mathrm{E}\left[e^{\theta(A(i,t)-S(i,s))}\right]$, i.e., we have to compute a sum with $s+1$ summands. With the additional assumption of (σ,ρ)-constraints, the output can be bounded by the closed form $\frac{e^{\theta(\rho_A(\theta)(t-s)+\sigma_A(\theta)+\sigma_S(\theta))}}{1-e^{\theta(\rho_A(\theta)+\rho_S(\theta))}}$ (see Subsection (III-C) for details).

By an analogous calculation, we obtain for our new output bound the following result:

Proposition 11. *Under the same assumptions as in Proposition 9, under the stability condition $\rho_A(l\theta) < -\rho_S(l\theta)$ we obtain that the output A' is $(\sigma_{A'}(\theta),\rho_{A'}(\theta))$-bounded with*

$$\begin{aligned}\sigma_{A'}(\theta) =& \sigma_A(l\theta)+\sigma_S(l\theta) - \frac{1}{l\theta}\log\left(1-e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))}\right),\\ \rho_{A'}(\theta) =& \rho_A(l\theta),\end{aligned}$$

where $l \geq 1$.

Proof: See Appendix A. ■

Thus, this new output bound can be also used within (σ,ρ)-constraints. I.e., it can easily be integrated in existing end-to-end analyses.

C. Single Server Example

Assume a single flow - single server setting as in Figure 2. We have already deduced that

$$\begin{aligned}\mathrm{E}\left[e^{\theta\left(A'(s,t)\right)}\right] &\overset{(3)}{\leq} \mathrm{E}\left[e^{\theta(A\oslash S)\,(s,t)}\right]\\ &\overset{(5)}{\leq} \sum_{i=0}^{s}\mathrm{E}\left[e^{\theta(A(i,t)-S(i,s))}\right].\end{aligned}$$

Given that the arrivals and the service have (σ,ρ)-constraints, for $\rho_A(\theta) < -\rho_S(\theta)$ we continue with

$$\begin{aligned}\mathrm{E}\left[e^{\theta\left(A'(s,t)\right)}\right] &\leq \sum_{i=0}^{s}\mathrm{E}\left[e^{\theta(A(i,t)-S(i,s))}\right]\\ &= \sum_{i=0}^{s}\mathrm{E}\left[e^{\theta A(i,t)}\right]\mathrm{E}\left[e^{-\theta S(i,s)}\right]\\ &\leq \sum_{i=0}^{s} e^{\theta\rho_A(\theta)(t-i)+\theta\sigma_A(\theta)}e^{\theta\rho_S(\theta)(s-i)+\theta\sigma_S(\theta)}\\ &= e^{\theta(\rho_A(\theta)(t-s)+\sigma_A(\theta)+\sigma_S(\theta))}\\ &\quad\cdot\sum_{j=0}^{s} e^{\theta(\rho_A(\theta)+\rho_S(\theta))j}\\ &\leq \frac{e^{\theta(\rho_A(\theta)(t-s)+\sigma_A(\theta)+\sigma_S(\theta))}}{1-e^{\theta(\rho_A(\theta)+\rho_S(\theta))}}, \qquad (11)\end{aligned}$$

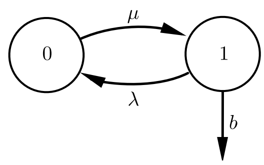


Fig. 3. MMOO Model.

where we have used the independence of arrivals and service in the second line, (σ,ρ)-bounds in the third line and the convergence of the geometric series in the last line.

If we used Lyapunov inequality instead, we would obtain in comparison

$$\begin{aligned}&\mathrm{E}\left[e^{\theta A'(s,t)}\right]\\ &\leq \inf_{l\geq 1}\left\{\left(\frac{e^{l\theta(\rho_A(l\theta)(t-s)+\sigma_A(l\theta)+\sigma_S(l\theta))}}{1-e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))}}\right)^{\frac{1}{l}}\right\}\\ &\leq \inf_{l\geq 1}\left\{\frac{e^{\theta(\rho_A(l\theta)(t-s)+\sigma_A(l\theta)+\sigma_S(l\theta))}}{\left(1-e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))}\right)^{\frac{1}{l}}}\right\}. \qquad (12)\end{aligned}$$

IV. Evaluation

In this section, we investigate the increased accuracy of our new output bound introduced in Section III. That is, we evaluate the gain of the output bound calculation in a single server setting in conjunction with the delay bound for a two server topology and a fat tree. The improvement factor is measured by calculating

$$\frac{\text{Bound standard approach}}{\text{Bound new method}},$$

where clearly larger values are desirable.

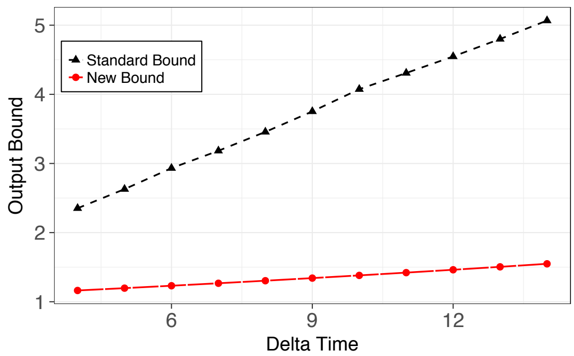
The formulae are implemented in the general-purpose programming language **Java**[2], version 8.

The arrivals are either exponentially distributed with parameter λ, i.e.,

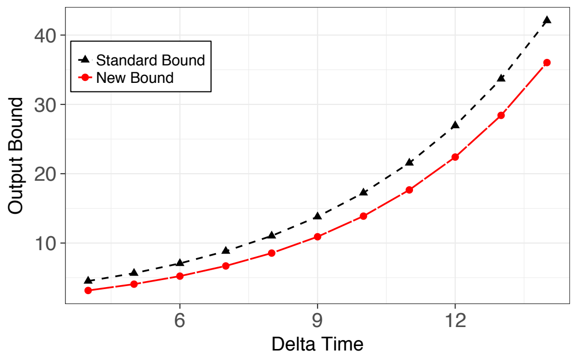
$$\mathrm{E}\left[e^{\theta A(s,t)}\right] = \left(\frac{\lambda}{\lambda-\theta}\right)^{t-s}, \quad 0<\theta<\lambda,$$

or follow the Markov-Modulated On-Off (MMOO) traffic model. That is, it consists of a continuous-time Markov chain with two states, 0 and 1, together with transition rates μ and

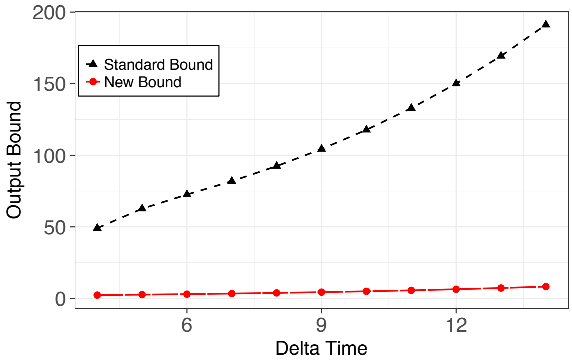
[2] https://java.com



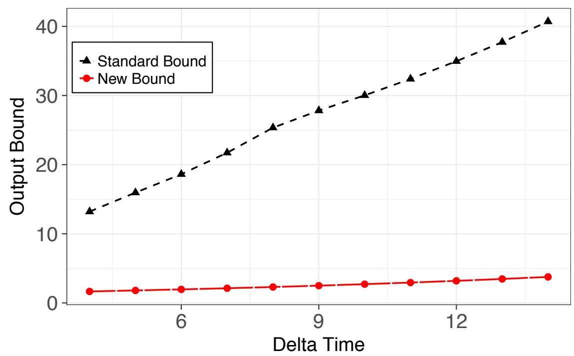
(a) Exponential arrivals with $\lambda = 3.8$, service rate $r = 3$



(b) Exponential arrivals with $\lambda = 0.5$, service rate $r = 10$



(c) MMOO with $\mu = 8$, $\lambda = 12$, $b = 3$, service rate $r = 1.5$



(d) MMOO with $\mu = 4$, $\lambda = 12$, $b = 3$, service rate $r = 1.5$

Fig. 4. Output bound comparison in the single server setting.

λ. If it is in state 0, it means that no traffic arrives, whereas in state 1, data with burst rate b are sent (see Figure 3). It has been shown in [27] that, for this arrival model, the MGF can be bounded by

$$\mathrm{E}\left[e^{\theta A(s,t)}\right] \leq e^{\theta\omega(\theta)\cdot(t-s)}, \quad \theta > 0,$$

where $\omega(\theta) = \frac{-d+\sqrt{d^2+4\mu\theta b}}{2\theta}$ and $d = \mu + \lambda - \theta b$. The service is always chosen to be work-conserving and of constant rate.

If not stated otherwise, θ and the Lyapunov parameters l_i are optimized by a naive grid search, i.e, we define points along a grid for each parameter, calculate the bound for each combination, and take the one with the best objective value.

With each application of this new inequality, an additional parameter has to be optimized. On the other hand, since the costs of incorporating Lyapunov's inequality in a given implementation are rather moderate, it gives us convenient new options: Either we prioritize accuracy and optimize all l_i (at the cost of higher computational effort), or focus more on speed setting many $l_i = 1$ (setting all l_i equal to 1 would yield the old approach). Hence, we gain more flexibility while being minimally invasive at the same time.

A. Single Server

For the single hop topology (Figure 2), we calculated the bounds in (11) and (12). For exponentially distributed arrivals and Markov-Modulated On-Off (MMOO) traffic, two examples for each distribution are depicted in Figure 4. As we can observe from these examples, the actual gain from our new output bound calculation can vary strongly depending on the scenarios' parameters. For that reason, we decided to systematically sample the parameter spaces in a Monte Carlo-type fashion. That is, we took samples from a uniform distribution as well as an exponential distribution (since the parameter space is only lower bounded) and computed the

Distribution	Average gain	Maximum gain
Exponential	1.30	1025.0
MMOO	1.34	381.9

Distribution	Average gain	Maximum gain
Exponential	1.23	233.7
MMOO	1.62	3449.9

TABLE I
OUTPUT BOUND IMPROVEMENT FOR A SINGLE SERVER (ABOVE: UNIFORM SAMPLING, BELOW: EXPONENTIAL SAMPLING).

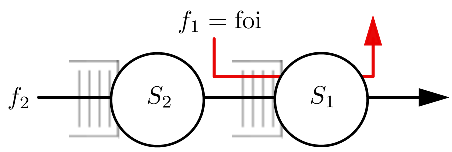
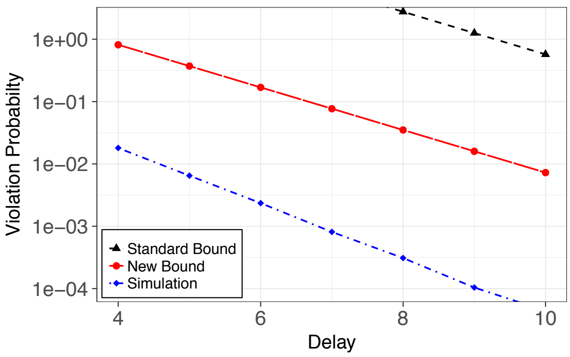
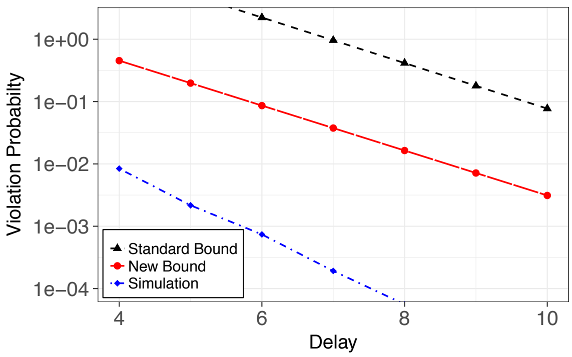


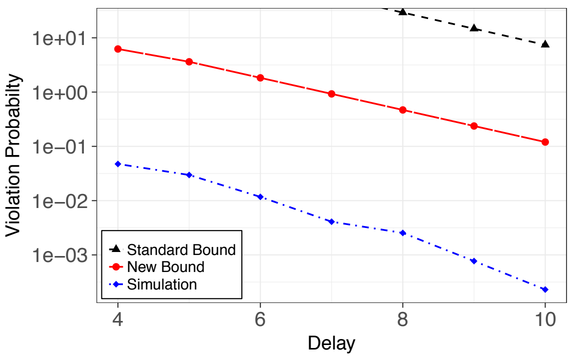
Fig. 5. Two server topology.



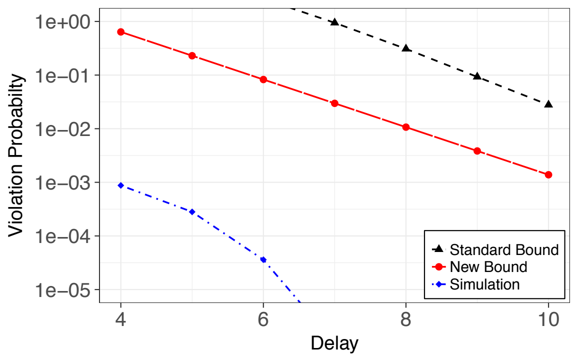
(a) Exponential arrivals with (λ_1, λ_2) = $(0.2, 8.0)$, service rates (r_1, r_2) = $(8.0, 0.2)$



(b) Exponential arrivals with (λ_1, λ_2) = $(0.4, 3.5)$, service rates (r_1, r_2) = $(4.5, 0.4)$



(c) MMOO with (μ_1, μ_2) = $(1.2, 3.7)$, (λ_1, λ_2) = $(2.1, 1.5)$, (b_1, b_2) = $(3.5, 0.4)$, service rates (r_1, r_2) = $(2.0, 0.3)$



(d) MMOO with (μ_1, μ_2) = $(1.0, 3.6)$, (λ_1, λ_2) = $(2.2, 1.6)$, (b_1, b_2) = $(3.4, 0.4)$, service rates (r_1, r_2) = $(2.0, 0.3)$

Fig. 6. Delay bound comparison in the two server setting.

average and largest improvement. The results are given in Table I.

We observe the possible gain to vary strongly with a maximum ratio Standard Bound / New Bound of three orders of magnitude. The overall average improvement factor is about 1.37, where exponentially distributed samples lead to larger improvements than the uniform ones.

B. *Two Server Topology*

In the previous subsection, we show that vast improvement on the output bound is possible in some cases. Next, we investigate the effect on the delay bound. Therefore, we extend the previous setting by an additional server (Figure 5). Here, a cross flow f_2 enters server S_2 and its output $(\leq (A_2 \oslash S_2))$ is prioritized over the flow of interest f_1 at server S_1. The improved output bound impacts the delay by being more accurate in terms of the foi's leftover service. Mathematically speaking, this leftover service at S_1 is described by $S_{1,\text{l.o.}} = [S_1 - (A_2 \oslash S_2)]^+$. In this topology, we calculate the delay bound (4) but take the new output bound invocation into account. Again, we display exponentially distributed arrivals and MMOO traffic. The plot is complemented by delay measurements in a packet-level simulation. Here, the violation probability is estimated by the empirical distribution computing the average number of occurred delays. All parameters are again randomly sampled by the Monte-Carlo type approach from the previous subsection.

Distribution	Average gain	Maximum gain
Exponential	1.14	255.2
MMOO	1.23	100.7

Distribution	Average gain	Maximum gain
Exponential	1.76	85.5
MMOO	1.81	342.0

TABLE II
IMPROVEMENT OF THE DELAY'S VIOLATION PROBABILITY FOR THE TWO SERVER SETTING (ABOVE: UNIFORM SAMPLING, BELOW: EXPONENTIAL SAMPLING).

As for the output bound, we often observe an improved delay bound, as one can see in the examples of Figure 6. It shows that even in the delay space (the difference in the delay bound for a given probability), the difference is up to 50%. Depending on the parameters, the gap between the simulation results and the analytically derived bounds can be closed considerably. Average behavior on the other hand is less significant. Table II indicates a highly non-linear behavior where some violation probabilities are improved by a factor of 342.0, whereas average gain is moderate with a total mean of 1.33.

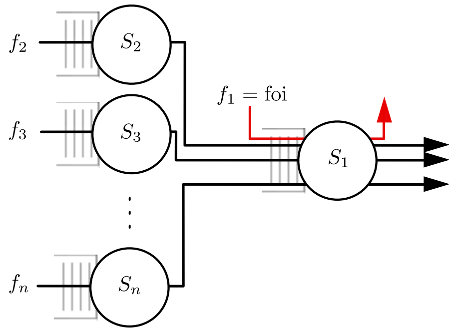


Fig. 7. Fat tree topology.

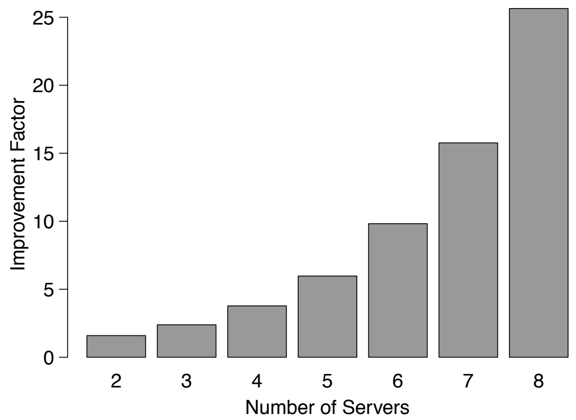


Fig. 8. Delay bound improvement for different numbers of servers.

C. Fat Tree

Starting off with the two server topology in Figure 5, we investigate the delay bound's scaling behavior for multiple invocations of Lyapunov's inequality. We now take a look at n flows, where $n-1$ are cross flows with corresponding server and their outputs jointly enter server S_1 (see Figure 7). The flow of interest is again, due to arbitrary multiplexing, assumed to be served after the cross traffic. In terms of leftover service provided for the foi, this means $S_{1,\text{l.o.}} = \left[S_1 - \sum_{i=2}^{n} (A_i \oslash S_i)\right]^+$.

We calculated the delay's violation probability for the following setting: The foi is exponentially distributed with parameter $\lambda_1 = 0.5$ and enters server S_1 with rate 4.5. The $n-1$ cross flows are also exponentially distributed, but with parameters $\lambda_i = 8$, $i = 2, \ldots, n$ and corresponding servers S_i with rates $r_i = 2$, $i = 2, \ldots, n$. The accuracy gain for different numbers of servers is depicted in Figure 8.

We observe that the ratio increases quickly to 25.6 in the case of 8 servers, even though only an improvement of 1.59 was achieved for the two server setting. This shows that the Lyapunov approach can fully develop its strengths in larger networks, when more output bound calculations have to be invoked.

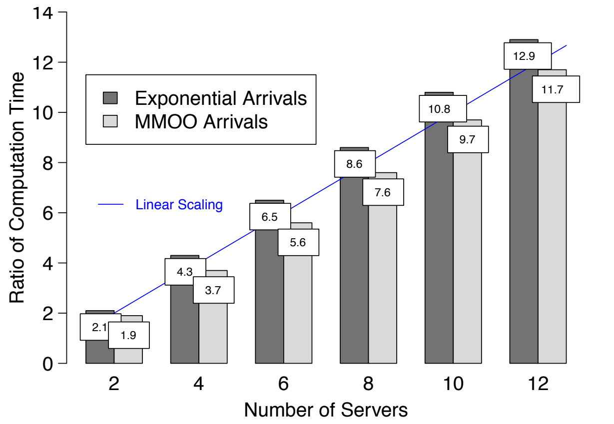


Fig. 9. Computation time comparison for the state-of-the-art and Lyapunov approach.

D. Runtime

So far, we focused on the Lyapunov bound's accuracy gain and observed favorable outcomes. Yet, the other side of the coin is the computational effort the new output bound calculation must invest to optimize over the higher-dimensional parameter space. To investigate this in more detail, we ran 10^4 experiments for exponentially distributed arrivals as well as MMOO-traffic in the two server topology (Figure 5) and the fat tree (Figure 7) with $2, 4, \ldots, 12$ flows. In this scenario, the aforementioned naive grid optimization runs quickly into computational problems, as a computation for 4 flows already took approximately a day. Therefore, we implemented the so called Pattern Search [28]. Here, a function is minimized by changing arguments only in a single direction. If multiple modifications lead to a descent, a step in the direction of all successful intermediate steps is attempted. The results of the ratio

$$\frac{\text{Computation time new method}}{\text{Computation time standard approach}}$$

for these experiments are depicted in Figure 9.

Under Pattern Search, we observe that computational overhead scales only linearly with the number of invocations of the Lyapunov inequality. This indicates that a good trade-off between cost and accuracy gain can be achieved, if optimization is done carefully.

V. Direct Application to Delay Bounds

At first glance, it is tempting to apply Lyapunov's inequality to the delay bound calculation as well. That is, we would modify the computation of the delay's violation probability as follows:

$$\begin{aligned} \mathrm{P}(d(t) > T) &\overset{(4)}{\leq} \mathrm{E}\left[e^{\theta (A \oslash S)\,(t+T,t)}\right] \\ &\overset{(2)}{=} \mathrm{E}\left[e^{\theta \max_{0 \leq i \leq t+T} \{A(i,t) - S(i,t+T)\}}\right] \\ &= \mathrm{E}\left[e^{\theta \max_{0 \leq i \leq t} \{A(i,t) - S(i,t+T)\}}\right] \end{aligned}$$

$$\overset{(9)}{=} \inf_{l\geq 1} \left\{ \left(\mathrm{E}\left[e^{l\theta \max_{0\leq i\leq t}\{A(i,t)-S(i,t+T)\}} \right] \right)^{\frac{1}{l}} \right\}$$

$$\overset{(5)}{\leq} \inf_{l\geq 1} \left\{ \left(\sum_{i=0}^{t} \mathrm{E}\left[e^{l\theta (A(i,t)-S(i,t+T))} \right] \right)^{\frac{1}{l}} \right\}, \quad (13)$$

where we used that $A(s,t) = 0$ for $s \geq t$ in the third line and the quasi-Union bound in the last inequality. Owing to the fact that this estimates a probability, only values below 1 are of interest for (13). Disappointingly for this case, no improvement can be obtained, as the next theorem states.

Theorem 12. *Let a delay bound T according to* (13) *exist such that*

$$\sum_{i=0}^{t} \mathrm{E}\left[e^{l\theta(A(i,t)-S(i,t+T))} \right] < 1. \quad (14)$$

If l and θ are optimized (denoted by l^* and θ^*), then $l^* = 1$, i.e., no improvement can be achieved.

Proof: Assume that l^* and θ^* are the optimal parameters for (13) and that $l^* > 1$. This means that there exist $1 \leq l' < l^*$ and $\theta' > \theta^*$ such that $l'\theta' = l^*\theta^*$. But this means

$$\left(\sum_{i=0}^{t} \mathrm{E}\left[e^{l^*\theta^*(A(i,t)-S(i,t+T))} \right] \right)^{\frac{1}{l^*}}$$

$$= \left(\sum_{i=0}^{t} \mathrm{E}\left[e^{l'\theta'(A(i,t)-S(i,t+T))} \right] \right)^{\frac{1}{l^*}}$$

$$> \left(\sum_{i=0}^{t} \mathrm{E}\left[e^{l'\theta'(A(i,t)-S(i,t+T))} \right] \right)^{\frac{1}{l'}},$$

where we inserted $l^*\theta^* = l'\theta'$ in the second line. In the third line, we used that $x^{\frac{1}{l^*}} > x^{\frac{1}{l'}}$ holds for all $x \in (0,1)$ and $l^* > l' \geq 1$. Clearly, this is a contradiction to our assumption that we had an optimal solution. Thus, the optimal l^* must be equal to 1. ∎

As a consequence, the Lyapunov approach can only indirectly decrease delay bounds via the output bound calculation. The same holds for the backlog bound (the proof follows along the same lines).

VI. Conclusion

In this paper, we proposed a novel approach to improve the MGF output bound calculation in the Stochastic Network Calculus using Lyapunov's inequality. We also gave a proof that shows why this is a valid bound and that it is always at least as accurate as the state-of-the-art method. It is also shown in comprehensive numerical evaluations that the delay's violation probability can be improved for two server topologies as well as fat trees. Our evaluation indicated a significant gain in some cases while leading to more moderate improvements on average. For a fat tree, we observed a very high gain as the number of cross flows is increased. These gains come conceptually for free, as no additional constraints have to be imposed, thus making our approach minimally invasive. Yet, from a computational perspective the gain comes at the price of a higher-dimensional optimization in the last stage of computing the bounds. Fortunately, our experiments indicate that the computational overhead only scales linearly with the invocations of the Lyapunov inequality under a carefully chosen optimization method.

Taking into account the crucial role of the output bound, we believe that we have made a significant contribution to the SNC network analysis. On the other hand, there are still many open challenges in the analysis of larger and more complex networks, e.g., dealing effectively with correlations in the traffic flows, which are left for future work.

References

[1] A. Singla, B. Chandrasekaran, P. B. Godfrey, and B. Maggs, "The internet at the speed of light," in *Proc. ACM Workshop on Hot Topics in Networks'14*, ser. HotNets-XIII, 2014, pp. 1–7.

[2] G. P. Fettweis, "The tactile internet: Applications and challenges," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 64–70, 2014.

[3] A. Whitmore, A. Agarwal, and L. Da Xu, "The internet of things–a survey of topics and trends," *Springer Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.

[4] R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *Proc. ACM Design Automation Conference'10*, 2010, pp. 731–736.

[5] F. Baccelli, G. Cohen, G. J. Olsder, and J.-P. Quadrat, *Synchronization and linearity: an algebra for discrete event systems*. John Wiley & Sons Ltd, 1992.

[6] R. L. Cruz, "A calculus for network delay, part I: Network elements in isolation," *IEEE Transactions on information theory*, vol. 37, no. 1, pp. 114–131, 1991.

[7] ——, "A calculus for network delay, part II: Network analysis," *IEEE Transactions on information theory*, vol. 37, no. 1, pp. 132–141, 1991.

[8] C.-S. Chang, *Performance guarantees in communication networks*. London: Springer-Verlag, 2000.

[9] J.-Y. Le Boudec and P. Thiran, *Network calculus: a theory of deterministic queuing systems for the internet*. New York: Springer-Verlag, 2001.

[10] F. Ciucu, A. Burchard, and J. Liebeherr, "A network service curve approach for the stochastic analysis of networks," in *Proc. ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'05)*, vol. 33, no. 1, 2005, pp. 279–290.

[11] M. Fidler, "An end-to-end probabilistic network calculus with moment generating functions," in *Proc. IEEE IWQoS'06*, Jun. 2006, pp. 261–270.

[12] Y. Jiang and Y. Liu, *Stochastic network calculus*. Springer, 2008, vol. 1.

[13] F. Ciucu and J. Schmitt, "Perspectives on network calculus – no free lunch, but still good value," in *Proc. ACM SIGCOMM'12 Conference*, New York, NY, USA, Aug. 2012, pp. 311–322.

[14] C. Li, A. Burchard, and J. Liebeherr, "A network calculus with effective bandwidth," *IEEE/ACM Transactions on Networking*, vol. 15, no. 6, pp. 1442–1453, 2007.

[15] F. Ciucu, F. Poloczek, and J. Schmitt, "Sharp per-flow delay bounds for bursty arrivals: The case of FIFO, SP, and EDF scheduling," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'14)*, Toronto, Canada, 2014.

[16] A. Rizk, F. Poloczek, and F. Ciucu, "Computable bounds in fork-join queueing systems," in *Proc. ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'15)*, vol. 43, no. 1, 2015, pp. 335–346.

[17] T. Zhu, D. S. Berger, and M. Harchol-Balter, "SNC-meister: Admitting more tenants with tail latency SLOs," in *Proc. ACM Symposium on Cloud Computing (SoCC'16)*, 2016.

[18] M. Fidler and Y. Jiang, "Non-asymptotic delay bounds for (k, l) fork-join systems and multi-stage fork-join networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'16)*, 2016, pp. 1–9.

[19] S. Bondorf, P. Nikolaus, and J. B. Schmitt, "Quality and cost of deterministic network calculus - design and evaluation of an accurate and fast analysis," in *Proc. ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'17)*, 2017.

[20] F. Poloczek and F. Ciucu, "Scheduling analysis with martingales," *Elsevier Performance Evaluation*, vol. 79, pp. 56–72, 2014.

[21] ——, "Service-martingales: Theory and applications to the delay analysis of random access protocols," in *Proc. IEEE Conference on Computer Communications (INFOCOM'15)*, 2015, pp. 945–953.

[22] F. Ciucu, F. Poloczek, and J. Schmitt, "Stochastic upper and lower bounds for general markov fluids," in *Proc. IEEE International Teletraffic Congress (ITC 28'16)*, vol. 1, 2016, pp. 184–192.

[23] J. B. Schmitt, F. A. Zdarsky, and M. Fidler, "Delay bounds under arbitrary multiplexing: When network calculus leaves you in the lurch ..." in *Proc. IEEE International Conference on Computer Communications (INFOCOM'08)*, Phoenix, AZ, USA, Apr. 2008.

[24] M. A. Beck, "Advances in theory and applicability of stochastic network calculus," Ph.D. dissertation, TU Kaiserslautern, 2016.

[25] F. Ciucu, F. Poloczek, and J. B. Schmitt, "Sharp bounds in stochastic network calculus," *CoRR*, vol. abs/1303.4114, 2013. [Online]. Available: http://arxiv.org/abs/1303.4114

[26] R. Nelson, *Probability, stochastic processes, and queueing theory: the mathematics of computer performance modeling*. Springer, 1995.

[27] C. Courcoubetis and R. Weber, "Buffer overflow asymptotics for a buffer handling many traffic sources," *Journal of Applied Probability*, vol. 33, pp. 886–903, 1996.

[28] R. Hooke and T. A. Jeeves, ""Direct Search" Solution of Numerical and Statistical Problems," *Journal of the ACM (JACM)*, vol. 8, no. 2, pp. 212–229, 1961.

APPENDIX

A. Proof of Proposition 11

We have already seen in Subsection III-A that

$$\begin{aligned}&\mathrm{E}\left[e^{\theta A'(s,t)}\right]\\ &\leq \inf_{l\geq 1}\left\{\left(\sum_{i=0}^{s}\mathrm{E}\left[e^{l\theta(A(i,t)-S(i,s))}\right]\right)^{\frac{1}{l}}\right\},\end{aligned}$$

which can be continued with

$$\begin{aligned}&\inf_{l\geq 1}\left\{\left(\sum_{i=0}^{s}\mathrm{E}\left[e^{l\theta(A(i,t)-S(i,s))}\right]\right)^{\frac{1}{l}}\right\}\\ =&\inf_{l\geq 1}\left\{\left(\sum_{i=0}^{s}\mathrm{E}\left[e^{l\theta A(i,t)}\right]\mathrm{E}\left[e^{-l\theta S(i,s)}\right]\right)^{\frac{1}{l}}\right\}\\ \leq&\inf_{l\geq 1}\left\{e^{\theta(\sigma_A(l\theta)+\sigma_S(l\theta))}\right.\\ &\left.\cdot\left(\sum_{i=0}^{s}e^{l\theta(\rho_A(l\theta)(t-i)+\rho_S(l\theta)(s-i))}\right)^{\frac{1}{l}}\right\},\end{aligned}$$

where we, again, used the independence of arrivals and service in the second line and the $(\sigma(\theta),\rho(\theta))$-constraints for arrivals and service in the third line.

Since we assume that $\rho_A(l\theta) < -\rho_S(l\theta)$, we obtain by convergence of the geometric series

$$\begin{aligned}\cdots=&\inf_{l\geq 1}\left\{e^{\theta(\rho_A(l\theta)(t-s)+\sigma_A(l\theta)+\sigma_S(l\theta))}\right.\\ &\left.\cdot\left(\sum_{j=0}^{s}e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))j}\right)^{\frac{1}{l}}\right\}\\ \leq&\inf_{l\geq 1}\left\{e^{\theta(\rho_A(l\theta)(t-s)+\sigma_A(l\theta)+\sigma_S(l\theta))}\right.\\ &\left.\cdot\left(\frac{1}{1-e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))}}\right)^{\frac{1}{l}}\right\}.\end{aligned}$$

This finishes the proof, as this is equal to

$$\begin{aligned}\cdots=&\inf_{l\geq 1}\left\{e^{\theta(\rho_A(l\theta)(t-s)+\sigma_A(l\theta)+\sigma_S(l\theta))}\right.\\ &\left.\cdot e^{\theta\left(-\frac{1}{l\theta}\log\left(1-e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))}\right)\right)}\right\},\end{aligned}$$

which yields

$$\begin{aligned}\sigma_{A'}(\theta)=&\sigma_A(l\theta)+\sigma_S(l\theta)-\frac{1}{l\theta}\log\left(1-e^{l\theta(\rho_A(l\theta)+\rho_S(l\theta))}\right),\\ \rho_{A'}(\theta)=&\rho_A(l\theta)\end{aligned}$$

as the theorem states.