On Per-Flow Delay Bounds in Tandem Queues under (In)Dependent Arrivals

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*—Tandem queues, i.e., several servers in series, occur in many performance models of networked and distributed systems. In this paper, we derive stochastic delay bounds for a flow of interest from a set of flows that traverses the tandem. To that end, we take an approach based on stochastic network calculus (SNC) using moment-generating functions. Directly applying existing SNC approaches from literature requires to take into account many stochastic dependencies unless we incorporate the pay multiplexing only once (PMOO) principle known from deterministic network calculus (DNC). We derive a general solution for the tandem of n servers. Moreover, we also enable the analysis of stochastic dependencies among flows. In numerical experiments with fractional Brownian motion as a traffic model, we compare the delay bounds obtained by our analysis, following the PMOO principle, with those from existing literature and observe an average improvement of the delay bounds by at least 30%. Furthermore, we observe that by careful parameter optimization, a significant improvement is achieved compared to standard choices for the parameter set. Finally, we evaluate the effect of stochastic dependencies among flows on the delay bounds.**

I. Introduction

A. Motivation

Predicting delays in packet-switched networks is a timeless topic. Accordingly, in many visions of the future Internet it plays a major role (e.g., Internet at the speed of light [1], Tactile Internet [2], Internet of Things [3]) as well as in many scenarios for the envisioned cyber-physical systems [4], which often face real-time requirements. The variable and thus hard to predict part of the packet delay is due to the traversal of several queues from sender to receiver, in queueing system jargon so-called tandem queues. In this paper, we investigate a specific tandem queue where a set of flows traverses together the tandem servers as depicted in Figure 1. We are interested in bounding the delay of one particular flow from that set, our flow of interest (foi). While such a tandem queue is, of course, not the most general network setting, it can be found in many scenarios as also evidenced by many existing studies on tandem queues in the queueing theory literature (for instance, a classic result [5] and a more recent one [6]). Moreover, the tandem queue as shown in Figure 1 can also serve as a basic building block in larger networks. In particular, in large multi-tier networks as the Internet this pattern frequently occurs in the backbone of transit providers.

ISBN 978-3-901882-94-4

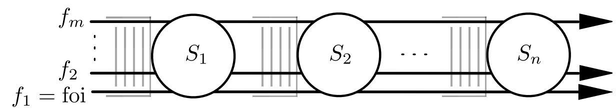


Fig. 1: Tandem with n servers.

B. Analysis Method and State-of-the-Art

To analyze the per-flow delays in tandem queues we employ the mathematical framework of stochastic network calculus (SNC) [7]–[10]. SNC is particularly suited to calculate stochastic per-flow delay bounds, i.e., bounds that hold with a specified probability, whereas conventional queueing theory typically neither supports the per-flow perspective, nor is tailored towards bounds. There are two branches of SNC: one based on moment-generating functions (MGF) bounds for arrival and service processes [7], [8] and one based on tail-bounds [9], [11].

Both branches are versatile with respect to arrival and server models, yet the analysis of more complex networks is still lacking. Existing work mainly treated a topology as depicted in Figure 2. In the tail-bound SNC, one may argue that any network scenario can be transformed into this topology from the perspective of the foi. While this is true and known from deterministic network calculus (DNC) as Separated Flow Analysis (SFA), we demonstrate in this paper that this results in much worse bounds than necessary.

In existing work of the MGF-based SNC for the topology from Figure 2 [8], the transformation from our tandem queue (in Figure 1) into that of Figure 2 would even be impossible because cross-traffic flows would become stochastically dependent, which is against the assumption in [8]. However, in our work we relax this assumption by using the Hölder inequality (when dependencies have to be assumed) and thus enable a stochastic SFA in the MGF-based SNC. As mentioned already, we can do better than that: by adapting what is known from DNC as the pay multiplexing only once (PMOO) principle [12] in the stochastic network analysis, we can achieve much better delay bounds for tandem queues. The trick in the PMOO analysis is to follow the rule of first convolving servers before subtracting cross-flows which is the exact opposite of SFA.

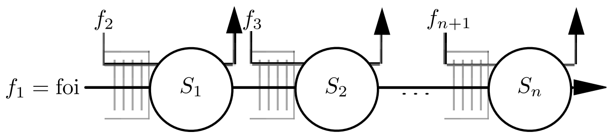


Fig. 2: Tandem in [8].

C. Modeling Assumptions

Apart from assuming a tandem queue as in Figure 1, which is shared by a set of flows, we make no further assumptions on how the multiplexing between flows is done at each of the servers. For the per-flow analysis this means we assume arbitrary multiplexing [13], which in our tandem queue setting corresponds to giving lowest priority to the foi in order not to compromise the delay bound calculation. So, in fact, we could also apply our results in a system where each of the server does priority scheduling.

For the set of arrival flows, the question arises whether we require them to be independent. Here, we want to provide maximum flexibility by not necessarily requiring mutual independence among the flows but also allow for stochastic dependencies. Such dependencies often occur in practice, for instance, if the tandem is just a component of the overall network and some flows have traversed shared queues before they arrive at the ingress of the tandem. In particular, we also support mixed scenarios with some flows being dependent and some being independent. This level of versatility in accommodating dependencies among flows is an original contribution in the framework of the MGF-based SNC. In [14] the accommodation of stochastic dependencies via copulas is investigated in the tail-bound SNC framework. In fact, there are ways to convert between the MGF-based and tail-bound SNC, see, e.g., [15]. However, our approach also differs from [14] in making no assumptions on the kind of stochastic dependencies that are present. Clearly, it could be very interesting to bring together both approaches, thus building open [14] to achieve better bounds in case we have more knowledge about dependencies. Yet, this is out of scope for this paper and is left for future work.

While the end-to-end delay analysis of the tandem queue is for general (MGF-bounded) arrivals, we instantiate our results in numerical experiments for the popular traffic model of fractional Brownian motion (fBm). As a long-range dependent traffic model, fBm is suitable to model the self-similar nature of Internet (backbone) traffic [16], [17]. While not being a simple traffic model (with some basic queueing results just appearing [18]), fBm fits well into our framework as an MGF exists for it and we can obtain some interesting insights in its behavior in non-trivial queueing models.

D. Outline

In Section II, we introduce the necessary notations for SNC and its main results as we need them in this paper. Section III contains the main contributions: the stochastic end-to-end delay analysis for tandem queues following the SFA and PMOO patterns known from DNC. A numerical evaluation of several aspects under the fBm traffic model is provided in Section IV: SFA vs. PMOO, delay bound scaling with tandem length, independent vs. dependent flows, and mixed scenarios. Section V concludes the paper and discusses some future work.

II. SNC Background and Notation

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

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. The connection between probability bounds and MGFs is established by the Chernoff bound $\mathrm{P}(X > a) \leq e^{-\theta a}\,\mathrm{E}\big[e^{\theta X}\big]$, with $\mathrm{E}\big[e^{\theta X}\big]$ as the moment-generating function (MGF) of a random variable X. 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

$$(x \otimes y)(s,t) \coloneqq \min_{s \leq i \leq t} \{x(s,i) + y(i,t)\},$$
$$(x \oslash y)(s,t) \coloneqq \max_{0 \leq i \leq s} \{x(i,t) - y(i,s)\}.$$

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:

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

Definition 3 (Leftover Service)**.** Since we assume arbitrary multiplexing, the worst-case analysis forces us to assume that our foi has the lowest priority at a given dynamic S-server. That is, if flow f_2 is prioritized over flow f_1, the leftover service for the according arrival A_1 is $[S - A_2]^+$.

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

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

The next theorem gives us the required MGF-bound.

Theorem 5 (Output and Delay Bound)**.** *[8] [15] Consider a dynamic S-server $S(s,t)$ with arrival process $A(s,t)$.*

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

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

This provides a bound on the violation probability of a given stochastic delay bound T:

$$\mathrm{P}(d(t) > T) \leq \mathrm{E}\Big[e^{\theta (A \oslash S(t+T,t))} \Big] . \tag{1}$$

Theorem 5 provides us with the start of obtaining stochastic delay bounds, but comes at the price of introducing the deconvolution operators. As it was shown in [8], they can be bypassed by the following inequalities:

$$\mathrm{E}\Big[e^{-\theta (X \otimes Y(s,t))} \Big] \leq \sum_{i=s}^{t} \mathrm{E}\Big[e^{-\theta (X(s,i) + \theta Y(i,t))} \Big] , \tag{2}$$

$$\mathrm{E}\Big[e^{-\theta (X \oslash Y(s,t))} \Big] \leq \sum_{i=0}^{s} \mathrm{E}\Big[e^{\theta (X(i,t) - Y(i,s))} \Big] . \tag{3}$$

As we discuss in detail below, allowing for stochastic dependencies between flows means to have possibly multiple products inside the expectation. We deal with this problem by applying the generalized Hölder inequality.

Theorem 6 (Generalized Hölder Inequality)**.** *[19] Let $X_1, \dots, X_n$ such that $X_i \in L^{p_i}$ be random variables. Then we have*

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

with $\sum_{i=1}^{n} \frac{1}{p_i} = 1$ and $p_i > 0$.

The special case for $n = 2$ and $p_1 = p_2 = 2$ is known as Cauchy-Schwarz inequality.

III. End-to-End Delay Bound Analysis

In this section, we extend the delay bounds from single servers to tandems. For illustrative purposes, starting with the simplest case of a 2-server tandem and stochastically independent flows, we derive the delay bounds for the SFA and PMOO analysis. Then, we generalize the setting to tandems of arbitrary size for independent and finally dependent flows.

A. The 2-Server Tandem

At first, we assume two servers together with two arrival flows (see Figure 3). Throughout this paper, we assume the servers to be work-conserving and mutually independent of the arrivals. Additionally, we assume the arrivals to be independent at first and later on relax this assumption.

a) Separated Flow Analysis (SFA): Here, we compute the delay bound server-by-server (local approach) by subtracting all cross-flows, i.e. computing the left-over service under arbitrary multiplexing. In a final step, all the servers along the tandem are convolved in order to reduce the delay bound calculation to the single-server, single flow case.

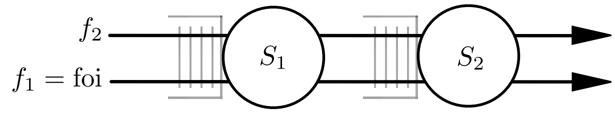


Fig. 3: Tandem with two servers.

b) Pay Multiplexing Only Once Analysis (PMOO): In contrast to the SFA, the PMOO inverts the order of subtraction and convolution. It first convolves all the servers along the tandem and leaves the subtraction of the cross-flows as a last step. In our tandem queues, this can be done perfectly as all servers share the same set of cross-flows, in a general topology this is much more complex [12]. The advantage of PMOO lies in its end-to-end perspective of the tandem (global approach).

In deterministic network calculus (DNC), it has been shown that neither of the analyses dominates the other [13], though in many realistic settings PMOO is superior [12].

Example 7 (Leftover Service Calculation)**.** See Figure 3. Flow f_1 is our foi whereas cross-flow f_2 is considered to be prioritized. For the SFA, in a first step, we subtract the cross flow arrivals, i.e., $[S_1 - A_2]^+$. Next, we subtract the cross-flow at the second server. Here, it is important that we take into account that it traversed the previous server. Consequently, we have to use its output from server 1: $[S_2 - (A_2 \oslash S_1)]^+$, leading to an overall leftover service of $[S_1 - A_2]^+ \otimes [S_2 - (A_2 \oslash S_1)]^+$. At this point, we remark that the cross flow arrivals A_2 appear twice in this formula of the SFA end-to-end leftover service (violation of the PMOO principle).

The PMOO, on the other hand, convolves the servers in the first step, $S_1 \otimes S_2$, (since they share the same cross traffic) and subtracts afterwards: $[(S_1 \otimes S_2) - A_2]^+$. Consequently, we have A_2 only once.

Theorem 5 together with the computation of the leftover service provide us with the tools to compute stochastic per-flow delay bounds. Now, we apply this to the 2-server tandem in Figure 3 according to the SFA and PMOO.

1) Delay Bound with SFA in the 2-server tandem: For the SFA delay bound, we calculate

$$\begin{aligned}
&\mathrm{P}(d(t) > T) \\
&\overset{(1)}{\leq} \mathrm{E}\Big[e^{\theta A_{\mathrm{foi}} \oslash S_{\mathrm{l.o.}}(t+T,t)} \Big] \\
&\overset{(3)}{\leq} \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta \left(A_{\mathrm{foi}}(k_0,t) - S_{\mathrm{l.o.}}(k_0,t+T) \right)} \Big] \\
&= \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta \left(A_1(k_0,t) - \left([S_1 - A_2]^+ \otimes [S_2 - (A_2 \oslash S_1)]^+ \right)(k_0,t+T) \right)} \Big] \\
&\leq \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(k_0,t)} \Big] \\
&\quad \cdot \sum_{k_1=k_0}^{t+T} \mathrm{E}\Big[e^{-\theta \left([S_1(k_0,k_1) - A_2(k_0,k_1)]^+ \right)}
\end{aligned}$$

$$\cdot e^{-\theta\left([S_2(k_1,t+T)-(A_2\oslash S_1)(k_1,t+T)]^+\right)}\Big].$$

Here, we have a stochastic dependence between the two exponentials inside the expectation. Thus, we proceed by applying Hölder's inequality:

$$\begin{aligned}
\cdots \leq & \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \\
& \cdot \sum_{k_1=k_0}^{t+T} \left(\mathrm{E}\Big[e^{-p_1\theta[S_1(k_0,k_1)-A_2(k_0,k_1)]^+}\Big]\right)^{\frac{1}{p_1}} \\
& \cdot \left(\mathrm{E}\Big[e^{-p_2\theta[S_2(k_1,t+T)-(A_2\oslash S_1)(k_1,t+T)]^+}\Big]\right)^{\frac{1}{p_2}} \\
\leq & \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \\
& \cdot \sum_{k_1=k_0}^{t+T} \left(\mathrm{E}\Big[e^{p_1\theta(A_2(k_0,k_1)-S_1(k_0,k_1))}\Big]\right)^{\frac{1}{p_1}} \\
& \cdot \left(\mathrm{E}\Big[e^{p_2\theta((A_2\oslash S_1)(k_1,t+T)-S_2(k_1,t+T))}\Big]\right)^{\frac{1}{p_2}} \\
= & \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \sum_{k_1=k_0}^{t+T} \left(\mathrm{E}\Big[e^{p_1\theta A_2(k_0,k_1)}\Big]^{\frac{1}{p_1}}\right. \\
& \cdot \mathrm{E}\Big[e^{-p_1\theta S_1(k_0,k_1)}\Big]^{\frac{1}{p_1}} \mathrm{E}\Big[e^{p_2\theta(A_2\oslash S_1)(k_1,t+T)}\Big]^{\frac{1}{p_2}} \\
& \left.\cdot \mathrm{E}\Big[e^{-p_2\theta S_2(k_1,t+T)}\Big]^{\frac{1}{p_2}}\right) \\
\leq & \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \sum_{k_1=k_0}^{t+T} \left(\mathrm{E}\Big[e^{p_1\theta A_2(k_0,k_1)}\Big]^{\frac{1}{p_1}}\right. \\
& \cdot \mathrm{E}\Big[e^{-p_2\theta S_1(k_0,k_1)}\Big]^{\frac{1}{p_1}} \\
& \cdot \left(\sum_{k_2=0}^{k_1} \mathrm{E}\Big[e^{p_2\theta A_2(k_2,t+T)}\Big] \mathrm{E}\Big[e^{-p_2\theta S_1(k_2,k_1)}\Big]\right)^{\frac{1}{p_2}} \\
& \left.\cdot \mathrm{E}\Big[e^{-p_2\theta S_2(k_1,t+T)}\Big]^{\frac{1}{p_2}}\right),
\end{aligned}$$

with

$$\frac{1}{p_1}+\frac{1}{p_2}=1.$$

2) Delay Bound with PMOO in the 2-server tandem: From the PMOO delay bound we obtain as follows

$$\begin{aligned}
& \mathrm{P}(d(t)>T) \\
\overset{(1)}{\leq} & \mathrm{E}\Big[e^{\theta A_{\mathrm{foi}}\oslash S_{\mathrm{l.o.}}(t+T,t)}\Big] \\
\overset{(3)}{\leq} & \sum_{k_0=0}^{t} \mathrm{E}\Big[e^{\theta\left(A_1(k_0,t)-[S_1\otimes S_2(k_0,t+T)-A_2(k_0,t+T)]^+\right)}\Big] \\
\leq & \sum_{k_0=0}^{t} \left(\mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big]\right. \\
& \left.\cdot \mathrm{E}\Big[e^{\theta(A_2(k_0,t+T)-S_1\otimes S_2(k_0,t+T))}\Big]\right) \\
= & \sum_{k_0=0}^{t} \left(\mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \mathrm{E}\Big[e^{\theta A_2(k_0,t+T)}\Big]\right. \\
& \left.\cdot \sum_{k_1=k_0}^{t+T} \mathrm{E}\Big[e^{-\theta S_1(k_0,k_1)}\Big] \mathrm{E}\Big[e^{-\theta S_2(k_1,t+T)}\Big]\right).
\end{aligned}$$

At this point, we observe that, even though we analyzed the same network, only the SFA has to apply Hölder's inequality. Generally, the usage of Hölder's inequality should be minimized and thus this must be seen as a shortcoming of the SFA. It forces us to apply Hölder's inequality more often than for the PMOO. This is even more evident if we increase the number of servers as we see in the general tandem case, next.

B. The General Tandem

This subsection extends the 2-server tandem to the general case of n servers and m flows (as in Figure 1).

Proposition 8 (Delay Bound with SFA)**.** *With the SFA, we obtain for the delay bound for m arrival flows and n servers*

$$\begin{aligned}
& \mathrm{P}(d(t)>T) \\
\leq & \mathrm{E}\Big[e^{\theta(A_{\mathrm{foi}}\oslash S_{\mathrm{l.o.}}(s,t))}\Big] \\
\leq & \sum_{k_0=0}^{t} \left(\mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \sum_{k_0\leq k_1\leq t+T} \cdots\right. \\
& \sum_{k_{n-2}\leq k_{n-1}\leq t+T} \mathrm{E}\Big[e^{p_1\theta\sum_{j=2}^{m}A_j(k_0,k_1)}e^{-p_1\theta S_1(k_0,k_1)}\Big]^{\frac{1}{p_1}} \\
& \cdots \mathrm{E}\Big[e^{p_n\theta\left(\left(\left(\left(\sum_{j=2}^{m}A_j\right)\oslash S_1\right)\ldots\right)\oslash S_{n-1}\right)(k_{n-1},t+T)} \\
& \left.\cdots e^{-p_n\theta S_n(k_{n-1},t+T)}\Big]^{\frac{1}{p_n}}\right),
\end{aligned}$$

where

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

On the contrary, the PMOO is able to circumvent the necessity to take into account a large number of dependencies:

Proposition 9 (Delay Bound with PMOO)**.** *The PMOO yields for the delay bound in the n-server tandem with m flows*

$$\begin{aligned}
& \mathrm{P}(d(t)>T) \\
\leq & \mathrm{E}\Big[e^{\theta(A_{\mathrm{foi}}\oslash S_{\mathrm{l.o.}}(t+T,t))}\Big] \\
\leq & \sum_{k_0=0}^{t} \left(\mathrm{E}\Big[e^{\theta A_1(k_0,t)}\Big] \prod_{j=2}^{m} \mathrm{E}\Big[e^{\theta A_j(k_0,t+T)}\Big]\right. \\
& \cdot \sum_{k_0\leq k_1\leq t+T} \cdots \sum_{k_{n-2}\leq k_{n-1}\leq t+T} \left(\mathrm{E}\Big[e^{-\theta S_1(k_0,k_1)}\Big]\right. \\
& \left.\left.\cdots \mathrm{E}\Big[e^{-\theta S_n(k_{n-1},t+T)}\Big]\right)\right).
\end{aligned}$$

C. Dependent Arrival Flows

So far we have assumed the arrivals to be independent. If we allow the flows to be dependent, the PMOO is also not able to circumvent the application of Hölder's inequality any more. Again, we compute the delay bound in the n-server tandem (Figure 1).

Proposition 10 (Delay Bound with PMOO and Dependent Flows)**.** *If all arrival flows are dependent, the PMOO yields in the n-server tandem with m flows:*

$$\begin{aligned}
&\mathrm{P}(d(t) > T) \\
&\leq \sum_{k_0=0}^{t} \Bigg(\mathrm{E}\Big[e^{p_1 \theta A_1(k_0,t)}\Big]^{\frac{1}{p_1}} \\
&\quad \cdot \left(\prod_{j=2}^{m} \mathrm{E}\Big[e^{p_2 p_{j+1} \theta A_j(k_0,t+T)}\Big]^{\frac{1}{p_{j+1}}} \right. \\
&\quad \cdot \sum_{k_1=k_0}^{t+T} \cdots \sum_{k_{n-1}=k_{n-2}}^{t+T} \Big(\mathrm{E}\Big[e^{-p_2 \theta S_1(k_0,k_1)}\Big] \\
&\quad \left. \cdots \mathrm{E}\Big[e^{-p_2 \theta S_n(k_{n-1},t+T)}\Big] \Big) \right)^{\frac{1}{p_2}} \Bigg),
\end{aligned}$$

where

$$\frac{1}{p_1} + \frac{1}{p_2} = 1,$$
$$\frac{1}{p_3} + \cdots + \frac{1}{p_{m+1}} = 1.$$

IV. Evaluation

In this section, we evaluate several aspects of the per-flow delay bounds in tandem queues by means of numerical experiments. We compare the performance of the different analysis techniques (SFA vs. PMOO), investigate the scaling behavior of the delay bounds as we increase the tandem lengths, and study the effect of stochastic dependencies. To that end, we have implemented the delay bound computation based on the equations from Section III using the general-purpose programming language **Python**[1], version 3.6.

For the arrival flows we choose the fractional Brownian motion (fBm) traffic model [17]. As discussed above, it is both a popular LRD model for Internet traffic and is amenable to our analysis as the MGF exists.

Definition 11 (Fractional Brownian Motion)**.** A stochastic process F_t is called *fractional Brownian motion*, if its MGF has the form

$$\mathrm{E}\big[e^{\theta F_t}\big] = e^{\lambda \theta t + \frac{(\sigma\theta)^2}{2} t^{2H}},$$

with drift $\lambda \geq 0$, variance at $t = 1$ equal to σ^2, i.e., $\mathrm{var}(F_t) = \sigma^2 t^{2H}$, and Hurst parameter $H \in (0, 1)$.

[1] https://www.python.org

Throughout the following experiments, the parameters in the fBm arrival model are always chosen to be $\lambda = 0.5$ (the average rate of the flow), $\sigma = 1$ (a burstiness parameter) and Hurst parameter $H = 0.7$ (degree of long-range dependence).

For the sake of simplicity, the number of flows traversing the tandem queues is always set equal to the number of servers. The servers are constant-rate and work-conserving. If not stated otherwise, we allocate the service rate to be $3 \cdot (\text{number of flows})$.

As we also want to investigate the effect of optimization in some of the experiments, we compare a thorough systematic search through the parameter space with fixed "natural" choices for the parameters. More specifically, in the unoptimized calculations, we set $\theta = 1$ for all MGFs involved, and for the Hölder parameters we set $\forall i : p_i = c$. The latter corresponds to using the Cauchy-Schwarz inequality (for $n = 2$). For example, if the Hölder parameters have to satisfy

$$\frac{1}{p_1} + \frac{1}{p_2} = 1$$
$$\frac{1}{p_3} + \frac{1}{p_4} + \frac{1}{p_5} = 1,$$

we set $p_1 = p_2 = 2$ and $p_3 = p_4 = p_5 = 3$.

A. Comparison between SFA and PMOO

In the first experiment, we compare the two analysis techniques, SFA and PMOO, for the case of two servers and independent flows. The delay bounds for both methods with and without optimized parameters are shown in Figure 4.

Obviously, there is a huge improvement of PMOO over SFA. Looking, for instance, at the optimized versions, there is about 3-5 orders of magnitude of improvement in terms of violation probabilities (increasing in the delay bound T for which it is calculated). In terms of delay bounds, the improvement is, of course, less spectacular but still very considerable, e.g., the violation probability of the SFA for $T = 6$ is slightly above the one of PMOO for $T = 4$. That means the delay bound improvement is, roughly speaking, more than 30% in this scenario. So, already in this short tandem, adhering to the PMOO principle pays off strongly. This is because much less dependencies have to be accommodated for in the PMOO analysis than in the SFA, thus avoiding the obviously "expensive" application of the Hölder inequality as much as possible.

Regarding the unoptimized delay bounds in that experiment, we can observe that the optimization plays a crucial role. For example, for PMOO the optimized violation probabilities are 3-4 orders of magnitude better than those for the "natural" choice (though these are still better than the optimized SFA). As more and more parameters are involved in more complex scenarios, this indicates to invest potentially high computational efforts to enable a thorough optimization instead of searching only coarsely.

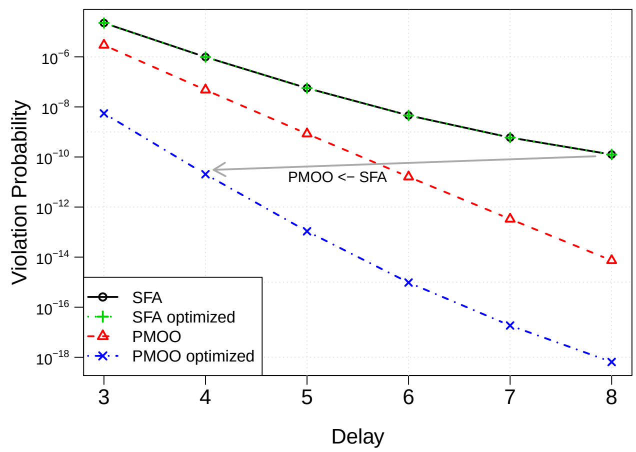


Fig. 4: Delay bound comparison for two servers of SFA and PMOO.

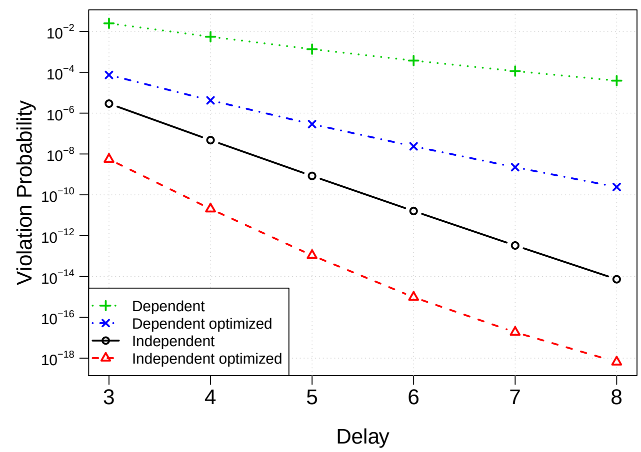


Fig. 6: Delay bound comparison for two servers of PMOO for independent and dependent arrivals for the standard and optimized parameter choice.

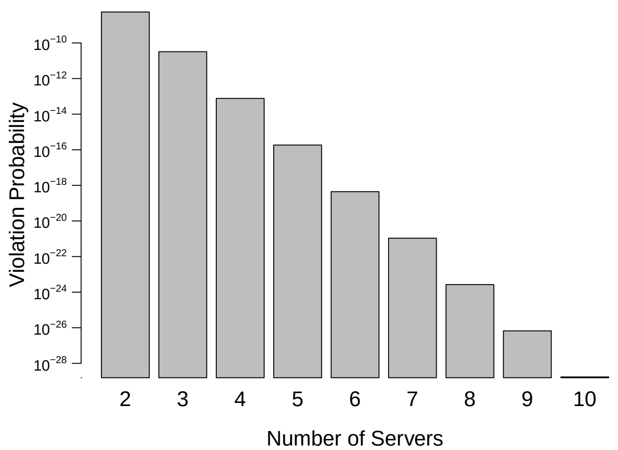


Fig. 5: PMOO delay bounds for different tandem lengths.

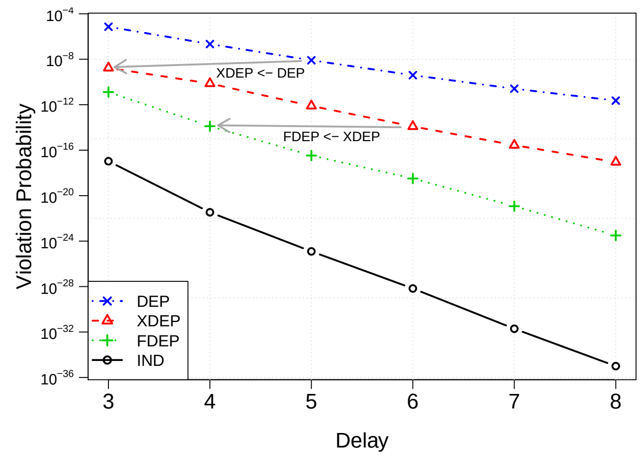


Fig. 7: Delay bound comparison for three servers with increasing degree of dependence.

B. Different Tandem Lengths

In the next experiment, we focus on the (optimized) PMOO and how its delay bounds scale with the tandem lengths and the number of flows, respectively. To that end, we calculate the violation probabilities for 2 to n servers in the tandem and a corresponding number of independent flows (for $T = 3$). The results are shown in Figure 5.

At first sight, the observation that the violation probabilities decrease with longer tandems may be surprising. Yet, since the number of flows increases as well, obviously the degradation effect of longer tandems is by far outweighed by the statistical multiplexing effect for the independent fBm flows. Roughly speaking, for each additional server we gain 2 orders of magnitude for the violation probability.

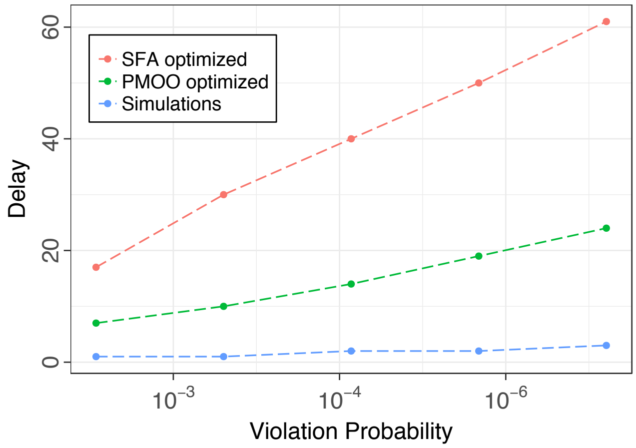
C. Independent vs. Dependent Flows

Next, we turn to the case of dependent flows and evaluate the penalty we obtain in comparison to the independent case (results from the experiment in Section IV-A). Hence, we run another numerical experiment with the PMOO for dependent flows for two servers. We also look again at the role of optimization. The violation probabilities of different delay bounds are shown in Figure 6.

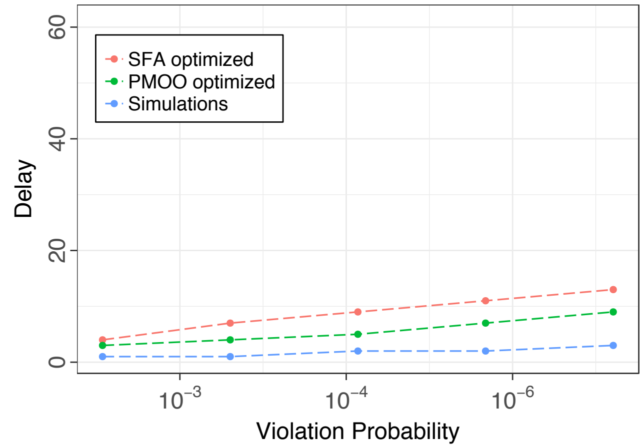
Expectedly, we observe a high penalty for our analytical treatment of dependent flows (using Hölder's inequality), although there are only two flows. Consequently, the optimization plays maybe an even more important role for dependent flows in order to bring them down to acceptable violation probabilities (assuming we are interested in the typical range of large deviations).

D. Mixed Independence / Dependence Scenarios

In the previous experiments, we have seen that dependencies can be very detrimental for the delay bounds. In this last experiment, we now investigate a mixed scenario with some flows being dependent and others being independent. More



(a) Fractional Brownian motion.



(b) Exponential distribution.

Fig. 8: Comparison with simulations for two servers and two arrival flows.

specifically, we use 3 servers in the tandem and 3 flows correspondingly. We evaluate the (optimized) PMOO delay bounds for four scenarios:

1) (IND) all flows are independent;
2) (DEP) all flows are dependent;
3) (XDEP) the cross-flows are dependent, but the foi is independent of them;
4) (FDEP) the foi is dependent with one of the cross-flows, but not with the other one, the cross-flows are independent of each other.

The analytical derivation of the new scenarios XDEP and FDEP can be found in the appendix.

In this special setting, we deviate from our usual choice and set the service to $r = 11$. This was necessary to achieve meaningful bounds in the dependent case. The results are depicted in Figure 7.

As expected, the cases with partial dependencies appear in the gap between full dependence and independence, respectively. XDEP's violation probability is three orders of magnitude better than the full dependence for a delay bound equal to 3. This gap increases over time. In the delay space, this leads to an improvement of about 40%. Switching the dependence from dependent cross flows to the dependence of the foi and one cross flow, we gain another approx. 30%. The independent case, on the other hand, outperforms the FDEP by far, especially for larger delays.

The results suggest again that dependence leads to a high penalty in the analysis. If, however, partial independence can be assumed, making use of this property yields significantly better bounds. As we observe, not only the number of dependent flows is important, but also their relation. Apparently, having a dependence among the cross flows is worse for our analysis than between the foi and one of the cross flows instead.

E. Comparison to Simulations

So far, we have compared different SNC analyses against each other. Clearly, it is also interesting how close the bounds capture actual system behavior. In order to assess this, simulations are obviously a good candidate. However, simulations of the previously described scenarios would, due to very low violation probabilities as, e.g., 10^{-20}, lead to prohibitively long runtimes. Furthermore, due to the very general treatment of dependencies, in our method it is difficult to instantiate such dependent traffic in an unbiased manner.

Therefore, we set up a new scenario in which we simulate fBm traffic (parameters chosen as in the beginning of Section IV) for two independent arrival flows that enter a tandem in parallel and observe the delay of one of them. Here, the service rate is reduced to $r = 2.9$ for both servers in order to allow for higher violation probabilities and correspondingly feasible simulation times. We compare the empirical delay quantiles of the simulations to the corresponding delay bounds obtained by the PMOO and SFA methods, respectively, in Figure 8a.

We observe that, for given empirical delay quantiles, simulations have a significantly lower empirical delay than the SFA and PMOO bounds, but with the latter being much closer to the simulations, emphasizing again PMOO's superiority. Nevertheless, the gap between simulations and the PMOO bounds is admittedly still considerable. However, this inaccuracy for traffic with high correlations when using standard SNC methods (in particular, the union bound) is known in the literature [20]. Although it had not yet been shown explicitly for fBm traffic, it is thus not surprising.

To investigate this further, we performed the same experiment but now we used traffic with exponentially distributed independent increments (with parameter $\lambda = 1.8$). The experiment's outcome is shown in Figure 8b and should be compared to the fBm traffic results in Figure 8a. Now, the bounds are much closer to the simulated empirical delay quantiles, with the PMOO performing superior over SFA again, though somewhat less pronounced.

V. Conclusion

In this paper, we have derived stochastic delay bounds in tandem queues using stochastic network calculus (SNC). To that end, we transferred known network analysis techniques from the deterministic network calculus to its stochastic counterpart, to be precise, to the MGF-based SNC. We showed that applying the pay multiplexing only once (PMOO) principle was key to avoid taking into account too many stochastic dependencies which, in turn, ruin the bounds' quality. In fact, in the MGF-based SNC the principle should rather be renamed into "pay stochastic dependencies only if you must". Besides this major insight, we also showed how to deal with dependencies among the flows or subsets of the flows and evaluated the effects on the delay bounds. And, last but not least, we observed in numerical experiments the important role of parameter optimization, as, in particular, the PMOO analysis showed a high sensitivity to a good parameter choice.

Overall, we perceive this paper as a promising first step into the challenging field of more complex network analysis using SNC. Clearly, future work will have to address more general topologies like feedforward networks or even under cyclic flow dependencies. As can be observed from our results, it will be crucial to smartly deal with stochastic dependencies both on the network analysis level (similar to our work) and the level of inequalities finding, e.g., better tools than Hölder's inequality, possibly leveraging approaches as in [14].

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] P. Burke, "The dependence of delays in tandem queues," *The Annals of Mathematical Statistics*, vol. 35, no. 2, pp. 874–875, 1964.

[6] L. Le and E. Hossain, "Tandem queue models with applications to qos routing in multihop wireless networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 8, pp. 1025–1040, 2008.

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

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

[9] Y. Jiang and Y. Liu, *Stochastic network calculus*. Springer, 2008.

[10] F. Ciucu and J. Schmitt, "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)*, New York, NY, USA, Aug. 2012, pp. 311–322.

[11] 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, 2003.

[12] J. Schmitt, F. A. Zdarsky, and I. Martinovic, "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)*, Mar. 2008.

[13] 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.

[14] F. Dong, K. Wu, and V. Srinivasan, "Copula analysis for statistical network calculus," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'15)*, 2015, pp. 1535–1543.

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

[16] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of ethernet traffic," *IEEE/ACM Transactions on networking (ToN)*, vol. 2, no. 1, pp. 1–15, 1994.

[17] I. Norros, "On the use of fractional brownian motion in the theory of connectionless networks," *IEEE Journal on selected areas in communications*, vol. 13, no. 6, pp. 953–962, 1995.

[18] J. Chen, H. S. Bhatia, R. G. Addie, and M. Zukerman, "Statistical characteristics of queue with fractional brownian motion input," *IET Electronics Letters*, vol. 51, no. 9, pp. 699–701, 2015.

[19] W.-S. Cheung, "Generalizations of Hölder's inequality," *International Journal of Mathematics and Mathematical Sciences*, vol. 26, no. 1, pp. 7–10, 2001.

[20] F. Ciucu, F. Poloczek, and J. B. Schmitt, "Sharp bounds in stochastic network calculus," in *Proc. ACM SIGMETRICS'13*, vol. 41, Jun. 2013, pp. 367–368.

Appendix

a) XDEP Scenario: In the XDEP setting described in Subsection IV-D, we obtain for the leftover service:

$$\begin{aligned}
&\mathrm{E}\left[e^{-\theta(S_{\mathrm{l.o.}}(k_0,t+T))}\right]\\
&=\mathrm{E}\left[e^{-\theta\left([S_1\otimes S_2\otimes S_3(k_0,t+T)-(A_2+A_3)(k_0,t+T)]^+\right)}\right]\\
&\leq \mathrm{E}\left[e^{\theta(A_2+A_3)(k_0,t+T)}\right]\\
&\quad\cdot \mathrm{E}\left[e^{-\theta(S_1\otimes S_2\otimes S_3(k_0,t+T))}\right]\\
&\overset{(\text{Hölder})}{\leq} \mathrm{E}\left[e^{p_1\theta A_2(k_0,t+T)}\right]^{\frac{1}{p_1}} \mathrm{E}\left[e^{p_2\theta A_3(k_0,t+T)}\right]^{\frac{1}{p_2}}\\
&\quad\cdot \sum_{k_1=k_0}^{t+T}\sum_{k_2=k_1}^{t+T}\mathrm{E}\left[e^{-\theta(S_1(k_0,k_1)+S_2(k_1,k_2)+S_3(k_2,t+T))}\right]\\
&=\mathrm{E}\left[e^{p_1\theta A_2(k_0,t+T)}\right]^{\frac{1}{p_1}} \mathrm{E}\left[e^{p_2\theta A_3(k_0,t+T)}\right]^{\frac{1}{p_2}}\\
&\quad\cdot \sum_{k_1=k_0}^{t+T}\sum_{k_2=k_1}^{t+T}\mathrm{E}\left[e^{-\theta S_1(k_0,k_1)}\right]\\
&\quad\cdot \mathrm{E}\left[e^{-\theta S_2(k_1,k_2)}\right]\mathrm{E}\left[e^{-\theta S_3(k_2,t+T)}\right],
\end{aligned}$$

where we applied Hölder's inequality owing to the cross flows' mutual dependence. Using this result, the delay bound yields

$$\begin{aligned}
&\mathrm{P}(d(t)>T)\\
&\leq \sum_{k_0=0}^{t}\mathrm{E}\left[e^{\theta(A_{\mathrm{foi}}(k_0,t)-S_{\mathrm{l.o.}}(k_0,t+T))}\right]\\
&=\sum_{k_0=0}^{t}\mathrm{E}\left[e^{\theta A_{\mathrm{foi}}(k_0,t)}\right]\mathrm{E}\left[e^{-\theta S_{\mathrm{l.o.}}(k_0,t+T)}\right]\\
&\leq \sum_{k_0=0}^{t}\mathrm{E}\left[e^{\theta A_1(k_0,t)}\right]\mathrm{E}\left[e^{p_1\theta A_2(k_0,t+T)}\right]^{\frac{1}{p_1}}
\end{aligned}$$

$$\cdot\,\mathrm{E}\Big[e^{p_2\theta A_3(k_0,t+T)}\Big]^{\frac{1}{p_2}}\sum_{k_1=k_0}^{t+T}\sum_{k_2=k_1}^{t+T}\mathrm{E}\Big[e^{-\theta S_1(k_0,k_1)}\Big]$$
$$\cdot\,\mathrm{E}\Big[e^{-\theta S_2(k_1,k_2)}\Big]\,\mathrm{E}\Big[e^{-\theta S_3(k_2,t+T)}\Big],$$

with

$$\frac{1}{p_1}+\frac{1}{p_2}=1.$$

b) FDEP Scenario: On the other hand, we calculate for the delay bound in the FDEP case (only A_{foi} and A_2 are dependent)

$$\begin{aligned}
&\mathrm{P}(d(t)>T)\\
&\leq \sum_{k_0=0}^{t}\mathrm{E}\Big[e^{\theta(A_{\text{foi}}(k_0,t)-S_{\text{l.o.}}(k_0,t+T))}\Big]\\
&=\sum_{k_0=0}^{t}\mathrm{E}\Big[e^{\theta A_1(k_0,t)}\\
&\quad\cdot e^{-\theta[S_1\otimes S_2\otimes S_3(k_0,t+T)-(A_2+A_3)(k_0,t+T)]^+}\Big]\\
&\leq \sum_{k_0=0}^{t}\mathrm{E}\Big[\Big[e^{\theta A_1(k_0,t)}\Big]\\
&\quad\cdot e^{\theta((A_2+A_3)(k_0,t+T)-S_1\otimes S_2\otimes S_3(k_0,t+T))}\Big]\\
&=\sum_{k_0=0}^{t}\mathrm{E}\Big[e^{\theta(A_1(k_0,t)+A_2(k_0,t+T)+A_3(k_0,t+T))}\Big]\\
&\quad\cdot\,\mathrm{E}\Big[e^{\theta(-S_1\otimes S_2\otimes S_3(k_0,t+T))}\Big]\\
&\overset{\text{(Hölder)}}{\leq}\sum_{k_0=0}^{t}\mathrm{E}\Big[e^{p_1\theta A_1(k_0,t)}\Big]^{\frac{1}{p_1}}\mathrm{E}\Big[e^{p_2\theta A_2(k_0,t+T)}\Big]^{\frac{1}{p_2}}\\
&\quad\cdot\,\mathrm{E}\Big[e^{\theta A_3(k_0,t+T)}\Big]\\
&\quad\cdot\sum_{k_1=k_0}^{t+T}\sum_{k_2=k_1}^{t+T}\mathrm{E}\Big[e^{-\theta(S_1(k_0,k_1)+S_2(k_1,k_2)+S_3(k_2,t+T))}\Big]\\
&=\sum_{k_0=0}^{t}\mathrm{E}\Big[e^{p_1\theta A_1(k_0,t)}\Big]^{\frac{1}{p_1}}\mathrm{E}\Big[e^{p_2\theta A_2(k_0,t+T)}\Big]^{\frac{1}{p_2}}\\
&\quad\cdot\,\mathrm{E}\Big[e^{\theta A_3(k_0,t+T)}\Big]\\
&\quad\cdot\sum_{k_1=k_0}^{t+T}\sum_{k_2=k_1}^{t+T}\mathrm{E}\Big[e^{-\theta S_1(k_0,k_1)}\Big]\,\mathrm{E}\Big[e^{-\theta S_2(k_1,k_2)}\Big]\\
&\quad\cdot\,\mathrm{E}\Big[e^{-\theta S_3(k_2,t+T)}\Big],
\end{aligned}$$

with

$$\frac{1}{p_1}+\frac{1}{p_2}=1.$$