Perspectives on Network Calculus – No Free Lunch, but Still Good Value

Florin Ciucu
T-Labs / TU Berlin

Jens Schmitt
University of Kaiserslautern

ABSTRACT

ACM Sigcomm 2006 published a paper [26] which was perceived to unify the deterministic and stochastic branches of the network calculus (abbreviated throughout as DNC and SNC) [39]. Unfortunately, this seemingly fundamental unification—which has raised the hope of a straightforward transfer of all results from DNC to SNC—is invalid. To substantiate this claim, we demonstrate that for the class of stationary and ergodic processes, which is prevalent in traffic modelling, the probabilistic arrival model from [26] is quasi-deterministic, i.e., the underlying probabilities are either zero or one. Thus, the probabilistic framework from [26] is unable to account for statistical multiplexing gain, which is in fact the raison d'être of packet-switched networks. Other previous formulations of SNC can capture statistical multiplexing gain, yet require additional assumptions [12, 22] or are more involved [14, 9, 28], and do not allow for a straightforward transfer of results from DNC. So, in essence, there is no free lunch in this endeavor.

Our intention in this paper is to go beyond presenting a negative result by providing a comprehensive perspective on network calculus. To that end, we attempt to illustrate the fundamental concepts and features of network calculus in a systematic way, and also to rigorously clarify some key facts as well as misconceptions. We touch in particular on the relationship between linear systems, classical queueing theory, and network calculus, and on the lingering issue of tightness of network calculus bounds. We give a rigorous result illustrating that the statistical multiplexing gain scales as $\Omega(\sqrt{N})$, as long as some small violations of system performance constraints are tolerable. This demonstrates that the network calculus can capture actual system behavior tightly when applied carefully. Thus, we positively conclude that it still holds promise as a valuable systematic methodology for the performance analysis of computer and communication systems, though the unification of DNC and SNC remains an open, yet quite elusive task.

SIGCOMM'12, August 13–17, 2012, Helsinki, Finland.

Categories and Subject Descriptors

C.4 [**Computer Systems Organization**]: Performance of Systems—*Modeling techniques*

General Terms

Theory, Performance

Keywords

Network Calculus, Statistical Multiplexing Gain

1. INTRODUCTION

Queueing theory is an important theory for the performance analysis of resource sharing systems such as communication networks. One of the success stories of queueing theory is Erlang's formula for the computation of the blocking probability that some shared resource is occupied [20]; this formula has been used for nearly a century to dimension telephone networks. Concomitantly, queueing theory has been generalized from Erlang's primordial single queue model with Poisson arrivals and exponential service times to the class of *product-form queueing networks* which can account for multiple service time distributions, scheduling, or routing (e.g., [3, 29]).

Notwithstanding the advances made in the classical branch of queueing theory [25], which is primarily concerned with exact models and solutions, the class of tractable queueing networks is largely constrained by the technical assumption of Poisson arrivals. This apparent limitation has motivated the development of alternative theories to queueing, especially over the past three decades witnessing a rapid growth of high-speed data networks. The relevance of the emerging theories, especially for the Internet community, has become evident with the discovery that Internet traffic is fundamentally different from Poisson [32, 40]. Moreover, as it became clear that improper traffic models can lead to bogus results, the necessity to overcome the Poisson assumption limitation has reached a wide consensus.

One of the alternatives to the classical queueing theory is the network calculus. This was conceived by Cruz [16] in the early 1990s in a deterministic framework, and extended soon after by Chang [11], Kurose [30], and Yaron and Sidi [49] in a probabilistic or stochastic framework. Subsequently, many researchers have contributed to both formulations of the network calculus (see the books of Chang [12], Le Boudec and Thiran [5], and Jiang [28]). While DNC was motivated by the need for a theory to compute deterministic (worst-case) bounds on performance metrics, the

raison d'être of SNC was to additionally capture statistical multiplexing gain when some violations of the deterministic bounds are tolerable. This feature enables a much more efficient dimensioning of resource sharing systems, such as packet-switched networks, and continues to play a pivotal role in the evolution of SNC.

The promise of the combined branches of network calculus is to jointly overcome the technical barriers of queueing networks on all fronts. Achieving this rather daunting task is enabled by two key features:

- *Scheduling abstraction.* At a single queue with multiplexed arrival flows, the specific properties of many scheduling algorithms, and also of many arrival classes, can be abstracted away by suitably constructing the so-called *service processes* (technical details are deferred to Section 3).
- *Convolution-form networks.* The service processes from single queues can be *convolved* across a network of queues, and thus a multi-node network analysis can be drastically simplified by reducing it to a single-node analysis.

Equipped with these two features, the network calculus can analyze many scheduling algorithms and arrival classes, over a multi-node network, in a *uniform manner*. That means that the Poisson model, in particular, plays no special role anymore in facilitating the analytical tractability of a whole network. Compared to classical queueing theory, which separately analyzes various combinations of arrivals and scheduling, network calculus conceivably offers a much more simplified and uniform framework. For this reason, the network calculus has been applied in many recent areas such as IntServ [6], switched Ethernets [44], systems-on-chip [10], avionic networks [41], the smart grid [48], etc.

This versatile applicability, however, is only possible at the expense of providing *bounds* on performance metrics. The bounds are a manifestation of resorting to inequalities, whenever exact derivations become intractable. The tightness of the bounds is certainly a major concern, since loose bounds may be more misleading than wrongly fitted Poisson models. The tightness issue has several dimensions depending on the nature of the bounds (deterministic or probabilistic) or the number of flows/queues. Deterministic bounds are generally tight for single queues ([5], p. 27), but can be very loose in some queueing networks with arbitrary multiplexing [43]. Moreover, the deterministic bounds can be very inefficient for network dimensioning when some violation probabilities are tolerable (e.g., running IntServ for many flows could result in very low network utilization). Probabilistic bounds are generally asymptotically tight (in terms of scaling laws in the number of queues) [8, 34], whereas numerical tightness ranges from reasonable [13] to quite loose [34], depending on the arrival model.

In this paper we touch on the lingering issue of tightness, as part of a broader perspective on network calculus. Concretely, we address the asymptotic tightness of probabilistic bounds, in the number of flows N, and demonstrate that such bounds improve upon corresponding deterministic bounds by a factor of $\Omega\left(\sqrt{N}\right)$. That means that, e.g., implementing a probabilistic extension of IntServ could significantly increase the network utilization. Our result not only rigorously reveals the magnitude of the statistical multiplexing gain achieved with SNC, but clearly highlights the fundamental advantage of SNC over DNC.

Our broader goal is to deliver an intuitive and yet comprehensive perspective of the two core concepts in network calculus, i.e., service and envelope processes, by focusing on subtleties and raising awareness of inherent pitfalls. Along the discussion we attempt to make a suggestive statement that there is no free lunch in the framework of the network calculus, yet it brings good value as a companion/alternative to the classical queueing theory. This perspective is motivated by a large effort in the literature to develop SNC formulations which reproduce in particular the 'convolution-form networks' feature from DNC. Arguably the simplest of such formulations has appeared in a Sigcomm 2006 paper [26], and has since raised the hope that DNC results can be transferred into SNC in a straightforward manner. Unfortunately, the formulation from [26] is based on a quasi-deterministic arrival model[1], which roughly means that the proposed SNC cannot capture statistical multiplexing gain. We believe that exposing this pitfall, through a rigorous analysis, is essential to the comprehensive understanding of SNC arrival models.

After introducing notations, the rest of the paper is structured as follows. In Section 3 we provide a comprehensive perspective on service processes by making a multilateral analogy of network calculus with linear systems and classical queueing theory. In Section 4 we present representative envelope processes and elaborate on the quasi-deterministic aspects of the one from [26]. In Section 5 we lead together envelope and service processes in order to shed some light on the often raised concern about the tightness of network calculus bounds. We conclude the paper in Section 6.

2. NOTATIONS

The time model is discrete starting from zero. The time indices are denoted by the symbols i, k, n. The *cumulative* arrivals and departures at/from a (queueing) node up to time n are denoted by non-decreasing processes $A(n)$ and $D(n)$. The doubly-indexed extensions are $A(k,n) = A(n) - A(k)$ and $D(k,n) = D(n) - D(k)$. The associated *instantaneous* arrival and departure processes are $a_n = A(n-1,n)$ and $d_n = D(n-1,n)$, respectively; by convention, $a_0 = d_0 = A(0) = D(0) = 0$. The vector representations are $\mathbf{A} = (A(0), A(1), \dots)$ and $\mathbf{a} = (a_0, a_1, \dots)$ for the arrivals, and $\mathbf{D} = (D(0), D(1), \dots)$ and $\mathbf{d} = (d_0, d_1, \dots)$ for the departures. These processes have primarily a spatial interpretation, e.g., a_n quantifies the number of data units (referred to as bits throughout) arrived at time n; with abuse of notation, $\mathbf{a}$ and $\mathbf{d}$ will also have a temporal meaning to be made locally clear.

The sets of natural, integer, real, and positive real numbers are denoted by $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$, and $\mathbb{R}_+$, respectively; their restriction to non-zero numbers are denoted by $\mathbb{N}^*$, $\mathbb{Z}^*$, $\mathbb{R}^*$, and $\mathbb{R}_+^*$. The integer part of a number $x \in \mathbb{R}$ is denoted by $\lfloor x \rfloor$. For $x \in \mathbb{R}$, the positive part is denoted by $[x]_+ = \max\{x, 0\}$. For some boolean expression E, the indicator function is denoted by 1_E and takes the values 1 or 0 depending whether E is true or false, respectively.

[1]The authors of [9] also point out the quasi-determinism issue in [26], but for a service model and without proof; as a further side remark, we have ourselves experienced a similar quasi-determinism pitfall in SNC [42].

For two functions $f, g : \mathbb{N} \to \mathbb{R}$, the $(\min, +)$ convolution operator '$*$' is defined as

$$f * g(n) := \min_{0 \le k \le n} \{f(k) + g(n-k)\} \ \forall n \ge 0 \ .$$

If the function g is bivariate, i.e., $g : \mathbb{N} \times \mathbb{N} \to \mathbb{R}$, then $f * g(n) := \min_{0 \le k \le n} \{f(k) + g(k, n)\} \ \forall n \ge 0$.

3. SERVICE PROCESSES

Network calculus operates by reducing a 'complex' nonlinear (queueing) system into a *'somewhat looking'* linear system. Because the reduced system is often analytically tractable—linearity conceivably implies simplicity—network calculus is regarded as an attractive approach to analyze complex queueing systems. In this section we elaborate on this key reduction operation by exploring conceptual similarities with the more traditional linear system and queueing theories, as well as on its main diverging point from the two. The final goal is to highlight the emergence of the concept of a *service process*, which is instrumental for abstracting away some of the technical challenges characteristic of non-linear systems.

The 'complex' system is a node, or a network of nodes, in which bits arrive and depart according to various factors such as probability distributions for arrival processes, scheduling, routing, etc. A fundamental networking and queueing problem which is at the core of the philosophy of network calculus is the following:

System Identification (SI) Problem: Is it possible to characterize a random process (the departures) based on another random process (the arrivals) while accounting for yet another random process determined by other arrivals, scheduling, routing, etc. (the noise)?

To answer, let us formalize the system by an operator (a.k.a. filter)

$$T : \mathcal{F} \to \mathcal{F}, \ T(\mathbf{a}) = \mathbf{d} \ ,$$

where $\mathcal{F}$ is the set of discrete-time sequences, i.e., $\mathcal{F} = \{\mathbf{a} = (a_0, a_1, \dots) : a_i \in \mathbb{N}\}$. The physical interpretation of T is that it takes $\mathbf{a} = (a_0, a_1, \dots)$ as input, it accounts for the noise, and outputs $\mathbf{d} = (d_0, d_1, \dots)$. The sequences $\mathbf{a}$ and $\mathbf{d}$ have two networking interpretations, depending on the type of information they quantify:

1. Spatial quantification (SQ): a_n and d_n quantify the number of bits which arrive and depart from the network system at time n.

2. Temporal quantification (TQ): a_n and d_n quantify the arrival and departure times of the n^{th} bit.

The **SI** problem requires thus the construction of T such that for *any* input $\mathbf{a}$, the output $\mathbf{d}$ can be completely determined as $\mathbf{d} = T(\mathbf{a})$. The problem is difficult not only because *all* inputs must be accounted for by a *single* expression of T, but also because T should account for noise and its correlations with output and possibly input as well.

The next two sections, 3.1 and 3.2, present two partial solutions for the **SI** problem by exploiting key properties from linear system and queueing theories, respectively. Then Section 3.3 combines the ideas from these partial solutions into a more general, though approximative solution.

3.1 Linear System Theory

The **SI** problem has a direct correspondent in linear system theory [31]. Assume that T is *linear* and *time-invariant* (LTI), i.e.,

$$\begin{cases} T(c_1\mathbf{a}_1 + c_2\mathbf{a}_2) = c_1 T(\mathbf{a}_1) + c_2 T(\mathbf{a}_2) \\ T\left(\mathbf{a}_{(-k)}\right) = T(\mathbf{a})_{(-k)} \end{cases} \quad (1)$$

for all signals $\mathbf{a}, \mathbf{a}_1, \mathbf{a}_2$, scalars c_1, c_2, and integers k. Here we tacitly assume 'signal' interpretations of the input and output sequences, and also their extension to doubly infinite sequences such that the shifted version $\mathbf{a}_{(-k)}$ of $\mathbf{a}$, i.e., $a_{(-k)n} := a_{n-k} \ \forall n, k \in \mathbb{Z}$, is well defined. Define the Kronecker input signal $\mathbf{u}$ (also called impulse signal) and its corresponding output signal $\mathbf{v}$ (also called impulse-response)

$$u_n = \begin{cases} 0 & , \ n \ne 0 \\ 1 & , \ n = 0 \end{cases} \ , \ \mathbf{v} = T(\mathbf{u}) \ . \quad (2)$$

The impulse signal $\mathbf{u}$ is (technically) motivated by the convolution property $a_n = \sum_k a_k u_{n-k} \ \forall \mathbf{a}, n$.

With these assumptions, it can be shown that for *any* input signal $\mathbf{a}$, the corresponding output signal $\mathbf{d}$ can be completely determined by the following convolution

$$d_n = \sum_{k=-\infty}^{\infty} a_k v_{n-k} \ \forall n \in \mathbb{Z} \ . \quad (3)$$

This result is central in linear system theory, as it simply solves for T. This solution for the **SI** problem, however, relies on the strong LTI assumption from Eq. (1).

3.2 Queueing Theory and (min,+) vs. (max,+) Algebras

Here we present another solution for the **SI** problem by inspecting basic queueing properties in a simplified scenario. First, let us readopt one of the networking interpretations of the sequences $\mathbf{a}$ and $\mathbf{d}$.

Consider a single work-conserving server (node) with capacity C and unlimited queueing (buffering) space; all arrivals have the same size of one bit. The construction of the operator T, that completely determines $\mathbf{d} = T(\mathbf{a}) \ \forall \mathbf{a} \in \mathcal{F}$, follows directly from basic queueing dynamics, and is almost analogous to the one from Eq. (3). Under SQ, the output sequence d is determined by

$$d_1 + d_2 + \dots + d_n = \min_{0 \le i \le n} \{a_1 + \dots + a_i + C(n-i)\} \ ,$$

for all $n \ge 1$. It is convenient to represent the partial sums by considering the input and output (*cumulative*) sequences $\mathbf{A} = (A(0), A(1), \dots)$ and $\mathbf{D} = (D(0), D(1), \dots)$, where $A(n) = a_1 + \dots + a_n$, $D(n) = d_1 + \dots + d_n \ \forall n \ge 1$, and $A(0) = D(0) = 0$. With these notations, the operator T satisfies $\mathbf{D} = T(\mathbf{A})$, where

$$\begin{aligned} D(n) &= \min\{nC, A(1) + (n-1)C, \dots, A(n)\} \\ &= \min_{0 \le k \le n} \{A(k) + (n-k)C\} \ \forall n \in \mathbb{N} \ . \end{aligned} \quad (4)$$

In turn, under TQ, $\mathbf{d}$ is determined by

$$\begin{aligned} d_n &= \max\left\{a_1 + \frac{n}{C}, a_2 + \frac{n-1}{C}, \dots, a_n + \frac{1}{C}\right\} \\ &= \max_{1 \le i \le n} \left\{a_i + \frac{n-i+1}{C}\right\} \ \forall n \in \mathbb{N} \ . \end{aligned} \quad (5)$$

These equations are fundamental elementary identities in queueing theory.

The operations from Eqs. (4) and (5) on input sequences resemble much with the convolution operation from Eq. (3), except for the underlying algebra. Concretely, while Eq. (3) is formed according to the traditional convolution involving the addition of products, Eqs. (4) and (5) are formed by minimizing and maximizing, respectively, sums. For this reason, it is said that the operator T operates in a $(\min, +)$ algebra in Eq. (4), and in a (max,+) algebra in Eq. (5).

We have thus presented another complete characterization of T. What makes this second solution partial as well is that the considered queueing system is noiseless, i.e., it assumes a constant-rate server, no scheduling, etc.

3.3 Emergence of the Service Process Concept

We now combine the ideas from the previous two subsections in order to present a much more general solution for the **SI** problem, and concomitantly to highlight the emergence of one of the key modelling concepts in network calculus: the *service process*.

We tailor the **SI** problem, for some general queueing system, in terms of an *unknown* operator

$$T : \mathcal{F} \to \mathcal{F},\ T(\mathbf{A}) = \mathbf{D}\ , \tag{6}$$

where $\mathbf{A}$ and $\mathbf{D}$ have the interpretations from the previous subsection, i.e., cumulative sequences counting bits. Recall that T has to be constructed in such a way that it completely determines the output $\mathbf{D}$ for *any* input $\mathbf{A}$.

Inspired from the previous two subsections, it is intuitive to reproduce the steps for the construction of an impulse-response from Section 3.1, but in the modified $(\min, +)$ algebra which was shown to be appropriate, in Section 3.2, to represent input-output relationships in queueing systems. This approach could be viewed as a merge between linear system theory and queueing theory.

As in Section 3.1, we first enforce an LTI assumption on T by reproducing Eq. (1) in the $(\min, +)$ algebra:

$$\begin{cases} T\left(\min\left\{c_1 + \mathbf{A}_1, c_2 + \mathbf{A}_2\right\}\right) \\ \quad = \min\left\{c_1 + T\left(\mathbf{A}_1\right), c_2 + T\left(\mathbf{A}_2\right)\right\} \\ T\left(\mathbf{A}_{(-k)}\right) = T(\mathbf{A})_{(-k)} \end{cases}\ , \tag{7}$$

for all sequences $\mathbf{A}, \mathbf{A}_1, \mathbf{A}_2$, scalars c_1, c_2, and shifts $k \in \mathbb{Z}$ (whether such an apparently strong assumption holds for typical queueing systems will be clarified in two follow-up examples). The second step is to define the analogue of the Kronecker impulse signal and its shifted version in the $(\min, +)$ algebra, i.e.,

$$\delta(n) = \begin{cases} 0 & ,\ n = 0 \\ \infty & ,\ n \neq 0 \end{cases}\ ,\ \delta_{-k}(n) = \delta(n-k),\ \forall\ n, k \in \mathbb{Z}\ . \tag{8}$$

Similarly to the Kronecker impulse signal, the newly defined impulse sequence $\boldsymbol{\delta}$ is motivated by the fact that any input sequence $\mathbf{A}$ can be expressed as the $(\min, +)$ convolution of itself with the impulse function, i.e.,

$$\mathbf{A} = \mathbf{A} * \boldsymbol{\delta},\ \text{or, equivalently,}$$
$$A(n) = \min_{0 \le k \le n} \left\{A(k) + \delta(n-k)\right\}\ \forall n \in \mathbb{N}\ .$$

When the input to the system is the impulse $\boldsymbol{\delta}$, define the corresponding output as the impulse-response

$$\mathbf{S} = T(\boldsymbol{\delta})\ . \tag{9}$$

Under the assumption that T is $(\min, +)$ LTI (in the sense of Eq. (7)), it follows from the $(\min, +)$ linear system theory (see [5], p. 136, or [2], p. 276) that for *any* input sequence $\mathbf{A}$ the corresponding output sequence $\mathbf{D}$ satisfies

$$D(n) = \min_{0 \le k \le n} \left\{A(k) + S(n-k)\right\}\ , \tag{10}$$

where $\mathbf{S} = T(\boldsymbol{\delta})$ is the impulse-response. Therefore, the unknown operator T is now fully characterized: for *any* input sequence $\mathbf{A}$, the corresponding output is $T(\mathbf{A}) = \mathbf{A} * \mathbf{S}$.

Note that despite the cyclic dependence between T and S, i.e., $T(\mathbf{A}) = \mathbf{A} * \mathbf{S}$ and $\mathbf{S} = T(\boldsymbol{\delta})$, T is well-defined. The reason is that the impulse-response $\mathbf{S}$, which induces the cyclic dependence, has a well defined physical meaning as the system's output for the input $\boldsymbol{\delta}$.

The key observation to make, however, is that T is yet another partial solution for the **SI** problem, due to the underlying LTI assumption from Eq. (7). In the following we present two queueing examples which show that, in spite of the fact that many queueing systems are generally *not* $(\min, +)$ linear, the unknown operator T, and thus a solution for the **SI** problem, can be constructed in great generality. Since there is no free lunch, this promising increase in generality is only possible at an inevitable price: sacrificing exactness, or, more concretely, replacing the equality from Eq. (6) by an inequality, i.e., $\mathbf{D} \ge T(\mathbf{A})$.

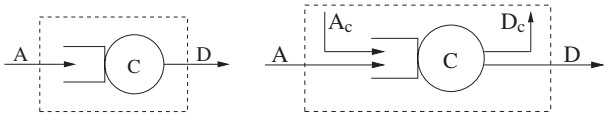
3.3.1 Example 1

Consider the (noiseless) queueing system from Figure 1.(a) (see next page) and recall the relationship from Eq. (4) between departures and arrivals. Fitting this relationship with Eq. (10) yields $S(n) = nC\ \forall n \in \mathbb{N}$. As mentioned, $\mathbf{S}$ has also the physical interpretation of the cumulative output from the queue when the input is the impulse $\boldsymbol{\delta}$ from Eq. (8).

This example, although repetitive, is meant to illustrate the (almost perfect) analogy in the arguments used in Sections 3.1 and 3.2. Concretely, it is apparent that there is a match between the construction of $\mathbf{S}$ from 1) the $(\min, +)$ linear system theory (as in Eqs. (9) and (10)), and 2) elementary queueing properties (as in Eq. (4)). The missing element for a perfect analogy is that the queueing system is $(\min, +)$ LTI under an artificial interpretation of the required 'plus' property $T(c + \mathbf{A}) = c + T(\mathbf{A})$ from Eq. (7): the addition of scalars c occurs, both in the input and output, before the queueing system actually starts. This is clearly an inconvenient physical system interpretation, but it enables the view of a constant-rate queueing server as a $(\min, +)$ LTI system. As a side remark, the physically more meaningful interpretation of the 'plus' property as a burst of c bits at time zero has the negative consequence that any system of practical interest, in particular the constant-rate work-conserving server, would be $(\min, +)$ non-linear. This can be seen by a result in ([5], Proposition 8.3.1) on a system with non-empty initial buffer clearly exhibiting a $(\min, +)$ non-linear behavior.

3.3.2 Example 2

Making an analogy between $(\min, +)$ LTI systems and elementary queueing properties is even more compounded for the FIFO queueing system from Figure 1.(b), which includes noise in the form of cross-traffic. The reason is that this queueing system is *not* anymore $(\min, +)$ linear, even under the artificial interpretation of the 'plus' operation. First, the 'min' property $T(\min\{\mathbf{A}_1, \mathbf{A}_2\}) = \min\{T(\mathbf{A}_1), T(\mathbf{A}_2)\}$



(a) No multiplexing (b) Multiplexing

Figure 1: Two queueing systems from the perspective of flow A: in (a) the flow is isolated, in (b) the flow shares the queue with another (cross) flow $\mathbf{A}_c$

fails; a quick example is $C = 3$, $A_c(t) = (0, 1, 3, 4)$, $A_1(t) = (0, 3, 4, 6)$, and $A_2(t) = (0, 1, 7, 10)$. Second, the time invariance property fails as well; a quick example is $C = 1$, $\mathbf{A} = (0, 1, \dots)$, $\mathbf{A}_c = (0, 1, 3, \dots)$, and the right shift $k = 2$. Because the queueing system is not $(\min, +)$ LTI, one cannot follow the construction of the impulse-response sequence $\mathbf{S}$ in order to exactly characterize the queueing system as in Eq. (10).

At this apparent impasse, the network calculus slightly diverges from LTI systems and queueing theory. The key idea is to transform the non-linear queueing system into a 'somewhat looking' linear system. The actual transformation occurs by directly constructing a 'somewhat analogous' impulse-response $\mathbf{S}$ (a.k.a. *service process*) [1] satisfying

$$\mathbf{D} \geq \mathbf{A} * \mathbf{S} \; \forall \mathbf{A} \; . \tag{11}$$

Therefore, instead of *exactly* characterizing the system as in Eq. (6), the network calculus makes the crucial concession of *inexactly* characterizing the system by resorting to an *inequality*, as in Eq. (11). For the FIFO multiplexing example, one choice of a service process $\mathbf{S}$ is the bivariate random process $S_i(k, n) = [C(n-k) - A_c(k, n-i)]_+ 1_{\{n-k>i\}}$, for some $i \geq 0$ [18], which satisfies

$$D(n) \geq \min_{0 \leq k \leq n} \{A(k) + S_i(k, n)\} \; \forall \mathbf{A}, n \in \mathbb{N} \; . \tag{12}$$

Except for the inequality, this characterization resembles much with both Eq. (10) (from $(\min, +)$ LTI systems theory) and Eq. (4) (from queueing theory). Note, however, the non-trivial expression of $S_i(k, n)$ stemming from non-trivial characteristics of FIFO multiplexing. In particular, the bivariate form is due to the lack of time invariance.

The FIFO multiplexing example reflects a fundamental tradeoff in network calculus. On one hand, as queueing systems are generally neither linear nor time invariant, network calculus resorts to inequalities for their characterization, as in Eqs. (11) or (12). On the other hand, the service processes $\mathbf{S}$ ought to be reasonably concise and also provide tight bounds in Eq. (11); otherwise, they may render a cumbersome analysis (see, e.g., [33] for the above choice of $\mathbf{S}$) or simply arbitrarily loose bounds. Constructing such 'nice' service processes for existing scheduling algorithms is a key challenge in network calculus; see [36] for some state-of-the-art examples of service processes concerning Δ-scheduling, which generalizes FIFO, static priority, and earliest deadline first (EDF).

Nevertheless, the underlying methodology of constructing service processes to abstract away the details of scheduling algorithms, in queueing scenarios with many flows, renders two central features of network calculus: *scheduling abstraction* and *convolution-form networks*. With the former, many classes of scheduling policies and arrival processes are amenable to a *uniform analysis*. In other words, once service processes are suitably constructed, the network calculus analysis does not conceptually differentiate between, e.g., FIFO and EDF policies, or Poisson and Markov arrival processes, for the purpose of computing *per-flow (or per-class) performance metrics*. With the latter feature, the multi-node queueing analysis is drastically simplified. Concretely, once service processes $\mathbf{S}^i$ are constructed at each node along a network path, the entire network analysis can be reduced to the analysis of a single-node, which is characterized by the following service process

$$\mathbf{S} = \mathbf{S}^1 * \mathbf{S}^2 * \ldots * \mathbf{S}^n \; , \tag{13}$$

i.e., the convolution of the service processes along the network path. What makes this reduction particularly appealing is that the multi-node performance bounds obtained in this manner are asymptotically tight in the number of nodes (see, e.g., [14, 22]).

In conclusion, network calculus provides a methodology to solve the **SI** problem by transforming a non-linear system (subject to various arrivals, scheduling, or multi-node) into a 'somewhat looking' linear system which is amenable to a quite straightforward analysis. The key challenge is the transformation itself, i.e., the construction of 'nice' service processes.

Most of the interpretations on network calculus illustrated in this section appear in the literature in isolation: for the analogy with linear systems see [19, 12, 5], for the analogy with queueing theory see [27], for a discussion on the non-linearity of FIFO systems see [35]. For a comprehensive survey of service processes we refer to [23]. Our contribution herein was to present a comprehensive perspective on the emergence and central role of service processes in network calculus, by weaving together linear systems, queueing theory, and network calculus.

4. ENVELOPE PROCESSES

We now shift the discussion to the other fundamental concept in network calculus: *envelope processes*. While their role is to model a very broad class of arrival processes, achieving this generality comes at the price of sacrificing exactness in the arrivals' representation. The goal of this section is to highlight the key aspects of envelope processes; it is not meant to provide a review of the types of envelopes, for which we refer to [37].

A (cumulative) arrival process $A(n)$ is typically described by either a complementary cumulative distribution function (CCDF) or a moment generation function (MGF), i.e.,

$$F_{A(n)}(\sigma) := \mathbb{P}\Big(A(n) > \sigma\Big) \; , \quad M_{A(n)}(\theta) := E\left[e^{\theta A(n)}\right] \; ,$$

respectively, for all $n \in \mathbb{N}$, $\sigma \in \mathbb{R}$, and $\theta \in \Theta$, where Θ is some space over $\mathbb{R}$. The two descriptions silently assume that $A(n)$ is a stationary random process, which means that the CCDF is invariant under time shift.

The existence of the MGF is equivalent to an exponentially bounded CCDF, in which case the CCDF uniquely determines the MGF according to the identity relating expectations and tails, i.e., $E[X] = \int_0^\infty \mathbb{P}(X > x)dx$ for positive random variable (r.v.) X. Conversely, in the case when Θ is an open interval including zero, the MGF uniquely determines the CCDF according to analytic function theory ([21], p. 274). Throughout we consider arrival processes which have an MGF.

Let us consider the following example of a compound Bernoulli arrival process

$$A(n) = \sum_{k=1}^{n} X_k \ \forall n \in \mathbb{N} \ , \tag{14}$$

where X_k's are i.i.d. Bernoulli(p) r.v.'s taking the values 1 and 0 with probabilities p and $1-p$, respectively. The simplicity of $A(n)$ will enable the illustration of some key insights in an intuitive and yet rigorous manner. The corresponding expressions for the CCDF and MGF are

$$\begin{array}{l} F_{A(n)}(\sigma) = \left\{ \begin{array}{l} 1, \ \sigma < 0 \\ \sum_{k=\lfloor \sigma \rfloor + 1}^{n} \binom{n}{k} p^k (1-p)^{n-k}, \ \sigma \geq 0 \end{array} \right. \\ M_{A(n)}(\theta) = e^{\theta r n} \end{array} \ , \tag{15}$$

where $r = \frac{\log\left(pe^{\theta}+1-p\right)}{\theta}$ is a rate and $\theta \in \mathbb{R}_+^*$.

Although the process $A(n)$ is fully described, the network calculus provides more 'flexibility' by offering further modelling alternatives depending on two factors:

1. type of analysis: deterministic (a.k.a. worst-case) or probabilistic. The former seeks to yield statements like "The (queueing) delay is smaller than some number." The latter seeks statements like "The delay is smaller than some number *with some probability*."
2. tradeoff between accuracy and elegance of the analysis itself.

For instance, in the case of a deterministic analysis, one must resort to deterministic models to (partially) suppress the uncertainty of arrivals. The suppression process occurs by replacing random processes with deterministic functions, which are referred to as deterministic envelopes (see Section 4.2). In turn, in the case of a probabilistic analysis, one can carry on with probabilistic arrival models, like CCDF or MGF, throughout the analysis.

A word of caution is in place regarding carrying on with *exact* probabilistic arrival models, like CCDF or MGF. Although they can lend themselves to tight (or possibly exact) bounds on performance metrics, the obtained results and the analysis itself may lack elegance and insight. The CCDF from Eq. (15) conceivably lends itself to quite a messy analysis. As another example, the very broad class of Markov arrival processes can be modelled by an exact MGF expression, as a weighted hyperexponential (see, e.g., [15])

$$M_{A(n)}(\theta) = \sum_{l=1}^{L} w_l e^{\theta r_l n} \tag{16}$$

with L terms, where $\sum w_l = 1$ and r_l's are rates. Carrying on with the entire sum of exponentials can be very cumbersome, and also prone to numerical problems, especially when L is large. Often, a much simplified arrival model consisting of an MGF bound with very few (one or two) dominant exponentials is sufficient with only a negligible loss in tightness. MGF or CCDF bounds are even more appropriate when exact expressions are difficult to derive.

This bounding approach is instrumental to the philosophy of network calculus concerning the modelling of arrival processes. On one hand, it significantly widens the modelling scope of arrivals. On the other hand, it lends itself to an elegant analysis, not only in the sense of carrying out concise formulas, but also in the sense that the final formulas may be amenable to, e.g., convex optimizations encountered in dimensioning problems. For the rest of this section we present some key arrival models in network calculus. The goal of the presentation is to give insight on the issue of 'What are suitable bounding models for arrival processes?'.

4.1 A Pitfall: The Simplest Arrival Bound

Perhaps the most tempting bound for an arrival process $A(n)$ is the following

$$A(n) \leq \mathcal{G}(n) \ \forall n \in \mathbb{N} \ , \tag{17}$$

where $\mathcal{G}(n)$ is a non-random function. It is important to remark that for every time n, the model bounds a r.v., i.e., $A(n)$, by a non-random number, i.e., $\mathcal{G}(n)$. As an example, if $A(n)$ is the compound Bernoulli arrival process from Eq. (14), then any function $\mathcal{G}(n)$ satisfying $\mathcal{G}(n) \geq n$ would fit Eq. (17); note that $\mathcal{G}(n) = n$ would be the tightest achievable deterministic bound. As another example, if $A(n)$ is a (discretized) Poisson process, or any random process taking arbitrarily large values with non-zero probabilities, then *no* bounded function $\mathcal{G}(n)$ would fit Eq. (17).

The actual drawback of the bounding model from Eq. (17) does not stand in its apparent weak modelling power, but rather in its incompleteness for computing performance measures, e.g., the queueing backlog. Let us reconsider the queueing scenario from Figure 1.(a). The backlog process B_n is defined as

$$B_n = A(n) - D(n) \ ,$$

i.e., the amount of bits in the system at time n. Recalling the expression for $D(n)$ from Eq. (4), one may immediately derive Reich's equation:

$$B_n = \max_{0 \leq k \leq n} \left\{ A(k,n) - C(n-k) \right\} \ . \tag{18}$$

This equation clearly indicates that the bounding model from Eq. (17) is insufficient to get a non-degenerate bound on the backlog process. That is because a bound on $A(n)$ does not necessarily induce a bound on $A(k,n)$, which is what is actually needed. For a quick example, take the arrival process $\mathbf{A} = (0, 0, 20)$ and $\mathcal{G}(n) = 10n$; clearly, $A(n) \leq \mathcal{G}(n)$ but $A(1,2) = 20 > 10 = \mathcal{G}(1)$. A more intuitive argument is that the backlog process depends on the entire history of the process, and relative to *any* time point, whereas Eq. (17) only captures the process relative to the origin.

Let us next make an additional stationarity assumption on $A(n)$. Then, if a deterministic computation of queueing measures was the goal, the model from Eq. (17) would remain insufficient; in fact, the model would only be sufficient for constant-rate arrivals. However, if a probabilistic computation was the goal, then the model from Eq. (17) could become useful. Indeed, a computation of the backlog tail would be for instance

$$\begin{array}{rcl} \mathbb{P}\left(B_n > \sigma\right) & = & \mathbb{P}\left(\max_{0 \leq k \leq n} \left\{ A(k,n) - C(n-k) \right\} > \sigma \right) \\ & \leq & \sum_{k=0}^{n} \mathbb{P}\left(A(k,n) - C(n-k) > \sigma \right) \ . \end{array} \tag{19}$$

The last line follows from Boole's inequality[2]. As a side remark, we point out that this apparently loose inequality is

[2]For some probability events E and F, Boole's inequality is $\mathbb{P}\left(E \cup F\right) \leq \mathbb{P}(E) + \mathbb{P}(F)$.

not so bad if the r.v.'s $X_k := A(k,n)$ are rather uncorrelated (e.g., when $A(n)$ is a (discretized) Poisson process), but it is quite loose if they are highly correlated [46]. In such a case one can make use of more sophisticated techniques with refined martingale inequalities (see [12], pp. 339-343). For the purpose of our presentation it is sufficient to adopt the simplified technique with Boole's inequality.

Due to the stationarity assumption, the last line can be continued by replacing $A(k,n)$ with $A(n-k)$, and further by $\mathcal{G}(n-k)$ according to the bounding arrival model from Eq. (17). Note however that $\mathcal{G}(n)$ should be defined as a random process, and the inequality from Eq. (17) should hold a.s. (almost surely, i.e., $\mathbb{P}\left(A(n) < \mathcal{G}(n)\right) = 0$). Otherwise, if $\mathcal{G}(n)$ was a non-random function, then the previous derivation would be quasi-deterministic since all the probabilities would evaluate either to 0 or 1.

4.2 Classic Deterministic Arrival Model

The previous pitfall indicates that, in general, it is not sufficient to bound arrivals on all intervals $(0,n)$ but rather on all intervals (k,n). This observation suggests the following bounding arrival model

$$A(k,n) \leq \mathcal{G}(n-k) \ \forall 0 \leq k \leq n \ , \tag{20}$$

where $\mathcal{G}(n)$ is a non-random function. Network calculus was essentially founded on this arrival model [16].

With this arrival model, the deterministic continuation of Eq. (18) is straightforward:

$$B_n \leq \max_{0 \leq k \leq n} \{\mathcal{G}(k) - Ck\} \ . \tag{21}$$

The RHS term can be computed explicitly in $\mathcal{O}(1)$ time if $\mathcal{G}(n)$ is a sufficiently 'nice' expression, e.g., $\mathcal{G}(n) = rn + b$ where r and b have the meanings of rate, and burst, respectively. Otherwise, if $\mathcal{G}(n)$ is given pointwise, then the RHS term can be computed in $\mathcal{O}(n)$ time.

We emphasize that there is no requirement of stationarity on the arrival process $A(n)$. In fact, the regularity constraint from Eq. (20) is satisfied by infinitely many (and possibly unknown) arrival processes, thus illustrating the high modelling potential of Eq. (20). Moreover, despite the apparent tradeoff between modelling potential and accuracy of representation, the derivation from Eq. (21) is actually *tight* ([5], p. 27). Tightness means that there exists an arrival process which 1) satisfies the arrival bound from Eq. (20), and 2) induces a backlog process which matches with the predicted bound from Eq. (21). Even more remarkably, the tightness of the backlog bound holds even when multiplexing many possibly 'conspiring' flows, e.g., producing large bursts at the same time.

To more concretely elaborate on the *tightness* of the deterministic modelling and analysis from Eqs. (20)-(21), let us consider an aggregate $A(n) = \sum_{j=1}^{N} A_j(n)$, where $A_j(n)$'s are compound Bernoulli processes as in Eq. (14). The flows are multiplexed at a server with capacity $C < N$, and there is no statistical independence assumption amongst them. The performance metric of interest is again a bound on the backlog process B_n. According to Eq. (21), to get the tightest bound on B_n, one must first construct the smallest non-random function satisfying Eq. (20), which is $\mathcal{G}(n) = Nn$. When plugged into Eq. (21), this yields the bound

$$B_n \leq \max_{0 \leq k \leq n} \{N(n-k) - C(n-k)\} = (N-C)n \ .$$

Because $N > C$ the bound diverges and clearly becomes useless when $n \to \infty$. We point out, however, that the bound is *tight* even under a statistical independence assumption on $A_j(n)$'s. Indeed, $\forall p, n > 0$ there exists a positive probability such that $A_j(n) = n \ \forall j$, i.e., there exists a sample-path which attains the apparently very loose bound on B_n.

Another illustrative example concerns possible degenerate results obtained from deterministic modelling and analysis. This is the case when the arrival process $A(n)$ can take infinitely large values, i.e., $\forall \ K > 0 \ \exists \varepsilon_K > 0$ such that $\mathbb{P}(A(n) > K) > \varepsilon_K$ (the Poisson process is an example). For such arrival processes, the regularity constraint from Eq. (20) is only satisfied by the degenerate function $\mathcal{G}(n) = \infty$, which clearly yields the degenerate, and useless, backlog bound $B_n \leq \infty$. However, reiterating the previous argument, this degenerate bound is also tight.

We conclude here by pointing out that if one seeks deterministic bounds from deterministic or even stochastic arrival models, then DNC is an attractive theory (illustrated here by Eqs. (20)-(21)): it has a high modelling potential and it (mostly) yields tight bounds. Otherwise, if one seeks *probabilistic* bounds, e.g.,

$$\mathbb{P}\left(B_n > \sigma\right) \leq \varepsilon(\sigma), \text{ where } \varepsilon(\sigma) \text{ is to be determined,}$$

then DNC is an inopportune theory. The main reason is that a purely deterministic analysis can yield extremely loose bounds due to not leveraging from statistical multiplexing gain. This discussion will be continued in Section 5.

4.3 Stochastic Arrival Models

A probabilistic analysis generally requires a probabilistic arrival model. Here we consider three of the main stochastic arrival models proposed in the SNC literature. For the sake of presentation we use the names SBB, S^{2}BB, and S^{3}BB, and omit original names. The goal of this subsection is to explain their motivation and benefits.

$$\text{SBB}: \ \mathbb{P}\Big(A(k,n) - \mathcal{G}(n-k) > \sigma\Big) \leq \varepsilon(\sigma) \ \forall \ k,n,\sigma$$

$$\text{S}^2\text{BB}: \ \mathbb{P}\Big(\max_{0 \leq k \leq n} \{A(k,n) - \mathcal{G}(n-k)\} > \sigma\Big) \leq \varepsilon(\sigma) \ \forall \ n,\sigma$$

$$\text{S}^3\text{BB}: \ \mathbb{P}\Big(\max_{0 \leq k \leq n \leq \infty} \{A(k,n) - \mathcal{G}(n-k)\} > \sigma\Big) \leq \varepsilon(\sigma) \ \forall \ \sigma$$

$\mathcal{G}(n)$'s are non-random and are called *envelope functions*. $\varepsilon(\sigma)$'s are called *error functions*. Some technical and quite intuitive conditions are that the envelope functions are non-decreasing, whereas the error functions are non-increasing. Note also that a degree of freedom of the bounding approach is that for all three models the arrival process $A(n)$ does not need to be stationary, although the bounds themselves (the envelopes $\mathcal{G}(n)$) are so.

Before we explain the three models, it is important to observe the formation of the bottom two: S^{2}BB [17] is formed by inserting the free variable k from SBB (short-hand for stochastically bounded burstiness) [45] into the probability, whereas S^{3}BB [26] is further formed by inserting the free variable n as well. Informally, S^{3}BB measures events consisting of all past histories of the process $A(n)$, i.e., relative to *all* times. In turn, S^{2}BB measures events consisting of a single past history, i.e., relative to a fixed time, whereas SBB measures events as single fragments of past histories.

Although the SBB model seems the simplest amongst the three, it is actually the S^{2}BB model which is the natural

extension of the classic deterministic model from Eq. (20). To see the reason, rewrite Eq. (20) as

$$\max_{0\le k\le n}\{A(k,n)-\mathcal{G}(n-k)\}\le 0\ \forall\ n\ . \tag{22}$$

Note that the S^{2}BB model enforces a bound on the CCDF of the LHS term above. In other words, the S^{2}BB model quantifies, with an upper bound, the probability that the deterministic model is violated by more than σ. An attractive property of S^{2}BB is that it immediately lends itself to the calculation of performance bounds. Indeed, a straightforward manipulation of Eq. (18) and S^{2}BB yields the bound

$$\mathbb{P}\left(B_n>\sigma_0+\sigma\right)\le\varepsilon(\sigma)\ , \tag{23}$$

where $\sigma_0=\max_{k\ge 0}\{\mathcal{G}(k)-Ck\}$ is exactly the deterministic backlog bound from Eq. (21). To recapitulate, S^{2}BB quantifies the violation probabilities of the deterministic model (see S^{2}BB and Eq. (22)), whereas the probabilistic backlog bound quantifies the violation probabilities of the deterministic backlog bound (see Eqs. (23) and (21)). As these violation probabilities are identical, one can argue that S^{2}BB is the 'natural' probabilistic extension of the deterministic arrival model from Eq. (20) (or, equivalently, from Eq. (22)).

In practice, the choice of SBB vs. S^{2}BB depends on the arrivals' input. If the input is a measurement trace, then S^{2}BB should be chosen since it immediately lends itself to performance bounds, as shown earlier. Given a trace $A(n)$ with n elements, an S^{2}BB fitting algorithm would follow the steps: 1) make a guess on $\mathcal{G}(n)$ (e.g., $\mathcal{G}(n)=(r+\delta)n$, where r is the average rate of the trace and $\delta>0$ is a tuning parameter), 2) compute the partial sums $A(k,n)$ and the LHS terms in Eq. (22), and 3) fit a distribution function. Ignoring the accuracy of the fitting, i.e., the range of values σ, the algorithm runs in $O(n^2)$ time. There is no specific rule for the tuning parameter δ, which is to be optimized numerically.

If the arrivals' input is some random process $A(n)$, then it is generally easier to first fit the SBB model. A typical way is to derive an MGF bound, e.g., $M_{A(n)}(\theta)\le e^{\theta r n}$, for some $\theta>0$ (see Eq. (16) for Markov arrival processes). If $A(n)$ is also stationary, then an SBB model can be fitted using the Chernoff bound[3], i.e.,

$$\mathbb{P}\left(A(k,n)>r(n-k)+\sigma\right)\le e^{-\theta\sigma}\ \forall k,n,\sigma\ . \tag{24}$$

This SBB model further lends itself to the S^{2}BB model. Indeed, one can write for some $\delta>0$:

$$\begin{aligned}\mathbb{P}&\left(\max_{0\le k\le n}\{A(k,n)-(r+\delta)(n-k)\}>\sigma\right)\\ &\le\sum_{0\le k<n}\mathbb{P}\left(A(k,n)>r(k-n)+\delta(k-n)+\sigma\right)\\ &\le\sum_{k\ge 1}e^{-\theta\delta k}e^{-\theta\sigma}\le\frac{1}{\theta\delta}e^{-\theta\sigma}\ .\end{aligned}$$

The second line follows from Boole's inequality, whereas the exponential bounds in the last line follow from Eq. (24). What is important to remark is that the transition from the SBB to S^{2}BB involves a rate increase from r to $r+\delta$. This penalty is due to the need to obtain a bounded error function for S^{2}BB; note that, if δ was zero, then the obtained error function would be unbounded.

[3]For some r.v. X and $x,\theta\in\mathbb{R}_+$, the Chernoff bound is $\mathbb{P}(X>x)\le M_X(\theta)e^{-\theta x}$.

4.4 Quasi-Determinism in the S^{3}BB Model

So far we have only commented on the applicability of the SBB and S^{2}BB models. The reason is that, as we demonstrate in this section, the S^{3}BB model is quasi-deterministic for the class of stationary and ergodic arrival processes. Formally, the quasi-determinism means that the corresponding violation probabilities, set through the error function $\varepsilon(\sigma)$, can only take the extreme values, i.e.,

$$\varepsilon(\sigma)\in\{0,1\}\ \forall\sigma\ . \tag{25}$$

The immediate consequence is that the resulting SNC formulation from [26] is essentially quasi-deterministic, and does not capture statistical multiplexing gain. In fact, multiplexing quasi-deterministic S^{3}BB flows yields quasi-deterministic aggregates, by using the Superposition Property from [26]; for the precise meaning of 'statistical multiplexing gain' we refer to Section 5.

We next prove the quasi-determinism claim for stationary and ergodic processes, and then construct two rather contrived arrival processes for which the S^{3}BB model is not necessarily quasi-deterministic.

4.4.1 Stationary and Ergodic Processes

First we give some definitions (see Breiman [7], pp. 104-120). Consider a random process $\mathbf{X}=(X_1,X_2,\dots)$ defined on some joint probability space $(\Omega,\mathcal{F},\mathbb{P})$; the Borel σ-field of the subsets of $\mathbb{R}$ is denoted by $\mathcal{B}$. We denote $I=\{1,2,\dots\}$, and the product spaces $\mathbb{R}^I=\{\mathbf{x}=(x_1,x_2,\dots):x_i\in\mathbb{R}\}$ and $\mathcal{B}^I=\{\mathbf{B}=(B_1,B_2,\dots):B_i\in\mathcal{B}\}$.

By definition, the process $\mathbf{X}$ is (strongly) stationary if

$$\begin{aligned}&\mathbb{P}\Big(X_{i_1}\le x_1,X_{i_2}\le x_2,\dots,X_{i_n}\le x_n\Big)\\ &\quad=\mathbb{P}\Big(X_{i_1+k}\le x_1,X_{i_2+k}\le x_2,\dots,X_{i_n+k}\le x_n\Big)\ ,\end{aligned}$$

for all $n,k\in\mathbb{N}^*$, $0\le i_1\le i_2\le\dots\le i_n$, and $x_1,x_2,\dots,x_n$. In other words, stationarity means that the distribution of any sequence $(X_{i_1},X_{i_2},\dots,X_{i_n})$ is invariant under shift.

Further, the notion of ergodicity requires the introduction of an explicit *shift operator* $T:\mathbb{R}^I\to\mathbb{R}^I$, defined as

$$T\left(x_1,x_2,\dots\right)=\left(x_2,x_3,\dots\right)\ ,$$

for all sequences $\mathbf{x}=(x_1,x_2,\dots)$. The stationarity of $\mathbf{X}$ implies that T is *measure preserving*, i.e., by definition

$$\mathbb{P}\left(\mathbf{X}\in B\right)=\mathbb{P}\left(T\mathbf{X}\in B\right)\ \forall B\in\mathcal{B}^I\ .$$

For $B\in\mathcal{B}$, the event $\{\mathbf{X}\in B\}$ is said to be *invariant* if

$$\{\mathbf{X}\in B\}=\{T\mathbf{X}\in B\}\ \ \mathbb{P}\text{-a.s.}\ ,$$

i.e., the events $\{\mathbf{X}\in B\}$ and $\{T\mathbf{X}\in B\}$ differ by a set of probability zero. In other words, the event $\{\mathbf{X}\in B\}$ is invariant if its incidence does not depend (a.s.) on any finite prefix of $\mathbf{X}$. Finally, the process $\mathbf{X}$ is *ergodic* if any invariant event has probability 0 or 1.

The following lemma will be used to prove the claim of quasi-determinism.

LEMMA 1. *Consider a stationary and ergodic process* $\boldsymbol{X}=(X_1,X_2,\dots)$. *Then*

$$\mathbb{P}\Big(\max\{X_1,X_2,\dots\}>\sigma\Big)\in\{0,1\}\ \ \forall\sigma\ .$$

The lemma implies that $\max\{X_1,X_2,\dots\}=K$ a.s., where K is a constant or $K=\infty$.

PROOF. Fix σ and let $B = (-\infty, \sigma]^I$. We shall prove that

$$\{\mathbf{X} \in B\} = \left\{ \max_{i \geq 1} X_i \leq \sigma \right\}$$

is an invariant event, which is equivalent to showing that

$$\mathbb{P}\left(\{\mathbf{X} \in B\} \,\Delta\, \{T\mathbf{X} \in B\}\right) = 0 \ , \tag{26}$$

where 'Δ' denotes the symmetric difference and $\{T\mathbf{X} \in B\} = \{\max_{i \geq 2} X_i \leq \sigma\}$.

Let us first note that

$$\begin{aligned} &\mathbb{P}\left(\max\{X_1, X_2, \dots\} > \sigma\right) \\ &\quad = \lim_{n \to \infty} \mathbb{P}\left(\max\{X_1, X_2, \dots X_n\} > \sigma\right) \\ &\quad = \lim_{n \to \infty} \mathbb{P}\left(\max\{X_2, X_3, \dots, X_{n+1}\} > \sigma\right) \\ &\quad = \mathbb{P}\left(\max\{X_2, X_3, \dots\} > \sigma\right) \ . \end{aligned} \tag{27}$$

The second and last lines follow from the monotone convergence theorem (if B_n is a non-decreasing sequence of events, then $\mathbb{P}(\lim_n B_n) = \lim_n \mathbb{P}(B_n)$). The third line follows from the stationarity of X_n.

Expanding the symmetric difference from Eq. (26) into the union of two events, we have for the first one

$$\mathbb{P}\left(\left\{ \max_{i \geq 1} X_i \leq \sigma \right\} \cap \left\{ \max_{i \geq 2} X_i > \sigma \right\}\right) = \mathbb{P}(\emptyset) = 0 \ .$$

For the second one we use the inclusion-exclusion formula:

$$\begin{aligned} &\mathbb{P}\left(\left\{ \max_{i \geq 1} X_i > \sigma \right\} \cap \left\{ \max_{i \geq 2} X_i \leq \sigma \right\}\right) \\ &\quad = \mathbb{P}\left(\max_{i \geq 1} X_i > \sigma \right) + \mathbb{P}\left(\max_{i \geq 2} X_i \leq \sigma \right) - 1 \\ &\quad = 0 \ . \end{aligned}$$

In the last line we applied Eq. (27). Collecting terms implies that Eq. (26) holds and thus the event $\{\mathbf{X} \in B\}$ is invariant. Because $\mathbf{X}$ is ergodic, it follows that $\mathbb{P}(\mathbf{X} \in B) \in \{0, 1\}$, which completes the proof. □

We are now ready to demonstrate the quasi-determinism claim. Let the S^{3}BB model from Section 4 for an arrival process $A(n)$, some envelope $\mathcal{G}(n)$, and error function $\varepsilon(\sigma)$. Assume that $a_n := A(n-1, n)$ is stationary and ergodic. It then follows that for any $m \in \mathbb{N}^*$, the *block* process $\left(X_n^{(m)}\right)_{n \geq 1}$ comprising blocks of m consecutive instances of a_n and defined as

$$X_n^{(m)} := A(n-1, n+m-1) - \mathcal{G}(m) \ \forall n \geq 1$$

is also stationary and ergodic (cf. [7], Propositions 6.6 and 6.31). According to Lemma 1, there exists $K^{(m)}$'s such that

$$\max\left\{ X_1^{(m)}, X_2^{(m)}, \dots \right\} = K^{(m)} \text{ a.s. ,}$$

for all $m \geq 1$. Taking $K = \max_m K^{(m)}$ we obtain that

$$\max_{0 \leq k \leq n \leq \infty} \{A(k, n) - \mathcal{G}(n-k)\} = K \text{ a.s. ,}$$

thus concluding that

$$\varepsilon(\sigma) \in \{0, 1\} \ \forall \sigma$$

in the definition of the S^{3}BB model.[4]

[4]We remark that S^{3}BB is not necessarily quasi-deterministic under the additional assumptions of restricting the 'max' operator to a finite interval $0 \leq k \leq n$ and letting $\varepsilon(\sigma)$ depend on the right margin n (see Definition 3.2.1 in [28]).

4.4.2 Non-Stationary or Non-Ergodic Processes

Here we give two examples of arrival processes for which the S^{3}BB model is not necessarily quasi-deterministic. Such processes have to be non-stationary or non-ergodic.

For an example of a non-stationary process consider

$$a_1 = r + X, \ a_n = r \ \forall n \geq 2 \ ,$$

where X is some r.v. satisfying $E[X] > 0$. Note that the cumulative process $A(n) = rn + X \ \forall n$ does not have stationary increments because $E[a_1] \neq E[a_2]$. Moreover, for $\sigma > 0$, the probability

$$\mathbb{P}\Big(\max_{0 \leq k \leq n \leq \infty} \{A(k, n) - r(n-k)\} > \sigma \Big) = \mathbb{P}\Big([X]_+ > \sigma \Big)$$

can be different from zero and one.

For an example of a non-ergodic process consider

$$a_n = X \ \forall n \geq 1 \ ,$$

for some r.v. X. The cumulative process $A(n) = nX$ is stationary but non-ergodic, as there are many realizations of the process for which the time averages are different. To construct a non-quasi-deterministic S^{3}BB model, one can take $\mathcal{G}(n) = rn$ and X be any Bernoulli r.v. with $E[X] = r$ but $\mathbb{P}(X \neq r) > 0$.

What the two examples have in common is that the sample-paths are completely determined from some time scale on. In particular, in the second example, the sample-paths are completely determined once time starts. We speculate that more compounded examples would also account for randomness but for a finite time scale *only*, in order to avoid the limiting argument in the preceding quasi-determinism proof. Due to this rather unnatural restricted capability in capturing randomness, the relevance of such models is unclear.

5. STATISTICAL MULTIPLEXING GAIN

In this section we justify the raison d'être of SNC; concretely, we present a result which rigorously reveals the magnitude of the statistical multiplexing gain, as a scaling law, achieved by SNC in the single-node case. Then we discuss on the existence of multiplexing gain in the multi-node case, and present numerical results.

5.1 Single-Node Case

Statistical multiplexing is an essential property of packet-switched networks, which are based on the principle of resource sharing. It basically says that the number of resources needed to support service for (say N) flows is *much smaller* than N times the number of resources needed to support service for a single flow. The raison d'être of SNC is to capture the gap between these two quantities, i.e., the *statistical multiplexing gain*, while closely reproducing the elegant methodology of DNC.

To illustrate the magnitude of the statistical multiplexing gain achieved with SNC we consider a node of capacity C serving N flows $A_j(n)$, each modelled with the envelope

$$A_j(k, n) \leq r(n-k) + b \ \forall \ k, n \ , \tag{28}$$

where $r > 0$ is a rate and $b \geq r$ is a burst size. Consider now the design question **Q1:** "*How large should C be such that the delay is smaller than some value, normalized to 1?*" To answer, it is convenient to derive a backlog bound. Assuming the stability condition $\rho := \frac{Nr}{C} \leq 1$, the bound follows

directly from Eq. (21), where $\mathcal{G}(k) = Nrk$:

$$B_n \leq Nb \; \forall n \; .$$

Because of the delay normalization to 1, which implies that C and the backlog scale identically, we conclude that the required capacity scales as $C = \mathcal{O}(N)$ in the burst b. Although this conclusion is based on a *tight* bound (recall the discussion from Section 4.2), the intuition is that a much smaller capacity would be sufficient under broad statistical assumptions on the flows $A_j(n)$, and as long as some violations of the delay constraint are tolerable.

Let us additionally assume that $A_j(n)$'s are stationary and statistically independent, and enforce the (tolerable) constraint $\mathbb{P}\left(\text{delay} > 1\right) \leq \varepsilon$, where ε is some small value, e.g., $\varepsilon = 10^{-3}$. With these assumptions, we can use a stochastic bound on the aggregate $A(n) := \sum_{j=1}^{N} A_j(n)$, i.e., [38]

$$\mathbb{P}\left(A(n) > Nrn + \sigma\right) \leq e^{-\frac{\sigma^2}{2Nb^2}} \; \forall \sigma \geq 0 \; . \tag{29}$$

A backlog bound can then be computed as in Eq. (19):

$$\begin{aligned} \mathbb{P}\left(B_n > \sigma\right) &\leq \mathbb{P}\left(\max_{0 \leq k < n} \left\{A(k,n) - C(n-k)\right\} > \sigma\right) \\ &\leq \sum_{k \geq 1} \mathbb{P}\left(A(k) > Nrk + (C - Nr)k + \sigma\right) \\ &\leq \int_0^\infty e^{-\frac{1}{2}\left(\frac{(C-Nr)s+\sigma}{\sqrt{N}b}\right)^2} ds \; . \end{aligned}$$

In the last line we used the bound from Eq. (29) and bounded a sum of non-increasing terms by an integral. With the change of variable $u = \frac{(C-Nr)s+\sigma}{\sqrt{N}b}$ the last term becomes

$$\frac{\sqrt{N}b}{C - Nr} \int_{\frac{\sigma}{\sqrt{N}b}}^\infty e^{-\frac{u^2}{2}} du \leq \frac{b^2}{r\left(\rho^{-1} - 1\right)\sigma} e^{-\frac{\sigma^2}{2Nb^2}} \; .$$

Here we used Gordon's inequality for the standard normal density function, i.e., $\int_x^\infty e^{-\frac{u^2}{2}} du \leq \frac{1}{x} e^{-\frac{x^2}{2}}$ [24]. Setting the last term to ε yields

$$\sigma^2 = 2Nb^2 \left(\log \frac{b^2}{\varepsilon r \left(\rho^{-1} - 1\right)} - \log \sigma\right) \; .$$

From here it follows that $C = \mathcal{O}\left(\sqrt{N}\right)$ in the burst b (recall that C scales identically with the backlog σ); note, however, that $C = \mathcal{O}(N)$ in the rate r in order to satisfy the stability condition. The $\mathcal{O}\left(\sqrt{N}\right)$ law can also be deduced from a result from [4], which is however obtained using an approximative application of the Central Limit Theorem, and hence not rigorous. Several other probabilistic bounds on multiplexed deterministically regulated arrivals (as in Eq. (28)) exist in the literature, e.g., [47, 50]; however, they do not appear to easily lend themselves to the $\mathcal{O}\left(\sqrt{N}\right)$ law.

The difference in the scaling laws $C = \mathcal{O}(N)$ (obtained with DNC) vs. $C = \mathcal{O}\left(\sqrt{N}\right)$ (obtained with SNC) reveals thus the magnitude of the statistical multiplexing gain achieved with SNC as $\Omega\left(\sqrt{N}\right)$.

5.2 Multi-Node Case

Lastly, we discuss on an unconventional type of statistical multiplexing which, to the best of our knowledge, has not been raised previously. Consider the tandem network from

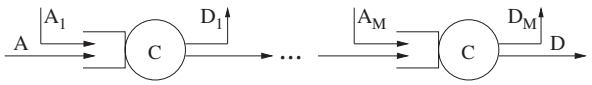


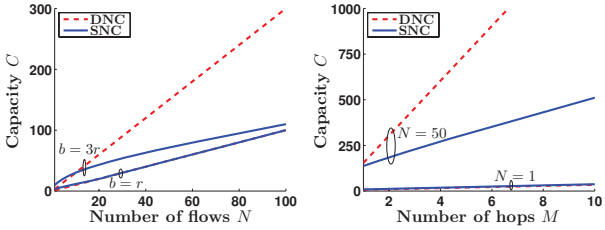
Figure 2: A tandem network with cross traffic

Figure 2 in which a flow A crosses M nodes in series; at each node $j = 1, \ldots, M$ along the end-to-end (e2e) path, A shares the local resource (the capacity C) with a local cross flow A_j. All flows are stationary and statistically independent. This type of resource sharing looks similar to the conventional one, except that the 'resource' is now a distributed one (i.e., all the capacities) and the cross flows do not share the same resource with each other. The arising question concerns the existence of a *distributed* multiplexing gain.

To answer, we apply and compare DNC and SNC for the following scenario: A is bounded by the envelope from Eq. (28) with rate r and burst b, and A_j's are bounded by the same envelope but with rate Nr and burst Nb (N will be used for tuning conventional (per-node) multiplexing gain). We enforce the stability condition $C \geq (N+1)r$ and assume that flow A gets lower priority at each node. We ask the design question **Q2**: "*How large should C be such that the e2e delay of A is smaller than 1?*".

To deal with the additional complexities due to scheduling and multi-node (i.e., the system's 'noise'), we run the network calculus engine, i.e., transform the network system into a 'somewhat looking' linear system. The first step is to derive the service processes at each node, i.e., $S^j(n) = [(C - Nr)n - Nb]_+$, and then apply the convolution formula from Eq. (13) yielding $S(n) = \left[\left(n - \frac{NMb}{C-Nr}\right)\right]_+ (C - Nr)$. From the transformed system, consisting of the input **A** and the service process **S**, the deterministic e2e delay bound is $W \leq \frac{(NM+1)b}{C-Nr}$. A probabilistic e2e delay bound can be also derived (not shown here) using the SNC formulation from Fidler [22] and the representation from Eq. (29).

Figure 3 illustrates the required capacities C, computed with DNC and SNC, for the questions **Q1** in (a) and **Q2** in (b). Note that (a) clearly shows the (conventional) multiplexing gain when $b = 3r$. Note also that the multiplexing gain kicks-in around ten flows. In turn, (b) illustrates that there is no distributed multiplexing gain when there is only one cross flow $N = 1$; however, for sufficiently large N, e.g., 50, the conventional multiplexing gain kicks-in and compensates for the lack of distributed multiplexing gain.



(a) Single-node (b) Multi-node ($b = 3r$)

Figure 3: Capacity dimensioning, with DNC and SNC, for the delay constraints (delay $\leq$ 1) and $\mathbb{P}$(delay > 1) $\leq \varepsilon$, respectively ($r = 1$, $\varepsilon = 10^{-3}$)

6. CONCLUSIONS

Network calculus (NC) has seen a lot of research over the last 20 years, e.g., Google Scholar yields $\approx$ 3000 hits when searching for "network calculus", comparing relatively well to its 'older brother' queueing theory, which yields $\approx$ 60000 hits. The broad perception on NC is that it holds good promises as an alternative/complementary methodology to classical queueing theory. Yet, in the camp of NC researchers as well as the larger audience, the mood somewhat oscillates between "Hooray, we found the Holy Grail!" and "Oh no, it's not gonna work." In this paper, we have tried to better locate where the truth stands and aimed at reducing the level of confusion—and hopefully not creating new one—by clarifying some important issues related to the core modelling abstractions in network calculus: arrival envelopes and service processes. On this mission, we also provided some new insights into network calculus in the larger frame of things. Specifically, we collected the following facts:

- NC is about approximating a complex (typically non-linear) queueing system by a min-plus linear one. The approximation closely follows the traces of both traditional LTI system theory and of elementary properties from classical queueing theory; the analogy is good, but not perfect.
- With respect to arrival envelopes we provided a number of pitfalls as well as advice on how to avoid them. The most salient and profound observation is that the S^3BB envelope model presented at ACM Sigcomm 2006 delivers quasi-deterministic performance bounds for a large class of arrivals (stationary and ergodic), which means that it cannot generally capture statistical multiplexing gain.
- Statistical multiplexing gain can be captured well by SNC, under carefully defined envelope and service processes. In particular, for the application scenario of multiplexing independent regulated arrival processes, we rigorously showed the gain to be on the order of $\Omega(\sqrt{N})$, which before had only been shown approximately. So there is light at the end of the tunnel and hopefully it is not by a train railing towards us.
- We have also discussed on the tightness of bounds, a lingering issue surrounding almost any discussion on NC. We clarified issues about the tightness of the DNC and provided insights into that of SNC. In short, DNC generally delivers tight bounds, i.e., the bounds can be attained; SNC is as tight as the underlying probability inequalities being used (Boole, Chernoff, martingale inequalities). As these inequalities can be plugged into SNC in a modular fashion, one may argue that the SNC analysis can be made as tight as the state-of-the-art in probability theory allows.

Summing up, while using NC does not provide us with a free lunch, it still seems to be of good value in analyzing traditionally hard fundamental queueing problems due to its scalable tradeoff between accuracy and ease of analysis.

7. ACKNOWLEDGMENTS

We thank the following people for their constructive influence on this manuscript: Jörg Liebeherr, Almut Burchard, Felix Poloczek, Yuming Jiang, Yashar Ghiassi-Farrokhfal, Markus Fidler, Hao Wang, Karthik Rajkumar, and all the students from the Stochastic Network Calculus class given at TU Berlin (WS/11-12). We also thank our shepherd Mark Crovella and the anonymous Sigcomm reviewers for their comments.

8. REFERENCES

[1] R. Agrawal, R. L. Cruz, C. Okino, and R. Rajan. Performance bounds for flow control protocols. *IEEE/ACM Transactions on Networking*, 7(3):310–323, June 1999.

[2] F. L. Baccelli, G. Cohen, G. J. Olsder, and J.-P. Quadrat. *Synchronization and Linearity: An Algebra for Discrete Event Systems*. John Wiley and Sons, 1992.

[3] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios. Open, closed and mixed networks of queues with different classes of customers. *Journal of the ACM*, 22(2):248–260, Apr. 1975.

[4] R. Boorstyn, A. Burchard, J. Liebeherr, and C. Oottamakorn. Statistical service assurances for traffic scheduling algorithms. *IEEE Journal on Selected Areas in Communications. Special Issue on Internet QoS*, 18(12):2651–2664, Dec. 2000.

[5] J.-Y. Le Boudec and P. Thiran. *Network Calculus*. Springer Verlag, Lecture Notes in Computer Science, LNCS 2050, 2001.

[6] R. Braden, D. Clark, and S. Shenker. Integrated services in the Internet architecture: an overview. IETF RFC 1633, July 1994.

[7] L. Breiman. *Probability*. SIAM, 1992.

[8] A. Burchard, J. Liebeherr, and F. Ciucu. On superlinear scaling of network delays. *IEEE/ACM Transactions on Networking*, 19(4):1043–1056, Aug. 2011.

[9] A. Burchard, J. Liebeherr, and S. D. Patek. A min-plus calculus for end-to-end statistical service guarantees. *IEEE Transactions on Information Theory*, 52(9):4105–4114, Sept. 2006.

[10] S. Chakraborty, S. Kuenzli, L. Thiele, A. Herkersdorf, and P. Sagmeister. Performance evaluation of network processor architectures: Combining simulation with analytical estimation. *Computer Networks*, 41(5):641–665, Apr. 2003.

[11] C.-S. Chang. Stability, queue length and delay, Part II: Stochastic queueing networks. In *31st IEEE Conference on Decision and Control*, pages 1005–1010, 1992.

[12] C.-S. Chang. *Performance Guarantees in Communication Networks*. Springer Verlag, 2000.

[13] F. Ciucu. Network calculus delay bounds in queueing networks with exact solutions. In *International Teletraffic Congress (ITC)*, pages 495–506, 2007.

[14] F. Ciucu, A. Burchard, and J. Liebeherr. Scaling properties of statistical end-to-end bounds in the network calculus. *IEEE Transactions on Information Theory*, 52(6):2300–2312, June 2006.

[15] C. Courcoubetis and R. Weber. Effective bandwidths for stationary sources. *Probability in Engineering and Informational Sciences*, 9(2):285–294, Apr. 1995.

[16] R. Cruz. A calculus for network delay, parts I and II. *IEEE Transactions on Information Theory*, 37(1):114–141, Jan. 1991.

[17] R. L. Cruz. Quality of service management in integrated services networks. In *1st Semi-Annual Research Review, CWC, University of California at San Diego*, June 1996.

[18] R. L. Cruz. SCED+: Efficient management of quality of service guarantees. In *IEEE Infocom*, pages 625–634, 1998.

[19] R. L. Cruz and C. Okino. Service guarantees for window flow control. In *34th Allerton Conference on Communications, Control and Computating*, 1996.

[20] A. Erlang. Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges. *Elektrotkeknikeren*, 13, 1917 (in Danish).

[21] W. Feller. *An Introduction to Probability Theory and Its Applications, Vol. 2*. Wiley, 1971.

[22] M. Fidler. An end-to-end probabilistic network calculus with moment generating functions. In *IEEE International Workshop on Quality of Service (IWQoS)*, pages 261–270, 2006.

[23] M. Fidler. A survey of deterministic and stochastic service curve models in the network calculus. *IEEE Communications Surveys & Tutorials*, 12(1):59–86, Feb. 2010.

[24] R. D. Gordon. Values of Mills' ratio of area to bounding ordinate and of the normal probability integral for large values of the argument. *Annals of Mathematical Statistics*, 12(3):364–366, Sept. 1941.

[25] M. Harchol-Balter. *Performance Modeling and Design of Computer Systems. Queueing Theory in Action*. Cambridge University Press, 2012.

[26] Y. Jiang. A basic stochastic network calculus. In *ACM Sigcomm*, pages 123–134, Sept. 2006.

[27] Y. Jiang. Network calulus and queueing theory: Two sides of one coin. In *ICST ValueTools*, 2009.

[28] Y. Jiang and Y. Liu. *Stochastic Network Calculus*. Springer, 2008.

[29] F. P. Kelly. Networks of queues with customers of different types. *Journal of Applied Probability*, 3(12):542–554, Sept. 1975.

[30] J. Kurose. On computing per-session performance bounds in high-speed multi-hop computer networks. In *ACM Sigmetrics*, pages 128–139, 1992.

[31] E. A. Lee and P. Varaiya. *Structure and interpretation of signals and systems*. Addison-Wesley, 2003.

[32] 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*, 2(1):1–15, Feb. 1994.

[33] L. Lenzini, E. Mingozzi, and G. Stea. A methodology for computing end-to-end delay bounds in FIFO-multiplexing tandems. *Performance Evaluation*, 65(11-12):922–943, Nov. 2008.

[34] J. Liebeherr, A. Burchard, and F. Ciucu. Delay bounds in communication networks with heavy-tailed and self-similar traffic. *IEEE Transactions on Information Theory*, 58(2):1010–1024, Feb. 2012.

[35] J. Liebeherr, M. Fidler, and S. Valaee. A system-theoretic approach to bandwidth estimation. *IEEE/ACM Transactions on Networking*, 18(4):1040–1053, Aug. 2010.

[36] J. Liebeherr, Y. Ghiassi-Farrokhfal, and A. Burchard. On the impact of link scheduling on end-to-end delays in large networks. *IEEE Journal on Selected Areas in Communications*, 29(5):1009–1020, May 2011.

[37] S. Mao and S. S. Panwar. A survey of envelope processes and their applications in quality of service provisioning. *IEEE Communications Surveys & Tutorials*, 8(3):2–20, Dec. 2006.

[38] L. Massoulié and A. Busson. Stochastic majorization of aggregates of leaky bucket-constrained traffic streams. *Preprint, Microsoft Research*, 2000.

[39] V. Misra. Public review on 'A Basic Stochastic Network Calculus' by Y. Jiang, ACM Sigcomm 2006. `http://conferences.sigcomm.org/sigcomm/2006/discussion/showpaper.php?paper_id=14`.

[40] V. Paxson and S. Floyd. Wide-area traffic: The failure of Poisson modelling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, June 1995.

[41] J.-L. Scharbarg, F. Ridouard, and C. Fraboul. A probabilistic analysis of end-to-end delays on an AFDX avionic network. *IEEE Transactions on Industrial Informatics*, 5(1):38–49, Feb. 2009.

[42] J. Schmitt and I. Martinovic. Demultiplexing in network calculus - a stochastic scaling approach. In *International Conference on the Quantitative Evaluation of Systems (QEST)*, pages 217–226, 2009.

[43] J. Schmitt, F. A. Zdarsky, and M. Fidler. Delay bounds under arbitrary multiplexing: When network calculus leaves you in the lurch... In *IEEE Infocom*, pages 1669–1677, 2008.

[44] T. Skeie, S. Johannessen, and O. Holmeide. Timeliness of real-time IP communication in switched industrial ethernet networks. *IEEE Transactions on Industrial Informatics*, 2(1):25–39, Feb. 2006.

[45] D. Starobinski and M. Sidi. Stochastically bounded burstiness for communication networks. *IEEE Transactions on Information Theory*, 46(1):206–212, Jan. 2000.

[46] M. Talagrand. Majorizing measures: The generic chaining. *Annals of Probability*, 24(3):1049–1103, July 1996.

[47] M. Vojnovic and J.-Y. Le Boudec. Bounds for independent regulated inputs multiplexed in a service curve network element. *IEEE Transactions on Communications*, 51(5):735–740, May 2003.

[48] K. Wang, F. Ciucu, S. Low, and C. Lin. A stochastic power network calculus for integrating renewable energy sources into the power grid. *IEEE Journal on Selected Areas in Communications; Smart Grid Communications Series*, 30(6), July 2012.

[49] O. Yaron and M. Sidi. Performance and stability of communication networks via robust exponential bounds. *IEEE/ACM Transactions on Networking*, 1(3):372–385, June 1993.

[50] Y. Ying, F. Guillemin, R. Mazumdar, and C. Rosenberg. Buffer overflow asymptotics for multiplexed regulated traffic. *Performance Evaluation*, 65(8):555–572, July 2008.