Performance Modelling and Analysis of Unreliable Links with Retransmissions using Network Calculus

Hao Wang and Jens Schmitt
DISCO Labs, TU Kaiserslautern, Germany
{wang,jschmitt}@informatik.uni-kl.de

Florin Ciucu
T-Labs / TU Berlin, Germany
florin@t-labs.tu-berlin.de

***Abstract*—During the last two decades, starting with the seminal work by Cruz, network calculus has evolved as an elegant system theory for the performance analysis of networked systems. It has found numerous usages as, for example, in QoS-enabled networks, wireless sensor networks, switched Ethernets, avionic networks, Systems-on-Chip, or, even to speed-up simulations. One of the basic assumptions in network calculus is that links are reliable and operate loss-free. This, of course, is a major abstraction from the reality of many application scenarios, where links are unreliable and often use retransmission schemes to recover from packet losses. As of today, standard network calculus cannot analyze such links.**

In this paper, we take the challenge to extend the reach of network calculus to unreliable links which employ retransmission-based loss recovery schemes. Key to this is a stochastic extension of the known data scaling element in network calculus [21], which can capture the loss process of an unreliable link. Based on this, modelling links with retransmissions results in a set of equations which are amenable to a fixed-point solution. This allows to find the arrival constraints of each flow that corresponds to a certain number of retransmissions. Based on the description of each retransmission flow, probabilistic performance bounds can be derived. After providing the necessary theory, we illustrate this novel and important extension of network calculus with the aid of a numerical example.

I. Introduction

A. Motivation

Network calculus is a min-plus system theory for deterministic queuing systems which builds on the calculus for network delay in [14], [15]. The important concept of *service curve* was introduced in [16], [31], [9], [28], [2]. The service curve based approach facilitates the efficient analysis of tandem queues where a linear scaling of performance bounds in the number of traversed queues is achieved as elaborated in [11] and also referred to as pay bursts only once phenomenon in [29]. A detailed treatment of min-plus algebra and of network calculus can be found in [4] and [29], [10].

Network calculus has found numerous applications, most prominently in QoS-enabled networks, e.g., based on the IntServ or DiffServ architecture, but also in other scenarios as, for example, wireless sensor networks [32], [26], switched Ethernets [34], Systems-on-Chip (SoC) [8], network coding [37], information theory [30], or even to speed-up simulations [25]. Hence, besides queueing theory it has established as a valuable methodology for the performance analysis of networked systems.

However, as a relatively young theory, compared to traditional queueing theory, there are also a number of challenges which network calculus still has to master. To name a few: there has been much interest and progress with respect to stochastic extensions (see [11], [19], [23] for basic approaches and [12] for a recent perspective); tool support for network calculus has been addressed by [33], [6] and brings about new interesting perspectives. A very tough challenge is also found in applying network calculus in network scenarios, where links (or servers) may be unreliable and some packets are lost. This actually shatters network calculus in one of its foundations which is the assumption of a lossless system operation. Yet, the key of our approach in dealing with such losses lies in the so-called data scaling element (of which a deterministic and stochastic variant have been proposed to accommodate flow transformations inside the network [21], [13]), and the simple realization that loss is just a specific flow transformation. To make things even more compounded, unreliable links usually employ retransmission-based loss recovery schemes, which produces a *feedback* cycle between in- and output – another difficult issue for the analysis.

In this paper, we tackle these challenges and specifically make the following contributions:

- We set up a stochastic network calculus model for an unreliable link employing retransmissions for loss recovery.
- We provide a new definition of a stochastic scaling curve that allows for a two-step procedure to solve the model by decoupling deterministic and stochastic arguments.
- For a specific loss process, the binary symmetric channel, we determine a tight stochastic scaling curve based on a Martingale argument.
- We solve the feedback cycle problem under the realistic assumption of a limited number of retransmissions.

B. Related Work

There is a set of results that deal with lossy systems in the framework of network calculus [3], [18], yet these deal with losses due to a finite buffer or a specific scheduling discipline that tries to take advantage from some loss tolerance, so these are only seemingly related to our work on unreliable links with retransmissions.

There is only little previous research that really tried to transfer network calculus concepts into the domain of networks with unreliable links: Remarkably, one of the earliest research on stochastic extensions of network calculus [17] already introduced a "channel impairment process" to model a time-varying channel capacity as well as retransmission-based loss recovery. The model, however, is very much tied to a specific scheduling algorithm (SCED) to operate on the link. At the time, the concept of scaling, which we use in this paper to capture the loss characteristics of unreliable links, had not yet been introduced in network calculus.

Another related work [1] investigates a network calculus model of a wireless link using the concept of a clipper [18] to capture data loss. The issue of operating the wireless link with a retransmission scheme is briefly discussed but not investigated in further detail, in particular the influence on the overall service capacity is not taken into account. The paper also remains unclear how the deterministic clipper component can actually capture the stochastic loss behavior of a wireless channel. In [35], the authors introduce a so-called effective capacity model of a wireless link, in analogy to the notion of effective bandwidth. The goal was to capture the time-varying characteristics of a fading channel. Losses and corresponding retransmissions were not considered. Similarly, though following a different analytical approach, [20] focused on the characterization of a fading channel with memory. In this work, a network calculus based on moment-generating functions was used to calculate probabilistic performance bounds. Again, the issue of data loss and retransmissions were not subject of this research.

In a recent study [36], data loss has explicitly been taken into account and abstracted as a stochastic process. As in [17], the introduced error process was composed into the server, in order to propose the so-called error server model. This is used in a subsequent work [22] in the context of cognitive radio networks to model all the retransmission processes together. Yet, only the virtual delay of the original flow can be calculated. We believe this approach to lose flexibility compared to using scaling for explicitly modelling the data loss process. A similar method is used in [27], where the retransmitted units must be analyzed together leveraging a free parameter. Again, only the virtual delay of the original arrivals is calculated. In our work, we focus on modelling losses due to unreliable transmission and the influence of retransmission schemes to compensate for these losses. With such a model, the delay of either the single flow (the original and the retransmitted flows) or the aggregate flow can be derived.

C. Outline

The rest of the paper is organized as follows. In the next section we recall some important network calculus concepts and results related to data scaling. In Section III we extend deterministic scaling into a probabilistic framework and give an example of a stochastic scaling process. In Section IV we present and analyze a model for an unreliable link with retransmission-based loss recovery, using our model for stochastic data scaling. In Section V we give a numerical example and in Section VI we wrap up the paper.

II. Preliminaries on Network Calculus and Data Scaling

In this section, we provide the necessary background on deterministic network calculus (see Le Boudec and Thiran [29] or Chang [10]), and also on some prior work related to data scaling.

A. Deterministic Network Calculus

Consider the set of real-valued, non-negative, and wide-sense increasing functions:

$$\mathcal{F} = \left\{ f : \mathbb{R}^+ \to \mathbb{R}^+, \forall t \geq s : \ f(t) \geq f(s), \ f(0) = 0 \right\} \ .$$

A flow is defined in terms of an arrival process $A(t)$ and a departure process $D(t)$, counting the number of input and output packets in some queueing system. Note that, for any sample-path, $A(t), D(t) \in \mathcal{F}$. The flow's *backlog* process at time t is defined as

$$b(t) = A(t) - D(t) \ .$$

The flow's *virtual delay*, more exactly of a packet (if any) arriving at time t, is defined as

$$d(t) = \inf \left\{ \tau \geq 0 : \ A(t) \leq D(t+\tau) \right\} \ .$$

Network calculus uses bounds for both arrivals and service. These are described in a min-plus algebra with the *convolution* operator $\otimes$ defined as

$$(f \otimes g)(t) = \inf_{0 \leq s \leq t} \{ f(t-s) + g(s) \} \ ,$$

for $f, g \in \mathcal{F}$. A bound on an arrival process A is defined as an *arrival curve* function $\alpha \in \mathcal{F}$ such that for $\forall t, s \geq 0, s \leq t:$

$$A(t) - A(t-s) \leq \alpha(s) \Leftrightarrow A \leq A \otimes \alpha \ .$$

In turn, a bound on the service provided by a system to an arrival process $A(t)$ is defined as a *service curve* function β such that for $\forall t \geq 0$

$$D(t) \geq A \otimes \beta(t) \ ,$$

where $D(t)$ is the corresponding departure process of $A(t)$.

From these bounds, network calculus provides bounds on backlog and delay processes, and also on the output flow:

$$\begin{aligned} &\text{backlog: } \forall t : \ b(t) \leq (\alpha \oslash \beta)(0) =: v(\alpha, \beta) \ , \\ &\text{delay: } \forall t : \ d(t) \leq \sup_{s \geq 0} \{ \inf \{ u \geq 0 : \alpha(s) \leq \beta(s+u) \} \} \\ &\qquad\qquad\qquad\qquad\qquad\qquad =: h(\alpha, \beta) \ , \\ &\text{output (arrival curve } \alpha' \text{for } D(\text{t})\text{): } \alpha' = \alpha \oslash \beta \ . \end{aligned}$$

Here, $\oslash$ is the *deconvolution* operator in the min-plus algebra defined as

$$(f \oslash g)(t) = \sup_{u \geq 0} \{ f(t+u) - g(u) \} \ ,$$

for $f, g \in \mathcal{F}$. All relationships so far hold for all sample-paths, or almost surely (a.s.), as stochastic processes (e.g., $A(t)$ or $b(t)$) are bounded with deterministic functions (e.g., $\alpha(t)$ or $v(\alpha, \beta)$).

An important subclass of service curves are the so-called *strict* service curves. If a system guarantees an input flow at least $\beta(u)$ service during any backlogged period of size u, then by definition $\beta(u)$ is a strict service curve. In this paper strict service curves will be used to calculate left-over service curves at a node serving two flows H and L, with non-preemptive priority given to H. If the node guarantees a strict service curve β to the aggregate of the two flows, and if α_H is the arrival curve of flow H, then the left-over service curve for flow L is

$$\beta_L(t) = [\beta(t) - \alpha_H(t)]^+ \ .$$

Moreover, the service curve guaranteed to flow H is $\beta_H(t) = [\beta(t) - l^L_{max}]^+$, where l^L_{max} is the maximum packet size for flow L, where $[x]^+ := \max\{x, 0\}$ for some number x.

B. Data Scaling

Here we review scaling elements in network calculus, as introduced in Fidler and Schmitt [21].

Definition 1. (Scaling Process) A scaling process $S \in \mathcal{F}$ *scales* a value a to a value $S(a)$.

The value a has the physical interpretation of data/packet count. Scaling processes are stochastic processes and are quite general in that they can model various data transformations in networks [13]. The physical process of *scaling* occurs infinitely fast, and scaling processes do not deal with queueing effects (still to be modelled separately by service curves).

Following the bounding principle of deterministic network calculus, (stochastic) scaling processes are modelled with (deterministic) bounds in order to increase their scope.

Definition 2. (Scaling Curve) A function $\overline{S} \in \mathcal{F}$ is said to be a maximum scaling curve for some scaling process S if

$$\overline{S} \geq S \oslash S \text{ (a.s.)}$$

The next lemma characterizes the effects of data scaling on an arrival flow constrained with an arrival curve.

Lemma 1. *(Scaling Constrained Arrivals) [21] Let $A(t)$ be an arrival process with arrival curve α. If a scaling process S, with maximum scaling curve $\overline{S}$, is applied to the arrival process $A(t)$, then the scaled output has the arrival curve $\alpha_S = \overline{S}(\alpha)$.*

III. Stochastic Data Scaling

In this section, we briefly review the stochastic network calculus (see also Jiang and Liu [24]), and then extend the data scaling model in a probabilistic framework.

This extension is motivated herein by the stochastic behavior of unreliable links. Our key idea is to model the data lost on an unreliable link as the *scaled* version of the output flow from a fully reliable link. In other words, we first assume lossless transmissions, and then make up for this assumption by removing the (actually lost) data units with a *scaling process*. Note that a deterministic scaling curve for the scaling process, as in Definition 2, would clearly render loose bounds.

A stochastic scaling model was introduced in a previous work of ours [13]. It consists of an arrival process $A(t)$, a scaling random process $\mathbf{X} = (X_i)_{i\geq 1}$ taking non-negative integer values, and a scaled process $A^{\mathbf{X}}(t)$ defined for all $t \geq 0$ as a compound process for all $t \geq 0$

$$A^{\mathbf{X}}(t) = \sum_{i=1}^{A(t)} X_i \ .$$

This pointwise characterization can be in principle used to the probabilistic analysis of unreliable links with retransmissions, but in a very tedious manner. To simplify the calculus, we propose a weaker *stochastic scaling curve* model which can lend itself to a more efficient probabilistic analysis by first making a sample-path deterministic argument, and then evaluating the probability of violating the underlying sample-path assumption.

Before introducing our stochastic scaling curve model, as a probabilistic extension of the one from Definition 2, we review similar probabilistic extensions of deterministic network calculus concepts [7]. A function $\alpha^\epsilon \in \mathcal{F}$ is called a stochastic sample-path arrival curve for an arrival process $A(t)$ if for $\forall t \geq 0$

$$P\left(A(t) \leq (A \otimes \alpha^\epsilon)(t)\right) \geq 1 - \epsilon \ .$$

A queueing system offers a stochastic service curve β^ϵ to some arrival process $A(t)$ if the corresponding departure process $D(t)$ satisfies for $\forall t \geq 0$ that

$$P\left(D(t) \geq (A \otimes \beta^\epsilon)(t)\right) \geq 1 - \epsilon \ .$$

Here, ϵ is the violation probability of the bounds.

A. Stochastic Scaling Curves

Here we introduce probabilistic scaling curves, based on which we can model realistic loss processes at unreliable links and yet preserve the simplicity of the calculus.

Definition 3. (Stochastic Scaling Curve) Consider a scaling process S. A function $\overline{S}^{\overline{\varepsilon}} \in \mathcal{F}$ is said to be a (maximum) stochastic scaling curve of S if for all $b \geq 0$

$$P\left(\sup_{0\leq a\leq b}\left\{S(b) - S(a) - \overline{S}^{\overline{\varepsilon}}(b-a)\right\} \leq 0\right) \geq 1 - \overline{\varepsilon} \ ,$$

where $\overline{\varepsilon}$ denotes the violation probability for the stochastic maximum scaling curve.

Note that the stochastic scaling curve property provides a sample-path bound for the scaling process. As a side remark, the definition may be slightly generalized by moving the supremum outside of the probability measure, but at the expense of loosing the sample-path property and consequently complicating the calculus. Note also that a sample-path stochastic scaling curve from Definition 3 extends the scaling curve from Definition 2 by mimicking the extension of arrival curves to stochastic sample-path arrival curves.

The next lemma mimics Lemma 1 in a probabilistic setting.

Lemma 2. *(Scaling of Stochastic Constrained Arrivals) Let $A(t)$ be an arrival process with sample-path arrival curve α^{ϵ_α}. If a scaling process S, with sample-path stochastic scaling curve $\overline{S}^{\overline{\varepsilon}}$, is applied to the arrival process $A(t)$, then the scaled output has the sample-path arrival curve $\alpha_S = \overline{S}^{\overline{\varepsilon}}(\alpha^{\epsilon_\alpha})$ with probability $\geq 1 - \overline{\varepsilon} - \epsilon_\alpha$.*

Proof: Assume that, on some sample-path, both the stochastic arrival curve α^{ϵ_α} and the scaling curve $\overline{S}^{\overline{\varepsilon}}$ are not violated. On this sample-path, Lemma 1 yields that $\alpha_S = \overline{S}^{\overline{\varepsilon}}(\alpha^{\epsilon_\alpha})$ is an arrival curve for the (sample-path) scaled output, denoted by $S(A(t))$. Accounting now for the violation probability of the sample-path assumption, we get

$$\begin{aligned} & P\left(S(A(t)) \leq \left(S(A) \otimes \overline{S}^{\overline{\varepsilon}}(\alpha^{\epsilon_\alpha})\right)(t)\right) \\ & \geq \quad P\left(\{\alpha^{\epsilon_\alpha} \text{ not violated}\} \wedge \left\{\overline{S}^{\overline{\varepsilon}} \text{ not violated}\right\}\right) \\ & \geq \quad 1 - \overline{\varepsilon} - \epsilon_\alpha \end{aligned} \tag{1}$$

due to Boole's inequality, which completes the proof. ■

We point out that we used a seemingly loose bound in Eq. (1) because the arrival and scaling processes may be correlated; in our calculus, such a correlation will be determined by *retransmission* processes which are assigned higher priorities than the original process.

B. Modelling a Binary Symmetric Channel (BSC)

To exemplify how the loss process at an unreliable link can be captured with stochastic scaling, we consider a binary symmetric channel (BSC) model; more complicated channel models can also be considered (see [13] for loss processes defined as Markov arrival processes). For the ease of presentation, and also to deal with inherent technical complications due to modelling retransmissions, we mainly focus on the BSC model.

We model the BSC with the scaling process

$$S(b) = X_1 + X_2 + \cdots + X_b \ , \tag{2}$$

where $X_i's \in \{0,1\}$ are *i.i.d.* Bernoulli random variables (r.v.) with parameter p, i.e., the crossover probability of the BSC. The next theorem gives the stochastic scaling curve for the BSC, which will be used later.

Theorem 1. *(*STOCHASTIC SCALING CURVE FOR BSC*) Consider the scaling process $S(b)$ for BSC from Eq. (2). Then the function*

$$\overline{S}^{\overline{\varepsilon}}(b) = pb + 1 - \overline{\varepsilon} \ . \tag{3}$$

is a stochastic scaling curve for BSC, in the sense of Definition 3.

Proof: Let us first construct the process

$$T(c) := S(c) - \overline{S}^{\overline{\varepsilon}}(c) + 1 \ \forall c \ .$$

Construct also the filtration $\mathcal{F}_c = \sigma(X_1, X_2, \ldots, X_c)$ capturing the partial histories of the process X_c. Then we have for some fixed c:

$$\begin{aligned} E\left[T(c+1) \mid \mathcal{F}_c\right] &= E\left[T(c) + X_{c+1} - p \mid \mathcal{F}_c\right] \\ &= T(c) + E\left[X_{c+1} - p\right] \\ &= T(c) \ , \end{aligned}$$

and thus $T(c)$ is a martingale. In the second line we used that $T(c)$ is $\mathcal{F}_c$-measurable, and that X_{c+1} is independent of $\mathcal{F}_c$. The last line follows since X_{c+1} is a Bernoulli r.v. with parameter p.

Since $T(c)$ is a martingale we can write for all $b \geq 0$:

$$\begin{aligned} & P\left(\sup_{0 \leq a \leq b} \left\{S(b) - S(a) - \overline{S}^{\overline{\varepsilon}}(b-a)\right\} \geq 0\right) \\ &= P\left(\sup_{0 \leq c \leq b} T(c) \geq 1\right) \\ &\leq E\left[T(0)\right] \\ &= \overline{\varepsilon} \ . \end{aligned}$$

In the second line we used the stationarity of the Bernoulli process. In the third line we used Doob's maximal inequality (see Billingsley [5], p. 466), applied to the martingale $T(c)$. ■

This result will be used to compute the arrival curves of retransmission processes.

IV. A Network Calculus Model of an Unreliable Link with Retransmission-based Loss Recovery

In this section, we propose a model for an unreliable link that employs a retransmission scheme to recover from data loss due to channel impairments. The model builds upon the novel concept of stochastic scaling as introduced in the previous section. Based on this model and existing results of network calculus we provide a method to derive the arrival curves of the retransmitted flows. This method applies a fixed-point approach from which we also obtain stability criteria for the overall system. Knowing the arrival curves of all the flows in the system, we are able to derive bounds on backlog, delay, and output. These bounds are probabilistic in nature due to the stochasticity of the scaling.

A. Basic Model and General Assumptions

Data loss is a frequent event when transmitting data over an unreliable link. In order to compensate for data loss, many protocols and methods employ retransmission schemes. The data loss process can be captured in a network calculus model by a scaling process S and its stochastic scaling curve $\overline{S}^{\overline{\varepsilon}}$ as elaborated in the example of a BSC channel in Section III-B. Before retransmitting a lost packet, we assume that the sender must wait to be certain that the packet has really been lost. Thus retransmitted packets will experience a certain delay. To cover typical retransmission schemes we model two mechanisms to detect such a loss event: a local countdown timer at the sender if no positive acknowledgment is received, and, optionally, an explicit negative acknowledgment from the receiver. Thus the delay experienced before a retransmission is performed is upper bounded by the countdown timer, but may be lower if a negative acknowledgment is received before timer expiration. We further assume the countdown timer to be set to a fixed value. Clearly, loss detection may not be perfect resulting in duplicate data packets. We assume that duplicate packets can be identified by the receiver (e.g., through using sequence number). In any case, due to the countdown timer or a negative acknowledgement received at the sender side, we can abstractly model the retransmission as a data flow being fed back to the sender.

Concretely, modelling the retransmissions of the lost data units consists of 1) a feedback loop of the lost data units to the entrance of the server, 2) a feedback delay each retransmitted packet potentially experiences, and 3) a deterministic strict service curve β for characterizing the service capacity available to the aggregate of original and retransmitted units. We point out that the deterministic nature of the service curve

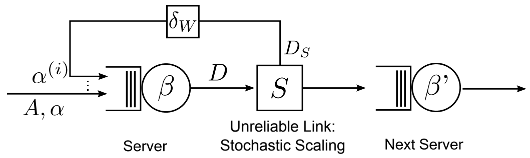

Figure 1. Network calculus model of an unreliable link with retransmissions.

precludes accounting for the possibly time-varying nature of the unreliable channel's capacity; while such more realistic scenarios can be captured using a stochastic service curve (see, e.g., [20]), we consider the simplified deterministic model for ease of exposition. Moreover, to model the feedback delay, we know that the waiting time of either the countdown timer or the potential negative acknowledgement is bounded by a maximum feedback delay denoted as W. Thus, the delay process has a service curve δ_W ([29]), with $W \geq 0$ and $\delta_W(t) = 0$ if $t \leq W$, otherwise $\delta_W(t) = \infty$. Note that retransmitted data units may have to be retransmitted again due to new loss, resulting in further retransmitted flows. We make the assumption, which holds true in most practical implementations, that the number of retransmissions is limited to some fixed value. This model of an unreliable link with retransmission-based loss recovery is depicted in Figure 1.

We distinguish between different retransmission flows, those consisting of data units being retransmitted once, twice, and so on. Correspondingly, we represent all flows in the system with their arrival curves $\alpha^{(0)} = \alpha$, $\alpha^{(1)}$, $\cdots$, $\alpha^{(N)}$, where N is the limit of the number of retransmissions for any data unit. Denote by S_i the scaling process of flow $i-1$ and by $\overline{S}_i^{\overline{\varepsilon_i}}$ the corresponding scaling envelope, where $i \geq 1$. Note that the S_i's form a partition of the overall scaling process S and $\sum_{i=1}^{N} S_i = S$. Here, $\overline{\varepsilon_i}$ is the violation probability of S_i. We also make the assumption that retransmission flow i, consisting of the data units retransmitted for i times, has lower priority than retransmission flow $i+1$, for all $i = 1, \ldots, N-1$. One instance of this policy is exemplified by a simple stop-and-wait protocol. More generally, any ARQ protocol that sends data units with lower sequence numbers first satisfies this assumption, for example, TCP does, too.

In the following, we first state the general problem of deriving the arrival curves for each of the retransmission flows based on the general assumptions just presented (Section IV-B). This general problem reduces to solving a fixed-point problem, whose analytical tractability requires imposing further assumptions, to be discussed in Section IV-C. As solving the fixed-point problem is done under deterministic assumptions, or put differently, for sample-paths that do not violate the respective scaling curves, we finally derive probabilistic bounds on the performance measures in Section IV-D.

B. Arrival Curves for Retransmission Flows: General Problem Statement

Under the general assumptions of Section IV-A, we can use existing network calculus results on priority multiplexing [29] to obtain the following formulations for the service and arrival curves of each retransmission flow. Under the assumption that the node offers a strict service curve β, using the output bound in Section II-A and Lemma 1, each flow i is offered a service curve $\beta^{(i)}$ defined as follows

$$
\begin{aligned}
&\beta^{(0)} = [\beta - \sum_{k=1}^{N} \alpha^{(k)}]^+ \ , \quad \alpha^{(0)} = \alpha \\
&\beta^{(1)} = [\beta - \sum_{k=2}^{N} \alpha^{(k)}]^+ \ , \quad \alpha^{(1)} = \overline{S}_1^{\overline{\varepsilon_1}}(\alpha^{(0)} \oslash \beta^{(0)}) \oslash \delta_W \\
&\beta^{(2)} = [\beta - \sum_{k=3}^{N} \alpha^{(k)}]^+ \ , \quad \alpha^{(2)} = \overline{S}_2^{\overline{\varepsilon_2}}(\alpha^{(1)} \oslash \beta^{(1)}) \oslash \delta_W \\
&\qquad \cdots \qquad\qquad\qquad\qquad \cdots \\
&\beta^{(N)} = \beta \ , \qquad \alpha^{(N)} = \overline{S}_N^{\overline{\varepsilon_N}}(\alpha^{(N-1)} \oslash \beta^{(N-1)}) \oslash \delta_W \ .
\end{aligned}
$$

We point out that, with a small loss of generality, we ignored packetization effects in the expressions of $\beta^{(i)}$'s. We can further simplify and remove the dependency of the $\alpha^{(i)}$'s and the $\beta^{(i-1)}$'s as follows

$$
\begin{aligned}
\alpha^{(0)} &= \alpha \\
\alpha^{(1)} &= \overline{S}_1^{\overline{\varepsilon_1}} \left(\alpha^{(0)} \oslash [\beta - \alpha^{(1)} - \alpha^{(2)} - ... - \alpha^{(N)}]^+ \right) \oslash \delta_W \\
\alpha^{(2)} &= \overline{S}_2^{\overline{\varepsilon_2}} \left(\alpha^{(1)} \oslash [\beta - \alpha^{(2)} - \alpha^{(3)} - ... - \alpha^{(N)}]^+ \right) \oslash \delta_W \\
&\cdots \\
\alpha^{(N)} &= \overline{S}_N^{\overline{\varepsilon_N}} \left(\alpha^{(N-1)} \oslash [\beta - \alpha^{(N)}]^+ \right) \oslash \delta_W \ .
\end{aligned}
$$

From these equations, it is apparent that the different arrival flows are dependent on each other and thus a probabilistic interpretation of the curves would need to take into account the corresponding correlations. However, since, in the first step, we argue purely deterministically this becomes no technical problem. The deterministic argument is under the assumption that we are on a sample path of the system for which the scaling curves are not violated. Only in the second step, in Subsection IV-D, when we evalutate the probability of this event, we reason stochastically, yet then the correlations between the arrival flows pose no technical problem any more.

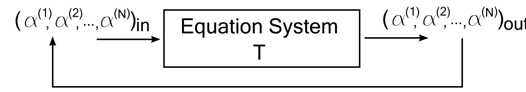

Figure 2. Self-dependent equation system.

Our goal next is to find explicit formulations of $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(N)}$ using this recursive set of equations (in the $\alpha^{(i)}$'s only, as the $\beta^{(i)}$'s were just removed). We use a fixed-point approach to resolve the self-dependency issue of the $\alpha^{(i)}$'s; see also Figure 2 for a graphical representation of the recursion system. Thereby we interpret our equation system as a mapping

$$T(\alpha_n) = \alpha_{n+1} \ ,$$

where $\alpha_n = (\alpha_n^{(1)}, \alpha_n^{(2)}, ..., \alpha_n^{(N)})$. To find a fixed-point for T we are going to rely on further assumptions on the shape of the functions α_i's, as elaborated next.

C. Arrival Curves for Retransmission Flows: Fixed-Point Calculation

In this subsection, we tackle the fixed-point problem just described by making further assumptions to instantiate a form of the problem that can actually be solved. The assumptions are simplifying but at the same time they are realistic and some of them are actually without loss of generality.

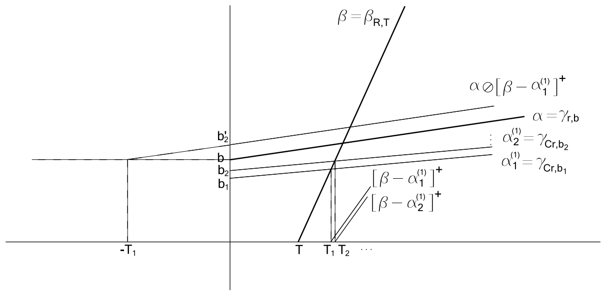

Figure 3. Illustration of the calculation for one retransmission flow.

1) Further Assumptions: We assume an affine arrival curve (a token bucket) for the original input flow $\alpha = \gamma_{r,b}$ (i.e., $\alpha(t) = rt + b$), and a rate-latency function as the strict service curve $\beta = \beta_{R,T}$ (i.e., $\beta(t) = R[t-T]^+$). The feedback is assumed to be delayed, so we use a positive W for the service curve δ_W of the feedback delay process. With respect to the scaling curves, we set $\forall i = 1, \ldots, N : \overline{S}_i^{\overline{\varepsilon}_i}(b) = \overline{S}^{\overline{\varepsilon}}(b) = Cb + B$. This means all the scaling processes S_i's of the retransmission flows can be bounded by the same stochastic scaling curve; this holds true for any overall scaling process that is *i.i.d.* (e.g., the BSC). For ease of exposition, we also assume homogeneity of the scaling violation probabilities $\overline{\varepsilon_i}$'s. Note further that under the BSC, C and B can be calculated using Eq. (3), and should satisfy $B \geq 0$ and $0 \leq C < 1$. The latter condition is necessary for the convergence of T, since otherwise the scaling would not act as a contractor but as an expander such that α_n would diverge to infinity. We point out that the analysis of more complex arrival, service, and scaling curves would follow a similar line of argument, and is left for future work.

2) Fixed-Point Calculation: For illustrative purposes, we demonstrate the derivation of the arrival curves of retransmission flows for two cases: a single retransmission case and the general case of N retransmissions. We focus especially on the first case, as the second one can be treated as a generalization by using the same method.

(1) N = 1 $\leftrightarrow$ one retransmission flow

We know that $\alpha^{(0)} = \alpha$, $\alpha^{(1)} = \overline{S}^{\overline{\varepsilon}}\left(\alpha^{(0)} \oslash \beta^{(0)}\right) \oslash \delta_W = \overline{S}^{\overline{\varepsilon}}\left(\alpha \oslash \left[\beta - \alpha^{(1)}\right]^+\right) \oslash \delta_W$. It is to be checked whether the mapping for $\alpha^{(1)}$ is convergent and what is the fixed point of the mapping $\alpha_\infty^{(1)}$. Put differently, if we set the initial input of $\alpha^{(1)}$ as $\alpha_1^{(1)} = \gamma_{Cr,b_1}$, where b_1 is the variable of burst needed to solve the fix-point equations, the task is to check whether there is a convergent limit b_∞ for b_1 and what is its value. Note that the rate of $\alpha^{(1)}$ is Cr because any other rate of $\alpha^{(1)}$ will be limited to the rate of α (i.e., r) after the deconvolution $\alpha \oslash [\beta - \alpha^{(1)}]^+$, and also after the invocation of the scaling curve $\overline{S}^{\overline{\varepsilon}}(b) = Cb + B$.

Next, we perform a step-by-step calculation of the formulation $\alpha^{(1)} = \overline{S}^{\overline{\varepsilon}}\left(\alpha \oslash [\beta - \alpha^{(1)}]^+\right) \oslash \delta_W$ until we achieve enough information to assess its convergence and are able to calculate the fixed-point value b_∞ of $b_1, b_2, \ldots$ This process is depicted in Figure 3 and is explained in the following steps:

1. Calculate curve $\left[\beta - \alpha_1^{(1)}\right]^+ = \beta_{R-Cr,T_1}$, where T_1 is the latency to be determined. From $CrT_1 + b_1 = R(T_1 - T)$ we have

$$T_1 = \frac{RT + b_1}{R - Cr} \ .$$

2. Calculate $\alpha \oslash \left[\beta - \alpha_1^{(1)}\right]^+$. Draw a line at point $(-T_1, b)$ with rate equal to the rate of the arrival curve α, which is r. Now calculate the burst of the above curve - b_2' as

$$b_2' = rT_1 + b \ .$$

3. At last, calculate $\alpha_2^{(1)} = \overline{S}^{\overline{\varepsilon}}\left(\alpha \oslash \left[\beta - \alpha_1^{(1)}\right]^+\right) \oslash \delta_W$ as $\alpha_2^{(1)} = \gamma_{Cr,b_2}$ with

$$b_2 = Cb_2' + B + CrW = C(rT_1 + b) + B + CrW \ .$$

Repeating step 1, 2 and 3 for $\alpha_2^{(1)}$ we obtain

$$T_2 = \frac{RT + b_2}{R - Cr} \ ,$$
$$b_3 = C(rT_2 + b) + B + CrW \ .$$

Then we repeat this calculation again to obtain $b_4 = C(rT_3 + b) + B + CrW$ and so on. That means the convergence of $\alpha^{(1)}$ depends on the sequence of T_1, T_2, T_3, ... We can write this sequence for integer $j > 1$ as follows:

$$\begin{aligned} T_1 &= \frac{RT + b_1}{R - Cr} \ , \cdots \\ T_j &= \frac{Cr}{R - Cr} T_{j-1} + \frac{RT + Cb + B + CrW}{R - Cr} \ . \end{aligned}$$

At the same time, we have

$$b_j = (R - Cr)T_j - RT \ .$$

For the deconvolution $\alpha \oslash \left[\beta - \alpha^{(1)}\right]^+$ to exist, the following condition must hold

$$\begin{aligned} \lim_{t\to\infty} \frac{\left[\beta - \alpha^{(1)}\right]^+(t)}{t} \geq \lim_{t\to\infty} \frac{\alpha(t)}{t} &\implies R - Cr \geq r \\ &\implies \frac{Cr}{R - Cr} \leq C < 1 \ . \end{aligned}$$

This is in fact a stability condition for the system. In particular, $R > r + Cr$ means that the long-term capacity of the server can satisfy the long-term needs of the original and the retransmitted flow (recall also our previous assumption that $C < 1$ to guarantee stability). Applying the stability condition to

$$T_j = \frac{Cr}{R - Cr} T_{j-1} + \frac{RT + Cb + B + CrW}{R - Cr} \ ,$$

we obtain that the sequence of T_j is convergent, i.e., there is a fixed point T_∞ with

$$T_\infty = \frac{RT + Cb + B + CrW}{R - 2Cr} \ .$$

Finally, we can calculate the arrival curve of the retransmission flow as

$$\begin{aligned} \alpha^{(1)} &= \gamma_{Cr,b_\infty} \ , \text{ where} \\ b_\infty &= (R - Cr)T_\infty - RT \ . \end{aligned}$$

(2) General N $\leftrightarrow$ N retransmission flows

Because the calculation process for $N \geq 2$ is very similar to $N = 1$ (the only difference is that now we face an equation system due to $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(N)}$), we ignore the computational details and provide the results directly. First, we assume the following condition

$$R > \left(1 + C + C^2 + ... + C^N\right) r = \frac{1 - C^{N+1}}{1 - C} r \qquad (4)$$

for the system's stability. This inequality can be used as a tool to adjust the server capacity, the input flow rate or maximum number of retransmissions. If we denote the convergent latency of $\left[\beta - \alpha^{(j)} - \alpha^{(j+1)} - ... - \alpha^{(N)}\right]^+$ with $T_{j,\infty}$, for all $1 \leq j \leq N$, then the fixed-point problem reduces to solving the following equation system

$$A \times (T_{1,\infty}, T_{2,\infty}, ..., T_{j,\infty}, ..., T_{N,\infty})^t = \phi, \text{ where}$$

$$A = \begin{pmatrix} R - 2r\sum\limits_{i=1}^{N} C^i & -r\sum\limits_{i=2}^{N} C^i & \dots & -r\sum\limits_{i=N}^{N} C^i \\ -r\sum\limits_{i=2}^{N} C^i & R - 2r\sum\limits_{i=2}^{N} C^i & \dots & -r\sum\limits_{i=N}^{N} C^i \\ \vdots & \vdots & \ddots & \vdots \\ -r\sum\limits_{i=N}^{N} C^i & -r\sum\limits_{i=N}^{N} C^i & \dots & R - 2r\sum\limits_{i=N}^{N} C^i \end{pmatrix}$$

$$\phi = \begin{pmatrix} RT + b\sum\limits_{i=1}^{N} C^i + B \cdot \sum\limits_{p=0}^{N-1}\sum\limits_{q=0}^{p} C^q + rW \cdot \sum_{i=1}^{N} iC^i \\ RT + b\sum\limits_{i=2}^{N} C^i + B \cdot \sum\limits_{p=1}^{N-1}\sum\limits_{q=0}^{p} C^q + rW \cdot \sum_{i=2}^{N} iC^i \\ \vdots \\ RT + b\sum\limits_{i=N}^{N} C^i + B \cdot \sum\limits_{p=N-1}^{N-1}\sum\limits_{q=0}^{p} C^q + rW \cdot \sum_{i=N}^{N} iC^i \end{pmatrix}.$$

We next use Cramer's Rule to derive

$$T_{1,\infty} = \frac{det(A_1)}{det(A)}, T_{2,\infty} = \frac{det(A_2)}{det(A)}, ..., T_{N,\infty} = \frac{det(A_N)}{det(A)},$$

where $A_{i,1\leq i\leq N}$ is the matrix A with the ith column of A replaced by ϕ. If all roots are positive, then a fixed point exists. In this case, the arrival curves of all N retransmission flows are given as follows

$$\begin{aligned} \alpha^{(j)} &= \gamma_{C^j r, b_{j,\infty}}, \text{ where} \\ b_{j,\infty} &= C^j r(T_{j,\infty} + T_{j-1,\infty} + ... + T_{1,\infty}) \\ &\quad + C^j b + (C^{j-1} + ... + C + 1)B + jC^j rW . \end{aligned}$$

D. Performance Bounds

In the previous section, we have shown a fixed-point approach to derive the arrival curves for each retransmission flow $\alpha^{(i)}$ and, consequently, also the service curves $\beta^{(i)}$ as seen by each of these flows. As discussed above, the arguments were given under a deterministic interpretation of arrival, service, and scaling curves. However, for the derivation of probabilistic performance bounds (e.g., the delay of a data unit through the unreliable link), we now need to take into account the stochastic nature of the unreliable link and therefore of the underlying scaling process.

Firstly, let us assume that there exists a sample-path over which the scaling functions of all the flows, original and retransmissions, do not violate their sample-path stochastic scaling curves $\overline{S}_i^{\overline{\varepsilon}}$'s. If this assumption applies, according to the calculation described in the previous section, we can firstly derive the arrival curve for the aggregate arrivals as $\sum_{i=0}^{N} \alpha^{(i)}$. And the service curve for the aggregate arrivals is β. Now we obtain the delay bound

$$\forall t : d(t) \leq h\left(\sum_{i=0}^{N} \alpha^{(i)}, \beta\right) .$$

Note that this delay bound excludes the direct delay contributions of the feedback for retransmitted packets, though it takes the feedback loops burstiness increase effect into account (see also Section V). These delay contributions can simply be added according to the number of necessary retransmissions and the maximum feedback delay, but are omitted in the following due to their rather uninteresting nature.

Secondly, we need to calculate the violation probability of the above sample-path assumption. As discussed above, each flow i is subject to a scaling process S_i, and all S_i's form a partition of the overall scaling process S. Given an *i.i.d.* overall scaling process S (e.g., the BSC), all S_i's are *i.i.d.* as well and mutually independent. A probabilitic delay bound can be computed by calculating the probability of the sample-path event that the stochastic scaling curves are not violated as follows

$$\begin{aligned} P\left(d(t) \leq h\left(\sum_{i=0}^{N} \alpha^{(i)}, \beta\right)\right) &\geq P\left(\bigwedge_{i=1}^{N} \left\{\overline{S}_i^{\overline{\varepsilon}} not\, violated\right\}\right) \\ &= \prod_{i=1}^{N} P\left(\overline{S}_i^{\overline{\varepsilon}} not\, violated\right) \\ &\geq (1 - \overline{\varepsilon})^N , \end{aligned} \qquad (5)$$

by invoking the statistical independence in the second line. Note that, if the S_i's were not *i.i.d.* , we could still use the union bound to compute the violation probability (of course, resulting in a more conservative bound).

Similar reasoning can be applied to compute the probabilistic backlog bound for all of the flows:

$$P\left(b(t) \leq v\left(\alpha^{(0)} + \alpha^{(1)} + ... + \alpha^{(i)}, \beta\right)\right) \quad \geq \quad (1 - \overline{\varepsilon})^N ,$$

with statistical independence assumptions.

V. Numerical Example

In order to illustrate the application of the model, let us go through a numerical example in this section. Before the calculation, we state some necessary assumptions.

Assumptions: *Consider the scaling process S to be a BSC with loss probability p varying from 0.1 to 0.9 (from normal state to channel collapse). The arrival curve of the input flow is $\alpha = \gamma_{r,b} = \gamma_{0.1,3}$. The service curve is $\beta = \beta_{R,T} = \beta_{1,3}$. The service curve of the feedback delay is $\delta_W = \delta_8$. The violation probability of the sample-path stochastic scaling curve is $\overline{\varepsilon} = 0.001$. The value of W will later be varied from 0 to 40, in order to illustrate the impact of the feedback delay.*

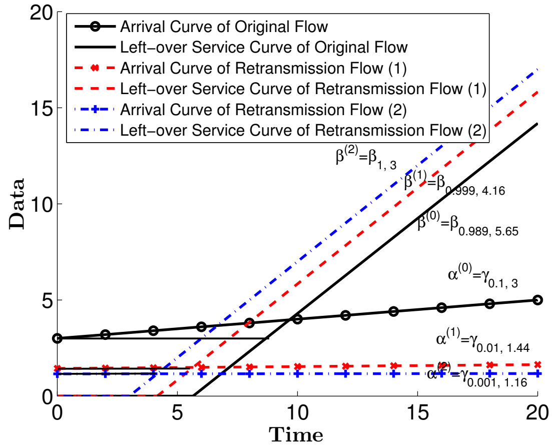

Figure 4. Arrival, service curves, and delay bounds.

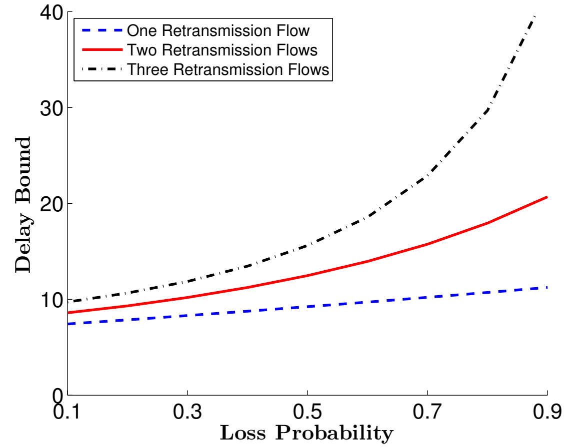

Figure 5. Delay bounds with retransmission attempts i ($i = 1,\ 2,\ 3$).

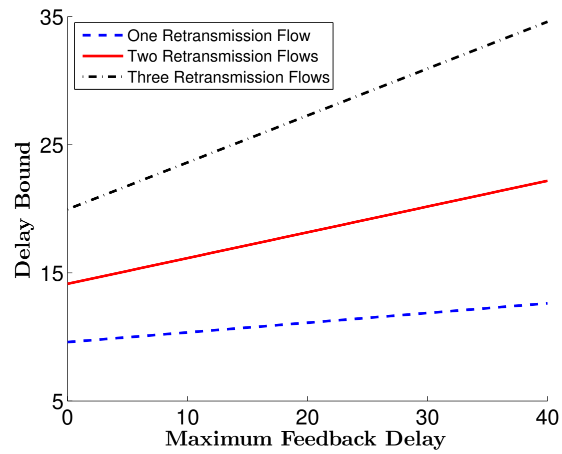

Figure 6. Delay bounds with changing maximum feedback delay (loss probability $p = 0.7$).

Target of Calculations: *We compute probabilistic delay bounds with different loss probabilities and for different number of retransmissions $N = 1,\ 2,\ 3$, as well as with varying feedback delays.*

First, in order to illustrate how to derive the arrival curves for all the retransmission flows, let us pick $N = 2$ as an example. From Subsection III-B we know that a sample-path stochastic scaling curve for a BSC with parameter p can be calculated as in Eq. (3). Let $C = p$ and $B = 1-\bar{\varepsilon}$. We can now use the results from Subsection IV-C to derive the formulae of arrival curves $\alpha^{(1)}$ and $\alpha^{(2)}$ for the two retransmission flows. As a result, we know the arrival curves $\alpha^{(1)}$, $\alpha^{(2)}$ and the left-over service curves $\beta^{(1)}$, $\beta^{(2)}$. For example, for $p = 0.1$, these are depicted in Figure 4 together with horizontal lines representing the delay bounds for each retransmission flow.

Comparing the rates of α and $\alpha^{(1)}$, we observe that the rate of $\alpha^{(1)}$ (i.e., Cr) is much smaller than the rate of α (i.e., r); this is because $\alpha^{(1)}$ is the retransmission flow caused by data loss and the probability of data loss is not too high, which means that the effect of the retransmission flow is rather weak (i.e., $C << 1$). Consequently, the retransmission flow for $\alpha^{(1)}$, which is $\alpha^{(2)}$, is even weaker. Next we observe from Figure 4 that $b > b_1 > b_2$. This is intuitive, since b as the original flow's parameter is expected to be greater than b_1 and b_2.

With the computed values of $\alpha^{(1)}$ and $\alpha^{(2)}$ we can next calculate the arrival curve for the aggregate flows as $\sum_{i=0}^{2} \alpha^{(i)} = \gamma_{0.111,5.59}$. And using Eq. (5) we can now compute the probabilistic delay bound with two retransmissions (recall that $N = 2$)

$$P\left(d(t) \le h\left(\sum_{i=0}^{2} \alpha^{(i)}, \beta\right) = 8.5902\right) \ge (1-\bar{\varepsilon})^2 = 0.9980\,.$$

Next, we show the delay bounds for $N = 1,\ 2,\ 3$ and $p = 0.1,\ 0.2,\ \cdots,\ 0.9$ in Figure 5. In the figure, for each N, we plot a curve for p from 0.1 to 0.9. The delay bounds are expectedly increasing in N and p. Clearly, the more data is lost and the higher reliability the communication requires (higher N), the higher is the delay. Most interestingly, the figure shows for $N = 3$ a steep rise for increasing loss probabilities, whereas for $N = 1, 2$ the bounds are relatively insensitive to the loss probability. Hence, we encounter an interesting phase transition phenomenon for the impact of the number of retransmissions on the performance bounds, i.e., there exists a threshold value for N above which the performance bounds blow up. As discussed in Subsection IV-D, the feedback delay is excluded from the total delay bound, such that this blow-up does not directly and trivially relate to the maximum feedback delay but is due to queueing effects only. From the stability condition Eq. (4), we can clearly derive a maximal N such that the node is still in stable state. A potential usage of this knowledge could be to dynamically adapt the maximum number of retransmissions per data unit to control the tradeoff between delay and reliability according to the current utilization and loss characteristics of a lossy link.

In Figure 6, we show the impact of the maximum feedback delay on the delay bounds for $N = 1, 2, 3$ and $W = 0, 1, \cdots, 40$. The delay bounds are increasing in the maximum feedback delay for all the three cases. Although we have excluded the direct contribution of the feedback delay, it still increases the possibility to cumulate burstiness in the retransmitted flow at the server. Clearly, this cumulated burstiness eventually increases the delay bound. The higher the maximum retransmission number, the more often a retransmitted packet may experience the feedback delay, and thus, the burstier the aggregate flow becomes. This is illustrated in the Figure 6 by the increasing slopes for $N = 1, 2, 3$.

VI. Conclusion

In this paper, we made a step forward on the way to model unreliable networks using the stochastic network calculus. Based on the stochastic extension of the data scaling element from deterministic network calculus we showed how to model and analyze an unreliable link that employs a retransmission-based loss recovery. Solving this model involved a fixed-point analysis yielding probabilistic performance bounds. In a numerical example, we illustrated how to apply the theoretical results and demonstrated the model's capabilities to provide interesting insights into the system behavior. In particular, we showed that even at small utilizations, a relatively small number of retransmission attempts already lends itself to a delay bound's blow-up. This provides incentives for protocols to dynamically adapt the maximum number of retransmissions. Moreover, our model can also reveal the quantitative impact of the feedback delay on the delay bound of the aggregate flow. An interesting and desirable future work is to investigate whether our technique to analyze single unreliable links and the concatenation principle from network calculus can be combined to analyze unreliable networks.

References

[1] F. Agharebparast and V. Leung. Modeling wireless link layer by network for efficient evaluations of multimedia QoS. In *Proceedings of IEEE ICC*, pages 1256–1260, May 2005.

[2] 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.

[3] S. Ayyorgun and R. L. Cruz. A composable service model with loss and a scheduling algorithm. In *Proceedings of IEEE INFOCOM*, March 2004.

[4] F. Baccelli, G. Cohen, G. J. Olsder, and J.-P. Quadrat. *Synchronization and Linearity: An Algebra for Discrete Event Systems*. Probability and Mathematical Statistics. 1992.

[5] P. Billingsley. *Probability and Measure (3^{rd} Edition)*. Wiley, 1995.

[6] A. Bouillard and E. Thierry. An algorithmic toolbox for network calculus. *Discrete Event Dynamic Systeme*, 18(1):3–49, March 2008.

[7] 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, September 2006.

[8] S. Chakraborty, S. Kuenzli, L. Thiele, A. Herkersdorf, and P. Sagmeister. Performance evaluation of network processor architectures: Combining simulation with analytical estimation. *Computer Networks*, 42(5):641–665, April 2003.

[9] C.-S. Chang. On deterministic traffic regulation and service guarantees: A systematic approach by filtering. *IEEE Transactions on Information Theory*, 44(3):1097–1110, May 1998.

[10] C.-S. Chang. *Performance Guarantees in Communication Networks*. Telecommunication Networks and Computer Systems. Springer-Verlag, 2000.

[11] F. Ciucu, A. Burchard, and J. Liebeherr. A network service curve approach for the stochastic analysis of networks. In *Proceedings of ACM SIGMETRICS*, pages 279–290, June 2005.

[12] F. Ciucu and J. Schmitt. Perspectives on network calculus: no free lunch, but still good value. In *Proceedings of ACM SIGCOMM*, pages 311–322, August 2012.

[13] F. Ciucu, J. Schmitt, and H. Wang. On expressing networks with flow transformation in convolution-form. In *Proceedings of IEEE INFOCOM*, pages 1979–1987, April 2011.

[14] R. L. Cruz. A calculus for network delay, Part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, January 1991.

[15] R. L. Cruz. A calculus for network delay, Part II: Network analysis. *IEEE Transactions on Information Theory*, 37(1):132–141, January 1991.

[16] R. L. Cruz. Quality of service guarantees in virtual circuit switched networks. *IEEE Journal on Selected Areas in Communications*, 13(6):1048–1056, August 1995.

[17] R. L. Cruz. Quality of service management in integrated services networks. In *Proceedings of the 1st Semi-Annual Research Review, Center for Wireless Communications*, 1996.

[18] R. L. Cruz and M. Taneja. An analysis of traffic clipping. In *Conference of Information Sciences and Systems*, March 1998.

[19] M. Fidler. An end-to-end probabilistic network calculus with moment generating functions. In *Proceedings of IEEE IWQoS*, pages 261–270, June 2006.

[20] M. Fidler. A network calculus approach to probabilistic quality of service analysis of fading channels. In *Proceedings of IEEE Globecom*, November 2006.

[21] M. Fidler and J. Schmitt. On the way to a distributed systems calculus: An end-to-end network calculus with data scaling. In *Proceedings of ACM SIGMETRICS/Performance*, pages 287–298, 2006.

[22] Y. Gao and Y. Jiang. Performance analysis of a cognitive radio network with imperfect spectrum sensing. In *IEEE INFOCOM Workshops*, pages 1–6, March 2010.

[23] Y. Jiang. A basic stochastic network calculus. In *Proceedings of ACM SIGCOMM*, pages 123–134, September 2006.

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

[25] H. Kim and J. C. Hou. Network calculus based simulation: theorems, implementation, and evaluation. In *Proceedings of IEEE INFOCOM*, March 2004.

[26] A. Koubaa, M. Alves, and E. Tovar. Modeling and worst-case dimensioning of cluster-tree wireless sensor networks. In *Proceedings of the 27th IEEE International Real-Time Systems Symposium*, pages 412–421, December 2006.

[27] R. Lübben and M. Fidler. On the delay performance of block codes for discrete memoryless channels with feedback. In *IEEE Sarnoff Symposium (SARNOFF)*, pages 1–6, May 2012.

[28] J.-Y. Le Boudec. Application of network calculus to guaranteed service networks. *IEEE Transactions on Information Theory*, 44(3):1087–1096, May 1998.

[29] J.-Y. Le Boudec and P. Thiran. *Network Calculus A Theory of Deterministic Queuing Systems for the Internet*. Number 2050 in Lecture Notes in Computer Science. Springer-Verlag, 2001.

[30] R. Luebben and M. Fidler. Non-equilibrium information envelopes and the capacity-delay-error-tradeoff of source coding. Report arXiv:1107.3087v1, arXiv, July 2011.

[31] H. Sariowan, R. L. Cruz, and G. C. Polyzos. Scheduling for quality of service guarantees via service curves. In *Proceedings of IEEE ICCCN*, pages 512–520, September 1995.

[32] J. Schmitt and U. Roedig. Sensor network calculus - a framework for worst case analysis. In *Proceedings of Distributed Computing on Sensor Systems*, pages 141–154, June 2005.

[33] J. Schmitt and F. Zdarsky. The DISCO Network Calculator - a toolbox for worst case analysis. In *Proceedings of VALUETOOLS*, November 2006.

[34] 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, February 2006.

[35] D. O. Wu and R. Negi. Effective capacity: A wireless link model for support of quality of service. *IEEE Transactions on Wireless Communications*, 2(4):630–643, July 2003.

[36] J. Xie and Y. Jiang. An analysis on error servers for stochastic network calculus. In *IEEE Conference on Local Computer Networks*, October 2008.

[37] Y. Yuan, K. Wu, W. Jia, and Y. Jiang. Performance of acyclic stochastic networks with network coding. *IEEE Transactions on Parallel and Distributed Systems*, 22(7):1238–1245, July 2011.