A Survey of Deterministic and Stochastic Service Curve Models in the Network Calculus

Markus Fidler

Abstract—In recent years service curves have proven a powerful and versatile model for performance analysis of network elements, such as schedulers, links, and traffic shapers, up to entire computer networks, like the Internet. The elegance of the concept of service curve is due to intuitive convolution formulas that determine the data departures of a system from its arrivals and its service curve. This fundamental relation constitutes the basis of the network calculus and relates it to systems theory, however, under a different, so-called min-plus algebra. As in systems theory, the particular strength of the min-plus convolution is the ability to concatenate tandem systems along a network path. This facilitates the notion of network service curve that has the expressiveness to characterize whole networks by a single transfer function.

This paper surveys the state-of-the-art of the deterministic and the recent probabilistic network calculus. It discusses the concept of service curves, its use in the network calculus, and the relation to systems theory under the min-plus algebra. Service curve models of common schedulers and different types of networks are reviewed and methods for identification of a system’s service curve representation from measurements are discussed. After recapitulating the state of knowledge on time-varying min-plus systems theory, stochastic service curve models are surveyed. These models allow utilizing the statistical multiplexing gain in a network calculus framework that features end-to-end network analysis by convolution of service curves.

Index Terms—Network performance analysis, network calculus, min-plus systems theory, service curves, scheduling.

I. INTRODUCTION

The well-founded design of future networks requires in-depth knowledge of the interaction of a variety of network mechanisms and protocols to optimize their joint performance. For almost a century queuing theory has been used very successfully to understand many aspects of communications. In the sixties it has put forward the paradigm shift from circuit to packet switching that is the basic technology of the Internet. In the nineties it was, however, observed that packet data traffic does not fulfill the memoryless property of Poisson processes that are used by this theory. In contrast, it was shown that Internet traffic exhibits significant correlations. During the last decade new theories have been developed to bridge this gap. With the advent of network architectures for quality of service the network calculus has been developed as an initially deterministic framework for analysis of worst-case backlogs and delays. Relevant results include the capacity that has to be provisioned, buffer sizes that are required to avoid packet loss, and end-to-end delays that need to be considered by time-critical applications, including telephony, video streaming, and online gaming. Recently, significant progress has been made towards a stochastic network calculus that makes use of the statistical multiplexing gain in packet switching networks, as opposed to the deterministic calculus.

The network calculus is a theory of queuing systems that emerged from the seminal works by Cruz [1], [2] on the \((\sigma, \rho)\) traffic characterization and a calculus for network delay and from the works by Parekh and Gallagher [3], [4] on the service curve characterization of Generalized Processor Sharing (GPS) schedulers. The concept of service curve has subsequently been formalized by Cruz [5], [6], Sariowan, Cruz, and Polyzos [7], Agrawal, and Rajan [8], Chang [9], Le Boudec [10], and Agrawal, Cruz, Okino, and Rajan [11] towards the general and elegant framework known as network calculus today.

The service curve model describes network elements, such as routers, schedulers, and links, using functions of time that specify the service that is offered by the element during a defined time interval. Similar to systems theory, service curves can be viewed as the impulse response of a linear system, however, under a different, so-called min-plus algebra (also known as tropical algebra). In this algebra, the data departures of a network element can be computed from convolution of the arrivals and the system’s service curve. This convolution form is significant since it establishes a general framework for analysis of entire networks. Individual systems along a network path can be easily concatenated by convolution of their service curves yielding a network service curve that specifies the end-to-end available service.

Closely related to min-plus algebra is the max-plus algebra that is detailed in the textbook by Baccelli, Cohen, Olsder, and Quadrat [12]. Connections exist also with the field of convex analysis, see e.g. the textbook by Rockafellar [13]. The deterministic network calculus is nicely covered in a tutorial by Le Boudec [14] as well as in the much more comprehensive textbook by Le Boudec and Thiran [15] that is also available online as a revised version [16]. The substantial textbook by Chang [17] covers the deterministic network calculus as well as the theory of effective bandwidths and first approaches to the stochastic network calculus. A selection of recent results on the stochastic network calculus are covered in the book by Jiang and Liu [18]. A related survey of envelope processes is provided by Mao and Panwar [19]. An overview of selected topics is also provided in the analytical textbook on networking by Kumar, Manjunath, and Kuri [20].

The most prominent applications of the network calculus are in the area of Internet quality of service, e.g. the Guaranteed
Service [21] in Integrated Services [22] networks. Likewise, the Expedited Forwarding Per-Hop Behavior [23], [24] of the Differentiated Services [25], [26] architecture has been defined using basic concepts of the network calculus. Numerous parameter-based, e.g. [27], [28], [29], [30], [31], [32], [33], as well as measurement-based admission control schemes, e.g. [34], [35], [36], [37], make use of the network calculus to achieve quality of service. Related models are used for conformance testing, e.g. [38]. For comprehensive surveys and tutorials on Internet quality of service see [39], [40], [41].

A variety of further applications of the network calculus have emerged, including wireless sensor networks [42], [43], switched Ethernet [44], systems-on-chip [45], [46], to speed-up simulations [47], [48], for measurement-based bandwidth estimation [49], [50], and even beyond computer networking in manufacturing blocking [51]. Hence, besides queuing theory the network calculus has become a valuable methodology for modeling and analysis.

To date a number of software packages exist that implement certain subsets of the overall framework of the network calculus and related functionalities. These include the CyNC toolbox [52], [53] that is based on Matlab/Simulink, the RTC toolbox [54] in Matlab, the DISCO network calculator [55], [56] in Java, the COINC toolbox [57] in C++, and the related commercial SymTA/S toolbox [58].

This paper provides a survey of deterministic and stochastic service curve models in the network calculus. It is concerned with the service that is offered by a network element or an entire network to its data arrivals. It complements the recent survey of envelope processes by Mao and Panwar [19] that considered the entire network to its data arrivals. It complements the recent survey is part of the thesis [59].

A. Work-conserving constant rate servers

We show an intuitive derivation of the notion of service curve using a work-conserving buffered link with rate $R$ as a simple yet important introductory example. Work-conserving means that the link does not idle if there are data in the system awaiting to be processed. Under the above assumptions data depart with rate $R$ whenever the system is backlogged, see Fig. 1 for illustration. If we select two time instances $t \geq 0$ that fall into the same backlogged period, that is the system does not become idle in the interval $[\tau, t]$, it holds that

$$D(t) \geq D(\tau) + R(t - \tau).$$

(1)

In (1) the amount of data that left the system in the interval $[0, t]$ denoted $D(t)$ is composed of the data that left in $[0, \tau]$ plus the data that has been forwarded by the link in $[\tau, t]$. The work-conserving link processes data in $[\tau, t]$ at full rate since it is continuously backlogged. Choosing $\tau$ to be the beginning of the last backlogged period before $t$ results in an empty system particular utility of the service curve approach when deriving performance bounds for tandem systems.

In the sequel we generally assume that all functions are non-negative, non-decreasing, and pass through the origin that is they are in

$$F_0 = \{ f : f(t) \geq f(\tau) \geq 0 \forall t \geq \tau, f(0) = 0 \}$$

Throughout this survey we make the following assumptions unless stated otherwise: We assume lossless systems which generally provide sufficient buffer space to store all incoming data. For notational simplicity we use a fluid model where data are infinitely divisible. We denote a system’s cumulative arrivals in an interval $[0, t]$ by $A(t)$. The arrivals in an interval $[\tau, t]$ follow as $A(\tau, t) = A(t) - A(\tau)$. For convenience we use $A(t)$ meaning $A(0, t)$. Similar definitions apply for the departures of a system denoted $D$. We assume that time is continuous and functions of time are left-continuous. From causality it holds that $A \geq D$ where we use shorthand notation meaning $A(t) \geq D(t)$ for all $t$. If $A(t) > D(t)$ we say that the system is backlogged at $t$. If arrivals from several flows denoted $A_i(t)$ for $i = 1, \ldots, m$ are multiplexed the aggregate has cumulative arrivals $A(t) = \sum_{i=1}^{m} A_i(t)$.

II. THE NOTION OF SERVICE CURVES

In this section we develop the notion of service curve from the simple example of a buffered link. We show the
at \( \tau \) such that we can substitute \( D(\tau) = A(\tau) \). Hence for any \( t \geq 0 \) it holds that
\[
\exists \tau \in [0,t]: D(t) \geq A(\tau) + R(t - \tau)
\]
\[\Leftrightarrow D(t) \geq \inf_{\tau \in [0,]} \{ A(\tau) + R(t - \tau) \} \tag{2}\]
In fact, it can be shown that the derived lower bound is also an upper bound such that (2) actually holds with equality, see [20] for an intuitive introduction. Closely related derivations of this result are shown in [17] using Lindley’s backlog recursion and in [20] using Reich’s backlog equation.

B. Service curve generalization

The network calculus generalizes the above observations and defines the concept of service curve \( S(t) \) to model the service that is provided by a system. The notion of service curve was introduced in [3], [4] and further generalized and formalized in [5], [7], [9], [10], [11]. A system, as shown in Fig. 2, is said to offer a service curve if it holds for all \( t \geq 0 \) that
\[
D(t) \geq \inf_{\tau \in [0,t]} \{ A(\tau) + S(t - \tau) \} =: A \otimes S(t) \tag{3}
\]
where the operator \( \otimes \) is referred to as the min-plus convolution. Service curves have proven a useful model for a variety of systems that are frequently used in computer networking. In case of the constant rate link we have \( S(t) = R t \). Beyond this example latency-rate functions \( S(t) = R[t - T]^+ \) where \( [x]^+ = \max\{0, x\} \) can be used to model a link with capacity \( R \) and propagation delay \( T \) or a rate-based scheduler such as Weighted Fair Queuing that assigns a rate \( R \) to a certain traffic class subject to a latency \( T \) [60]. Affine functions \( S(t) = (\sigma + \rho t) 1_{\{t \geq 0\}} \) where \( 1_{\{x\}} \) equals one if the argument \( x \) is true and zero otherwise correspond to a leaky-bucket traffic shaper.

Fig. 3 shows an example of arrivals \( A(t) \) at a system with latency-rate service curve \( S(t) = R[t - T]^+ \). A lower bound for the departures \( D(t) \) is constructed graphically. \( S(t) \) is shifted to the right by \( \tau \) and upwards by \( A(\tau) \). After repeating this step for all \( \tau \in [0,t] \) the infimum, i.e. taking the greatest lower bound, yields a lower bound for the departures.

Since (3) provides a lower bound for the departure process the defined service curve is also frequently referred to as a lower service curve. Similarly, upper service curves can be defined to provide an upper bound for the departures. If a system implements \( S(t) \) both as a lower and an upper service curve (3) holds with equality and \( S(t) \) is referred to as an exact service curve. Examples of systems which implement an exact service curve are constant rate links \( S(t) = R t \) and leaky-bucket traffic regulators \( S(t) = (\sigma + \rho t) 1_{\{t \geq 0\}} \). In the following the unspecified term service curve is used to mean lower service curve.

Finally, if a function \( S(t) \) fulfills an accordingly generalized version of (1)
\[
D(t) \geq D(\tau) + S(t - \tau) \tag{4}
\]
for all \( t \geq \tau \geq 0 \) that fall into a continuously backlogged period the service curve is referred to as a strict service curve [15], respectively, strong service guarantee [61]. The concept of strict service curve is used to provide service guarantees over any continuously backlogged period. Mimicking the above argument for the constant rate link, it becomes obvious that the service curve property follows from the strict service curve property but the converse is not generally true. An example are systems that delay arrivals by an amount of time \( T \). Such systems offer \( \delta_T(t) = \infty \) for \( t > T \) and \( \delta_T(t) = 0 \) otherwise as an exact service curve but not as a strict service curve.

The concepts of service curves and strict service curves have been further consolidated in [61] where adaptive service guarantees are introduced, see also [15]. A system provides an adaptive guarantee \((S(t), S'(t))\) if for all \( t \geq \tau \geq 0 \) it holds that
\[
D(t) \geq \min \left\{ D(\tau) + S'(t - \tau), \inf_{\vartheta \in [\tau,t]} \{ A(\vartheta) + S(t - \vartheta) \} \right\} \tag{5}
\]
Adaptive service guarantees combine the definition of strict service curve (4) with the definition of service curve (3) which, however, is applied to a restricted time interval only. Adaptive service guarantees can be derived immediately from the properties of strict service curves. If \( \tau \) and \( t \) fall into the same backlogged period the first expression holds due to the definition of strict service curve. Otherwise the beginning of the last backlogged period before \( t \) must be in the interval \( [\tau, t] \) that is included in the second service curve expression by taking the infimum over all \( \vartheta \in [\tau, t] \). It follows that a system that offers a strict service curve \( S(t) \) also provides an adaptive service guarantee \((S(t), S(t))\).

C. Envelopes and single system performance bounds

Service curves facilitate an easy derivation of deterministic performance bounds for backlog and delay given the arrivals...
In case of leaky-bucket arrival envelope, The arrivals at a system can be upper bounded by an envelope function. The arrivals \( A(t) \) are said to conform to a deterministic upper envelope \( E(t) \) if it holds for all \( t \geq \tau \geq 0 \) that

\[
A(\tau, t) \leq E(t - \tau)
\]

or equivalently \( A(t) \leq A \otimes E(t) \) for all \( t \geq 0 \). Considering several flows \( A^i(t) \) each with envelope \( E^i(t) \) that are multiplied it follows immediately that \( \sum_{i=1}^n E^i(t) \) is an envelope of the traffic aggregate \( \sum_{i=1}^n A^i(t) \).

Arrival envelopes can be enforced using traffic regulators, such as a leaky-bucket shaper. A traffic regulator is called greedy, if it delays data only if the envelope would be violated otherwise. It is interesting to note that lossless greedy regulators with sub-additive envelope \( E(t) \) generally have an exact service curve \( S(t) = E(t) \), see [62], [8], [14], [63], [9], [10]. Vice versa, the departures of a system that has a sub-additive exact service curve \( S(t) \) have an envelope \( E(t) = S(t) \). This relation facilitates a consistent formulation of regulators as service curve elements that provides important insights, such as "reshaping comes for free" [15]. For a recent overview on envelope processes see [19].

Given a system with service curve \( S(t) \) and upper constrained arrivals with envelope \( E(t) \) an envelope of the departure process can be derived as

\[
F(t) = \sup_{\tau \geq 0} \{ E(t + \tau) - S(\tau) \} =: E \otimes S(0)
\]

where \( \otimes \) is referred to as the min-plus de-convolution\(^1\). The backlog of a system is defined as \( B(t) = A(t) - D(t) \). It is the vertical distance between the cumulative arrival and departure functions. By insertion of the definition of service curve and of arrival envelope a worst-case bound for the maximal backlog \( B_{\text{max}} \) follows as

\[
B_{\text{max}} \leq \sup_{\tau \geq 0} \{ E(\tau) - S(\tau) \} = E \otimes S(0).
\]

In case of first-in first-out (FIFO) ordering the delay, respectively, waiting time of data arriving at time \( t \) is \( W(t) = \inf \{ w \geq 0 : A(t) - D(t + w) \leq 0 \} \). The delay is the horizontal distance between the cumulative arrival and departure functions. In the same way, the maximal delay is bounded by

\[
W_{\text{max}} \leq \inf \{ w \geq 0 : \sup_{\tau \geq 0} \{ E(\tau) - S(\tau + w) \} \leq 0 \}
\]

The three bounds for departures, backlog, and delay have been proven to be tight, that is there exist feasible sample paths of arrivals and service such that they are actually attained, see section 1.4.2 in [15] for details.

Backlog and delay have an intuitive graphical representation, where the backlog bound is the maximum vertical deviation between arrival envelope and service curve and the delay bound is the maximum horizontal deviation. An example is shown in Fig. 4 for leaky-bucket constrained arrivals at a latency-rate server.

\[
A(t) \rightarrow S(t) \rightarrow S(t) \rightarrow \cdots \rightarrow S(t) \rightarrow D(t)
\]

Fig. 5. Tandem systems can be combined into a single system by convolution of their service curves.

D. End-to-end concatenation of tandem systems

An exceptionally strong property of the network calculus is the extension of the concept of service curve from single systems to an arbitrary number of systems in series. This facilitates an immediate application of single node results, such as the performance bounds shown before, to entire networks.

The departure process of a tandem of two systems with service curves \( S_1 \) and \( S_2 \), respectively, can be computed by recursive insertion as \( D(t) \geq (A \otimes S_1) \otimes S_2(t) \). Since the min-plus convolution is an associative operation we can write \( D(t) \geq A \otimes (S_1 \otimes S_2)(t) \). By iteration it follows that a network composed of \( n \) service curves \( S_i \) with \( i = 1 \ldots n \) in series as shown in Fig. 5 is equivalent to a single system with service curve

\[
S_{\text{net}}(t) = S_1 \otimes S_2 \otimes \cdots \otimes S_n(t).
\]

where \( S_{\text{net}} \) is referred to as the network service curve. Due to commutativity of the min-plus convolution, the order of systems does not have any effect on the network service curve, e.g. given a bottleneck link, its location does not matter.

Note, that the composition result is not established for the strict service curve property, i.e. if \( S_i \) for \( i = 1 \ldots n \) are strict then \( S_{\text{net}}(t) \) from (10) fulfills the definition of service curve, however, it is not provably strict. Composition results do, however, exist for adaptive service guarantees. These are provided by [61] and can also be found in [15].

The particular advantage of (10) becomes apparent when computing end-to-end performance bounds, for example in Integrated Services networks. Here we show a simplified example using arrivals that conform to a leaky-bucket envelope \( E(t) = (\sigma + \rho t)1_{\{t \geq 0\}} \), whereas the Integrated Services model assumes peak rate constrained leaky-bucket arrivals. We consider a network that consists of \( n \) homogeneous nodes in series each with latency-rate service curves \( S_i(t) = R[t - T]^+ \) when \( \rho \leq R \) for stability. The network service curve from

---

\(^1\)Despite its name the min-plus de-convolution is not an exact inverse of the min-plus convolution.
(10) becomes $S_{\text{set}}(t) = R[t - nT]^+$ and the delay bound follows from (9) as

$$W_{\text{max}} \leq nT + \frac{\sigma}{R}.$$ 

As an alternative an additive delay bound can be computed from per-node delay bounds [2] as $W_{\text{max}} = W_{1,\text{max}} + W_{2,\text{max}} + \cdots + W_{n,\text{max}}$. In this case the arrival envelope has to be computed at each node using (7) resulting in $E_i = (\sigma + (i-1)\rho T + \rho t)1_{(t>0)}$. The additive delay bound becomes

$$W_{\text{max}} \leq nT + \frac{n(\sigma + \frac{n-1}{2}\rho T)}{R}.$$ 

Clearly, the end-to-end convolution outperforms the additive delay bound. The reason for this is that the additive bound considers the worst-case output envelope and the worst-case delay at each node which, however, are mutually exclusive. As a consequence the additive delay bound scales in $O(n^2)$ compared to $O(n)$ if end-to-end convolution is used, as observed in [64], [65], [66]. It is straightforward to construct causal systems, a greedy source and lazy service curve systems, that attain the delay bound derived from end-to-end convolution. The proof of tightness of single system performance bounds [15] can immediately be extended to network service curves, since (10) is derived as an identity.

Comparing the two bounds it becomes apparent that the burst term $\sigma$ appears $n$ times in the additive bound as opposed to once in case of the end-to-end convolution, an effect that is observed for rate proportional processor sharing in [4] see section IV-C and coined “pay bursts only once phenomenon” in [15]. Note that the delay bound assumes FIFO scheduling. For non-FIFO nodes [67] shows conditions under which the pay bursts only once phenomenon does not occur.

While (10) provides a network service curve for a simple line topology results for more complex topologies including multiplexing, scheduling, and de-multiplexing of multiple flows will be reported in Sect. IV and Sect. V.

E. Basic assumptions and some relaxations

At the beginning of Sect. II a number of assumptions are made which will be widely used throughout this survey. Some of these can be easily relaxed and are made mainly for the purpose of notational simplicity whereas others are more difficult to overcome. In this subsection we review these assumptions and refer the interested reader to works that extend the basic network calculus framework shown here.

1) Packet systems: Arrivals and departures of systems are characterized by cumulative functions $A(t)$ which include all arrivals in the interval $[0, t]$ but not the arrivals at $t$. Packet arrivals at $t$ cause a discontinuity of $A(t)$ to the right such that $A(t)$ is merely assumed to be left-continuous. This framework is preferred in the textbook [15] whereas [20] uses right-continuous functions. Generally, the difference between these models is small [15] and rather a matter of convention. In contrast [17] uses a discrete time model where the arrival function can be viewed as a counter of constant sized packets, cells, or even bits. A continuous time model can be mapped to a discrete time model by sampling. As a consequence information is lost such that the reverse mapping is not exact. For details see [15].

The basic network calculus that is presented here uses a fluid flow model where data are infinitely divisible. This model does not capture irregularities that are due to constant or variable sized packets, such as store-and-forward delays. The concept of a packetizer [68], [69] was introduced to model these effects. The approach is to decompose a system that operates on packets into a tandem of a fluid system and a packetizer which subsequently delays data until entire packets are completed to restore the packet sequence. An important result is that the combination of a fluid system with service curve $S(t)$ and a packetizer with maximum packet size $l_{\text{max}}$ has $[S(t) - l_{\text{max}}]^+$ as a service curve, for details see [17], [15], [70]. A further irregularity is caused by packets in case of non-preemptive scheduling of several flows. This issue is discussed in Sect. IV.

Closely related to the min-plus network calculus is a formulation in max-plus algebra [17], [12] where, arrival times are viewed as a function of data, e.g. $T_A(n)$ denotes the arrival time of packet number $n$. The function $T_A(n)$ is the inverse, respectively, pseudo-inverse of the corresponding arrival function $A(t)$ in min-plus algebra. Since the max-plus framework uses the notion of packets it naturally facilitates a treatment of variable sized packets without resorting to the concept of a packetizer [71], [17]. In the sequel we will restrict our exposition to min-plus systems theory and only use the max-plus approach where it is particularly useful. We note, however, that many concepts can be mirrored in the max-plus algebra, also referred to as time-domain modeling in [18].

2) Lossy systems: A further assumption that is made by the basic network calculus is that systems are lossless, that is they generally provide sufficient or a priori unlimited buffer space to store all incoming data.

Certain lossy systems can be modeled using the concept of a traffic clipper that was introduced by Cruz and Taneja [72]. Like a traffic regulator the traffic clipper enforces that its output conforms to an envelope, however, the clipper does not shape the traffic by delaying data but instead actively discards non-conforming data, i.e. it is a bufferless device. Le Boudec and Thiran [73] derive optimal representations for constrained traffic regulation problems where packets that either violate buffer constraints or delay constraints are discarded by a controller. Optimal solutions to the constrained regulation problem with buffer and delay constraints are presented by Chang and Cruz [74] and by Chang, Cruz, Le Boudec and Thiran [75], [76]. The general approach to the constrained traffic regulation problem is a decomposition of the overall system into a bufferless clipper and a buffered regulator in series. Results for the loss rate of systems with limited buffer space are provided in [72], [73]. An overview can also be found in [17], [15].

A service curve approach to model lossy systems is presented by Ayyorgun and Cruz in [77], [78], [79], where a service curve model for systems that actively discard packets that cannot be delivered within a predefined deadline is developed. The derived service curve model enjoys an end-to-end concatenation property similar to (10). The active discard policy applies, however, only to specific schedulers.
A deterministic service curve model for isolated systems with buffer constraints is derived in [80]. Given arrivals with envelope \( E(t) \) at a system with service curve \( S(t) \) the authors find that an upper bound on the loss ratio \( l \) is achieved if the available buffer space is at least \( (1-l)E \otimes S(0) \). Intuitively, the result resembles the backlog bound (8) if the arrival process were scaled down by a factor \( (1-l) \). A closely related result is reported in [81].

A network calculus for end-to-end analysis of lossy systems in series that does not require specific assumptions has, however, not been developed so far and may be difficult to obtain. Yet, a useful approximation is provided by the stochastic network calculus that will be introduced in Sect. VIII. While known models do not consider loss explicitly, the tail distribution of the backlog at a lossless system can be used to approximate loss probabilities at a system with finite but large buffer, see e.g. [17] p. 292. The intuition is that if the backlog exceeds a certain threshold \( b \) then data would be lost if the buffer size were limited to \( b \).

3) Arrival constraints: The derivation of worst-case performance bounds assumes the existence of a deterministic arrival envelope that can for example be enforced by a traffic regulator. If arrivals are random such a deterministic envelope may not exist. In many relevant cases the problem can be overcome if statistical envelopes are used that are for example subject to a certain violation probability. Such envelopes require a stochastic formulation of the network calculus that will be considered in Sect. VIII. For a recent survey on such envelopes see [19].

Further on, the basic network calculus shown so far does not consider arrival constraints that are due to feedback control. An example is the window-based flow control and congestion control algorithm implemented by TCP. Such systems have been analyzed using the network calculus by Agrawal, Cruz, Okino, and Rajan [8], [82], [11], by Chang [9], and by Le Boudec and Thiran [73]. Here, we provide only a simple example as shown in Fig. 6 to discuss the functionality. The window-based congestion control algorithm ensures that at most \( w \) units of data are under transmission at any time. Given arrivals to the controller \( A_c(t) \) and departures of the network \( D(t) \), respectively, the controller may throttle excess data such that the effective arrivals to the network become \( A(t) = \min\{A_c(t), D(t) + w\} \). If the network offers a service curve \( S(t) \) we have \( D(t) \geq A \otimes S(t) \) and by recursive insertion a service curve of the combined system consisting of controller and network can be derived as \( S(t) \) convolved with the sub-additive closure\(^2\) of \( S(t) + w \). Details can also be found in the textbooks [17], [15]. A related approach for analysis of different variants of TCP feedback control in the max-plus algebra is provided by Baccelli and Hong [83].

Finally, we mention that the network calculus is primarily concerned with systems that forward data but do not alter the amount of data as for example a video transcoder that may be located within a network. Networks that include processing are investigated by Chakraborty, Künzli, Thiele, and Gries [45], [46]. These works model data scaling using a multiplicative correction of arrival envelopes. A related approach that establishes scaling envelopes is used by Fidler and Schmitt [84]. In this paper a method is derived that maps a scaling of arrivals to an inverse scaling of service curves such that end-to-end convolved service curves can be derived from (10) even in the presence of data processing units along the network path.

III. Systems theory under the min-plus algebra

In this section we show how the concept of service curve relates to systems theory. We introduce important algebraic properties of the min-plus convolution and we discuss the role of the Legendre transform in the network calculus.

A. Time-invariant min-plus linear systems

Basic systems theory deals with time-invariant linear systems. Given pairs of input and corresponding output signals of a system \((A, D)\) time-invariance means that a time-shifted version of an input signal \( A(t-\tau) \) results in an accordingly shifted but otherwise identical output signal \( D(t-\tau) \). A system is linear, if any linear combination of input signals \( c_1A_1 + c_2A_2 \) results in a corresponding linear combination of output signals \( c_1D_1 + c_2D_2 \) where \( c_1 \) and \( c_2 \) are constants. If and only if a system is time-invariant and linear the output signal of the system is given as

\[
D(t) = \int_{-\infty}^{\infty} A(\tau) \cdot S(t-\tau) \, d\tau =: A \ast S(t) \tag{11}
\]

where \( \ast \) is the convolution operator and \( S(t) \) is the system response to the Dirac unity impulse \( \delta(t) \). The Dirac impulse is defined to be infinity at \( t = 0 \), zero otherwise, and \( \int_{-\infty}^{\infty} \delta(t) \, dt = 1 \). It is the neutral element of the convolution.

The similarity to the definition of service curve becomes apparent if (11) is rephrased in min-plus algebra. Here, the addition takes the place of the multiplication and the minimum takes the place of the addition, respectively, such that the definition of exact service curve is recovered

\[
D(t) = \inf_{\tau} \{ A(\tau) + S(t-\tau) \} = A \otimes S(t). \tag{12}
\]

The service curve \( S(t) \) is the response of the system, if the input signal is a burst impulse that is defined as\(^3\)

\[
\delta(t) = \begin{cases} 
0 & \text{for } t \leq 0 \\
\infty & \text{for } t > 0.
\end{cases} \tag{13}
\]

\(^2\)The sub-additive closure of a function is the largest sub-additive function that is point-wise smaller than the original function.

\(^3\)For brevity we reuse notation \( \delta(t) \) to mean either the Dirac impulse or the burst impulse depending on the algebra that is used.
Similar to the Dirac impulse, the burst impulse is the neutral element of the min-plus convolution, that is $\delta \otimes S(t) = S(t)$. Analogous to (11) in systems theory a queuing system satisfies the input-output relation (12), i.e. it has an exact service curve, if and only if the system is time-invariant and min-plus linear. Min-plus linearity means that any min-plus linear combination of input signals $c_1 + A_1 \land c_2 + A_2$ results in the output signal $c_1 + D_1 \land c_2 + D_2$ where $\land$ denotes the minimum operator.

Examples of systems that are min-plus linear are constant rate links and leaky-bucket traffic regulators. An important class of systems in the Internet are FIFO multiplexer which, however, are non-linear [49] and hence do not have an exact service curve. Such systems can nevertheless be modeled in the network calculus using the concepts of lower and upper service curves to bound the available service. In the same way, lower and upper service curves can also be used to bound the service offered by a time-varying system. For further illustration see the intuitive introduction in [15].

B. Basic properties of min-plus convolution

The min-plus convolution obeys a number of useful properties which are closely related to classical convolution. They are, however, subject to some subtle distinctions. Formally, the algebraic structures $(\mathbb{R} \cup \infty, \land, +)$ and $(\mathcal{F}_R, \lor, \otimes)$ are commutative dioids, but not rings since there exists no inverse element for the minimum operation. For details see sections 2.1.1 and 2.1.2 in [17] and sections 3.1.2 and 3.1.6 in [15], respectively. Here, we only name the most important properties.

All three operators $\land$, $+$, and $\otimes$ share the algebraically nice properties of commutativity and associativity. Moreover, $+$ and $\otimes$ are distributive with respect to the $\land$ operator. Particularly important is the associativity of $\otimes$ as it facilitates the concatenation of systems in series which yields a favorable linear scaling of performance bounds as already shown by application of (10) in Sect. II-D. Moreover, commutativity and associativity of $\otimes$ imply that the sequence of nodes along a network path does not impact the network service curve nor end-to-end performance bounds.

The min-plus de-convolution is dual to the min-plus convolution in the sense that

$$S(t) \geq D \otimes A(t) \iff D(t) \leq A \otimes S(t). \quad (14)$$

The de-convolution is, however, not an inverse of the convolution. It is not commutative and instead of associativity it obeys the composition rule

$$(E \otimes S_1) \otimes S_2(t) = E \otimes (S_1 \otimes S_2)(t)$$

that is iterative computation of the output envelope of a tandem system using (7) repeatedly yields an envelope that is identical to the one computed from the convolved service curve $S(t) = S_1 \otimes S_2(t)$. For further properties of the min-plus de-convolution see e.g. [15].

C. The role of the Legendre-Fenchel transform

Classical systems theory frequently resorts to the Fourier transform, which constitutes the basis of a convenient dual theory for analysis of linear systems. The Fourier transform decomposes functions of time into their spectral densities that are represented as functions of frequency, thus providing an intuitive interpretation. The Fourier transform has an inverse transform such that solutions in the frequency domain can be transferred into the time domain. This property is particularly useful since the Fourier transform takes the convolution to the much simpler multiplication, hence providing a powerful method for analysis of systems in the frequency domain.

In the min-plus systems theory convex and concave Fenchel conjugates, also referred to as Legendre transform, are known to take the place of the Fourier transform. For first applications in the network calculus see [1], [85], for an overview in the context of min-plus systems theory [12], [86], [87], [88], [89], and for an elaborative exposition on convex analysis [13]. Similar to the Fourier transform, the Legendre transform can be derived from the eigenfunctions of the convolution operation. The affine functions $A(t) = b + rt$ are eigenfunctions of the min-plus convolution $A \otimes S(t)$. By insertion it follows that $(b + rt) \otimes S(t) = b + rt - \sup \{rt - S(\tau)\}$ yielding the additive eigenvalues

$$L_S(r) := \sup_{t \geq 0} \{rt - S(t)\} \quad (15)$$

that define the Legendre transform denoted $L$. The domain that is established by the Legendre transform can be interpreted as a rate domain in the network calculus, where the rate $r$ can also be viewed as the frequency of packet arrivals. The Legendre transform of the service curve of a linear system is the maximum backlog attained in case of constant rate arrivals $A(t) = rt$, i.e. it holds that

$$B_{max}(r) = L_S(r). \quad (16)$$

If a system provides only a lower service curve the Legendre transform is an upper backlog bound, i.e. it holds that $B_{max}(r) \leq L_S(r)$. The backlog provides an intuitive interpretation of a system’s Legendre transform and features a characterization of systems using backlog bounds.

The correspondence between a system’s service curve and its Legendre transform is one-to-one if the service curve is convex. Hence, a system with convex service curve can be completely characterized by its Legendre transform. Moreover, the Legendre transform is its own inverse such that the service curve can be recovered. In general it holds, however, only that the bi-conjugate computes the convex hull, i.e.

$$L_{L_S}(r) = \text{conv}_{S}(t) \leq S(t) \quad (17)$$

where $\text{conv}_{S}(t)$ denotes the convex hull of $S(t)$ that is the largest convex function that is point-wise smaller than $S(t)$. Thus, for non-convex service curves the inverse Legendre transform returns a lower bound which consequently fulfills the definition of lower service curve.

A powerful property of the Legendre transform is that it takes the min-plus convolution to a simple addition that is

$$L_{A \otimes S}(r) = L_A(r) + L_S(r). \quad (18)$$

As an example consider the impulse function $\delta(t)$ which jumps from zero to infinity at $t = 0$. The Legendre transform becomes $L_\delta(r) = 0$ for all $r \geq 0$ such that $L_{S \otimes S}(r) = L_S(r)$.
reconfirms that the impulse function is the neutral element of min-plus convolution.

Transforming (10), i.e., computation of a network service curve by min-plus convolution, results in the much simpler additive form

\[ L_{S_{net}}(r) = L_{S_1} + L_{S_2} + \cdots + L_{S_n}. \]

Once the Legendre transforms of the individual systems are known it is straightforward to compute the transform of their series connection by summation instead of convolution. The computational cost of a Legendre transform and of a min-plus convolution are similar. For certain applications the transform can significantly reduce computational overhead. An example is service curve based routing, where a large number of partially overlapping paths have to be evaluated [87].

In the same way as the convex conjugate takes the min-plus convolution to an addition, the concave conjugate takes the min-plus de-convolution to a simple subtraction, for details see [87]. To obtain the concave conjugate replace the supremum in (15) by an infimum.

While the Legendre transform has nice algebraic properties it has to be noted that neither the time domain analysis nor the rate domain analysis are generally superior to each other. While the Legendre transform takes the min-plus convolution to a simple addition the inverse transform implies that it takes additions to min-plus convolutions. As a consequence, additions in the time domain are transformed into more complicated min-plus convolutions in the rate domain. Moreover, it has to be emphasized that the Legendre transform provides a one-to-one relation only in case of convex functions. Clearly, this property is fulfilled by simple latency-rate service curves as well as typical leftover service curves, see Sect. IV. It does, however, not hold for leaky-bucket traffic regulators. For further details on the Legendre transform see [13].

IV. SCHEDULING AND LEFTOVER SERVICE CURVES

A particular strength of the service curve model is that it comprises a variety of scheduling algorithms. Given a single system with service curve \( S(t) \), through-traffic arrivals \( A'(t) \), and cross-traffic arrivals \( A''(t) \) as shown in Fig. 7, it is generally possible to derive performance bounds for the aggregated arrivals \( A(t) = A'(t) + A''(t) \). Trivially, these bounds hold also for each of the flows individually.

The configuration becomes more challenging, if through and cross-traffic are scheduled at a system and de-multiplexed afterwards, e.g., due to routing. Subsequently, the through-traffic may encounter fresh cross-traffic at a downstream system, a scenario that is considered in Sect. V and exemplified in Fig. 8. Iterating the above approach for each system implies computing additive performance bounds. These are known to be inferior to bounds from end-to-end convolved service curves, see Sect. II-D. Instead, it is desirable to identify the amount of service that is left over by cross-traffic at each of the systems to derive service curves as seen by through-traffic only. Afterwards, these so-called leftover service curves can be concatenated to derive a network service curve that characterizes the service offered by a whole network to a single through-traffic flow. Respective results for networks will be addressed in Sect. V. The feasibility of end-to-end convolution is also the reason why service curves are usually preferred over possibly tighter schedulability conditions, see e.g. [27], [90], that apply, however, only to single systems.

This section reviews a number of idealized scheduling algorithms and their fluid-flow service curve representations. We first show how a simple leftover service curve for a general scheduling model can be derived from the strict service curve property. A variety of more sophisticated leftover scheduling models follow along the same line. Finally, the fluid-flow assumption is relaxed and we summarize generic models that capture the irregularities of packet systems.

The following service curve models are deterministic and thus make certain worst-case assumptions. If a stochastic cross-traffic model is used the leftover service will become a random process. Details on respective stochastic service curves will be given in Sect. VIII.

A. Blind multiplexing and priority scheduling

From the definition of strict service curve (4) it follows for all \( t \geq \tau \geq 0 \) where \( \tau \) is the beginning of the last backlogged period before \( t \) that

\[ D'(t) \geq A'(\tau) + S(t-\tau) - (D''(t) - A''(\tau)). \]

Here, the assumption that the system is empty at \( \tau \) is used to replace \( D(\tau) = A(\tau) \). In a second step the arrivals and departures are decomposed into through and cross-traffic

\[ A(t) = A'(t) + A''(t) \quad \text{and} \quad D(t) = D'(t) + D''(t), \]

respectively. On account of causality \( D''(t) \leq A''(t) \) can be substituted and we can bound \( A''(t) - A''(\tau) \leq E''(t-\tau) \) for all \( t \geq \tau \geq 0 \)

using the cross-traffic envelope \( E'(t) \). A service curve for the through-traffic follows as

\[ S'(t) = [S(t) - E'(t)]^+. \] (19)

The \([.]^+\) condition follows from the assumption that the system is empty at \( \tau \leq t \) such that \( D'(t) \geq D''(\tau) = A''(\tau) \). It is important to note that the leftover service curve (19) cannot be derived from the more general service curve property. In particular the \([.]^+\) condition is derived under the assumption that the system is empty using the definition of strict service curve. A counterexample which proves that (19) cannot be derived from the general service curve property is shown in [15].

The model is referred to as blind multiplexing in [15] since it does not make assumptions about the order in which through and cross-traffic are served. It considers the worst-case and hence it is pessimistic for most scheduling disciplines.
The model is, however, typically employed to derive leftover service curves at a Static Priority (SP) scheduler, for example [15], [91], [92]. Given m traffic classes, respectively, flows with decreasing priorities 1 . . . m the leftover service curve seen by class i follows along the same line as (19). Subtracting the envelopes of arrivals at higher priority classes yields

\[ S^i(t) = \left[ S(t) - \sum_{j=1}^{i-1} E^j(t) \right]^+ . \]

The blind multiplexing model (19) is frequently used to show the worst-case burstiness increase of through-traffic that is due to interaction with cross-traffic at a multiplexer. Given two leaky-bucket constrained flows with parameters \((\sigma^t, \rho^t)\) and \((\sigma^c, \rho^c)\) at a latency-rate server with parameters \((R, T)\) where \(\rho^t + \rho^c \leq R\). The output burstiness of the through-traffic follows as an immediate result of basic network calculus as \(\sigma^t + \rho^c T + \rho^t (\sigma^t + \rho^c T)/(R - \rho^c)\) [15]. The result is refined in [93] assuming a constant rate server, i.e. for \(T = 0\). The server simultaneously acts as a traffic shaper with maximum rate \(R\) such that the departures generally have an additional upper envelope \(Rt\). Nevertheless, departure envelopes grow quickly and if applied for an additive analysis they result in large and usually loose end-to-end worst-case performance bounds.

**B. First-in first-out multiplexing**

The service curve under blind multiplexing can be improved if the order of scheduling through-traffic and cross-traffic is considered. For the case of FIFO multiplexing a family of leftover service curves with parameter \(\eta \geq 0\) is derived for the through-traffic as

\[ S^i(t, \eta) = [S(t) - E^i(t - \eta)]^+ 1_{\{t > \eta\}} \]  

(20)

where \(S(t)\) is the (in this case not necessarily strict) service curve offered to the aggregate of both flows and \(E^i(t)\) is an envelope of the cross-traffic arrivals. The result was first reported by Cruz [6]. A detailed proof can be found in [15]. For \(\eta = 0\) (20) matches the service curve under blind multiplexing (19).

It is important to note that while (20) provides a service curve for any \(\eta \geq 0\) the maximum of several such service curves, e.g. \(\sup_{\eta \geq 0\{S^i(t, \eta)\}\}}\), does generally not fulfill the definition of service curve [15]. It is, however, possible to derive families of upper bounds for the backlog, delay, and departures using different \(\eta \geq 0\). Since any of these is a valid upper bound the minimum of such bounds is also a valid upper bound.

An example is the departure envelope \(F^i(t) = \inf_{\eta \geq 0\{E^i(0) \cup S^i(t, \eta)\}\}}\). A solution to this min-max optimization problem is provided in [15] for leaky-bucket constrained through-traffic \(E^i(t) = \sigma^t + \rho^t t\) at a latency-rate server \(S(t) = R[t - T]^+\). Furthermore, if the cross-traffic is leaky-bucket constrained with envelope \(E^c(t) = \sigma^c + \rho^c t\) where \(\rho^c \leq R\) an optimal leftover service curve according to (20) is attained at \(\eta = T + \sigma^c / R\) [15]. The resulting service curve is of the latency-rate type \(S^i(t) = (R - \rho^c) [T - T - \sigma^c / R]^+\) and the output burstiness of the through flow follows as \(\sigma^t + \rho^c T + \rho^t \sigma^c / R\). More general solutions have been derived in [94], [95] in particular for dual leaky-bucket constrained through-traffic.

While the solution for the departure envelope implies a solution for the backlog bound, the minimal delay bound may be attained using a different parametrization of \(\eta\). A tight delay bound for leaky-bucket constrained through- and cross-traffic and latency-rate servers is derived in [96]. As opposed to the minimal output envelope, the parameter \(\eta\) that achieves the minimal delay bound depends on the parameters of the through-traffic. An early discussion of related examples is also provided in [97].

**C. Generalized processor sharing**

An important class of schedulers seeks to achieve a weighted fair allocation of resources to a number of flows \(i = 1 . . . m\) according to weights \(\phi_i\). The aim is to implement or rather approximate the Generalized Processor Sharing (GPS) policy by Parekh and Gallager [3]. Given an ideal GPS scheduler and a flow \(i\) that is continuously backlogged in the interval \([\tau, t]\) its departures \(D^i\) satisfy the defining weighting

\[ \phi^i D^i(\tau, t) \geq \phi^i D^i(\tau, t) \]

(21)

for any other flow \(j\) with departures \(D^j\). Summing (21) for all \(j = 1 . . . m\) it can be shown that a GPS scheduler at a work-conserving link with capacity \(C\) implements a lower service curve

\[ S^i(t) = \frac{\phi^i}{\sum_{j=1}^m \phi^j C} t \]

(22)

for flow \(i\). The service curve is derived using the fact that \(\sum_{j=1}^m D^j(\tau, t) = C \cdot (t - \tau)\) and a busy period argument similar to (2), see for example [17], [20].

Compared to the SP scheduler, a properly configured GPS scheduler does not increase the burstiness of leaky-bucket constrained flows. If the arrivals of class \(i\) conform to a leaky-bucket envelope with parameters \((\sigma^t, \rho^t)\) where \(\rho^t \leq C \phi^i / \sum_{j=1}^m \phi^j\) then the departures of this flow conform to a leaky-bucket envelope with the same parameters \((\sigma^t, \rho^t)\) [17].

Such an allocation can be achieved using rate proportional processor sharing [4] where \(\phi^i / \phi^j = \rho^j / \rho^t\) for all flows \(i, j\). Generally, the maximum output burstiness \(\sigma^i\) is identical to the maximum backlog of flow \(i\) [3].

The service curve (22) is derived under the conservative assumption that all flows are continuously backlogged such that they consume their allocated share of the service entirely. If the traffic of one or more classes does not fully utilize the allocated resources, the leftover service will be redistributed to backlogged classes according to their weights to satisfy (21). Thus, if flows are upper constrained and conform to arrival envelopes \(E^i(t)\) a better leftover service curve can be derived. The difficulty compared to the derivation of the leftover service curve under blind multiplexing (19) is that there is no notion of a joint busy period but the busy periods at each of the classes individually need to be considered to determine the resource allocation.

Define the set of flows \(M = \{1, . . . , m\}\) and let \(L\) be a nonempty subset of \(M\). Denote \(F^j(t)\) the departure envelope of flow \(j\). Then

\[ S^j(t) = \max_{L \subseteq M : L \neq \emptyset} \left\{ \mathcal{J} \left( \frac{\phi^j}{C} \left( t - \sum_{j \notin L} F^j(t) \right) \right) \right\} \]

(23)
is a leftover service curve for flow $i$. The service curve (23) is a slightly generalized version of the derivation reported by Chang [17]. It can be derived by summing (21) for all $j \in \mathbb{L}$, using that $\sum_{j=1}^{m} D_j^j(t,\tau) = C \cdot (t - \tau)$ and $D_j^j(t,\tau) \leq F_j(t,\tau)$. Repeating these steps for any nonempty subset $\mathbb{L} \subseteq \mathbb{M}$ yields the service curve, see [17].

The departure envelopes $F_j^j(t)$ can be easily derived from (7) using respective arrival envelopes $E_j^j(t)$ and the service curve of class $j$ according to (22). The service curve (23) can be significantly simplified using the previously mentioned results on rate proportional processor sharing. In this case $E_j^j(t) = \sigma_j + p_j t = F_j^j(t)$ can be substituted in (23) yielding the universal service curve result in [3]. For a concise derivation see also [17].

Leftover service curves for GPS are also derived in [98], [99], [100], [92] as a basis for the stochastic network calculus, see Sect. VIII. Here, we show the result from [100], [92] that is derived for concave cross-traffic envelopes

$$S_i^j(t) = \sum_{j=1}^{m} \phi_j^j \left( C t + \sum_{j \neq i} \left[ \phi_j^j \phi_k^j \phi_j^k C t - E_j^j(t) \right]^+ \right)$$

where we assumed that cross-traffic arrivals conform to deterministic envelopes $E_j^j(t)$. For a stochastic formulation using statistical cross-traffic envelopes see [98], [99], [100], [92].

The GPS policy assumes a fluid flow model where packets are transmitted virtually in parallel. Packetized GPS (PGPS) systems [3] approximate GPS in the presence of packet data traffic. To this end, PGPS emulates a GPS system seeking to schedule packets in the order of their finishing times under GPS. A major difficulty is that GPS finishing times cannot be computed at packet arrival instants. This problem is solved by Demers, Keshav, and Shenker in their Weighted Fair Queueing algorithm using a virtual time process [101].

The remaining effects that are due to the packet granularity, e.g. a packet that has to be scheduled next to achieve GPS order may not have arrived at the PGPS system yet, account for a deviation between WFQ and GPS of at most $l_{\text{max}}/C$ units of time. A number of scheduling algorithms that emulate GPS more coarsely thereby requiring less computational complexity have been proposed. For an overview see the survey by Zhang [102], the works by Goyal, Lam and Vin [103] (see also the comments in [104]), Stiliadis and Varma [60], and Jiang [105], as well as the details on packet systems in Sect. IV-E.

D. Earliest deadline first scheduling

The earliest deadline first (EDF) scheduling algorithm assumes that each data packet is assigned a deadline. At each scheduling instant an EDF scheduler dynamically selects the packet with the smallest deadline for transmission in a work-conserving manner. Schedulability conditions for EDF are derived by Zheng and Shin [106], Liebeherr, Wrege, and Ferrari [27], and Georgiadis, Guérin, and Parekh [107]. These works show that EDF systems enjoy optimality in the sense that if there exists a valid scheduling order such that all deadlines can be met then no deadline will be violated under EDF ordering.

Stochastic leftover service curves for EDF scheduling are derived in [98], [99], [100], [92]. Consider a number of flows $i = 1 \ldots m$ with arrivals $A_i^j(t)$ and envelopes $E_i^j(t)$. Each flow is assigned a target maximum delay $d_i^j$, such that packets of flow $i$ that arrive at $t$ have deadline $t + d_i^j$. A leftover service curve for flow $i$ is

$$S_i^j(t) = \left[ C t - \sum_{j \neq i} E_j^j(t - [d_j^j - d_i^j]^+) \right]^+.$$  

Here, we only phrase the deterministic case, i.e. the cross-traffic flows are assumed to conform to deterministic envelopes $E_i^j(t)$. See [98], [99], [100], [92] for respective stochastic results. The intuition behind (25) is that arrivals of flows $j$ that have a smaller target delay $d_j^j \leq d_i^j$ are generally served before flow $i$. In contrast, if $d_j^j > d_i^j$ arrivals from flow $j$ that arrive after $t - (d_j^j - d_i^j)$ will not be served by $t$ and will be overtaken by flow $i$ arrivals [92].

Selecting appropriate packet deadlines $A_i^j(t)$ of a flow with index $i$ at an EDF system. Assume the deadlines within a single flow are increasing and denote $N_i^j(t)$ the amount of data from flow $i$ with deadline smaller or equal $t$. The system offers a lower service curve $S_i^j(t)$ to flow $i$ if all departures $D_i^j(t)$ meet the deadlines given by $N_i^j(t) = A_i^j \otimes S_i^j(t)$, i.e. $D_i^j(t) \geq N_i^j(t)$.

A major result on the schedulability of flows at a work conserving link with capacity $C$ and SCED scheduling is that given a set of $m$ flows any set of target service curves $S_i^j \in F_0$ for $i = 1 \ldots m$ that fulfill

$$\sum_{i=1}^{m} S_i^j(t) \leq C t$$

for all $t \geq 0$ are feasible [108]. As long as this condition holds all deadlines $N_i^j(t) = A_i^j \otimes S_i^j(t)$ for all $t \geq 0$ and all $i = 1 \ldots m$ will be satisfied by the SCED scheduler. If in addition arrival envelopes $E_i^j(t)$ for $i = 1 \ldots m$ are known a sufficient condition is [108]

$$\sum_{i=1}^{m} E_i^j \otimes S_i^j(t) \leq C t.$$  

Concluding SCED inherits the optimal schedulability result from EDF and provides a very flexible way of assigning deadlines, however, at an increased computational complexity. A demonstration of the main results on EDF and SCED scheduling is also provided in the textbooks [17], [15] and an extension that considers variable length packets and additional switching delays is derived in [109].

E. Service curve models for packet schedulers

Packet scheduling algorithms introduce irregularities compared to the fluid-flow models presented above. An accurate fluid-flow emulation can only be achieved within certain fundamental limits and a close realization, e.g. of packet-by-packet GPS, can be computationally heavy. A number of practical packet scheduling implementations with differing
precision exist today. To this end, parameterized models have been developed to specify the service provided by different types of packet schedulers.

Goyal, Lam, and Vin developed the notion of guaranteed rate scheduling [103], [110] that is based on the work by Xie and Lam [111] and serves as a model for a variety of scheduling algorithms, such as GPS emulations. Based on a rate guarantee $R$ the target or virtual departure time $V_D(n)$ of packet number $n$ is recursively defined as

$$V_D(n) = \max\{T_A(n), V_D(n-1)\} + \frac{l(n)}{R} \quad (28)$$

where $T_A(n)$ denotes the arrival time of packet $n$ and $l(n)$ the packet length. By definition $V_D(0) = 0$ is assumed. The target departure times $V_D(n)$ are referred to as Guaranteed Rate Clock (GRC) values. The actual packet departure times $T_D(n)$ of a guaranteed rate scheduler may be late compared to (28) at most by a defined error term $T_E$, i.e.

$$T_D(n) \leq V_D(n) + T_E \quad (29)$$

where the error term depends on the scheduling algorithm, see [103], [110]. It can be viewed as describing the deviation from an ideal GPS system.

Similarly, Stiliadis and Varma [60] identify the parameters of the latency-rate service curve model for a variety of scheduling algorithms that emulate GPS. The two models, guaranteed rate scheduling that is phrased in max-plus algebra and latency-rate service curves in the min-plus network calculus, are closely related. Le Boudec [10] shows that guaranteed rate schedulers with rate $R$ and error term $T_E$ offer a latency-rate service curve $S(t) = R[t - l_{\text{max}}/R - T_E]^+$. The relation is further elaborated by Jiang [105] as well as Sun and Shin [112] who prove that guaranteed rate schedulers offer a strict service curve of the latency-rate type and vice versa.

Bennett et al. [113] extend the concept of guaranteed rate scheduling and introduce the notion of Packet Scale Rate Guarantee (PSRG). Using the same notation as above packet departure times $T_D$ ($T_D(0) = 0$ by definition) are allowed to deviate at most by some error term $T_E$ according to (29) from target departure times that are defined as

$$V_D(n) = \max\{T_A(n), \min\{T_D(n-1), V_D(n-1)\}\} + \frac{l(n)}{R},$$

It can be easily seen that the definition of PSRG is stricter than the GRC model (28), i.e. scheduling algorithms that offer PSRG ($R, T_E$) can also be modeled as guaranteed rate schedulers with parameters ($R, T_E$), whereas the converse is not necessarily true. Benett et al. [113] show that the definition of PSRG is satisfied by numerous scheduling algorithms, such as priority scheduling and GPS emulations. Jiang [114] provides details on the PSRG of hierarchical schedulers.

The notion of PSRG can be translated to adaptive service guarantees (5) with latency-rate shape, for details on the exact relation of the two concepts see [113]. Based on known results for adaptive service guarantees [113] proves a concatenation theorem for PSRG systems. A direct proof of this property is provided by Jiang [114]. A number of further important properties have been derived for PSRG systems, including a relation between delay and backlog [113], and a method to derive the PSRG of a system or even a whole network from a known delay bound [115]. While [113] mainly considers FIFO systems the concept of PSRG is extended to non-FIFO systems by Le Boudec and Charny in [116]. The concept of PSRG has been used to re-define the Expedited Forwarding Per-Hop Behavior [23], [24] in Differentiated Services networks.

V. NETWORK TOPOLOGIES

The concatenation of tandem systems by min-plus convolution of the systems’ service curves (10) is considered one of the strongest features of the network calculus. It facilitates an immediate application of single node results to networks of nodes. As a prerequisite the service curves that are offered to a flow of interest at each of the nodes along its network path have to be known.

Scheduling disciplines such as GPS or EDF can provide service guarantees to individual flows without requiring knowledge of cross-traffic arrivals using only predefined parameters as in (22) or (26), respectively. In this way, the service that is offered to a through flow by each of the systems along its path can be decoupled from existing cross-traffic, yielding the line topology according to Fig. 5. This property is one of the foundations of the Internet Integrated Services architecture [22] to provide Guaranteed Quality of Service [21]. It has been used to define configuration rules, e.g. in [4], and to derive end-to-end delay bounds, e.g. [117], [103], [118]. Similar results are achieved if flows are multiplexed using fair aggregation [119].

If leftover service curves are used, such as in case of priority scheduling, blind multiplexing (19) or FIFO multiplexing (20), but also in case of the refined GPS and EDF models (23), (24), (25), and (27), respectively, the derivation of a network service curve can be much more involved, since arrival envelopes of cross-traffic have to be known in advance. Usually, it is assumed that arrival envelopes are known at the network ingress. The envelopes of cross-traffic at multiplexing points in the network core may, however, deviate significantly from the ingress and are not known a priori (see for example the discussion of burstiness increase in Sect. IV-A). As a consequence, the network service curve of a through flow depends on the history of the cross-traffic flows whose properties may be significantly altered along their routes due to multiplexing and de-multiplexing. Solutions to these challenges can provide the much needed methods for deriving performance guarantees in Differentiated Services networks [25] such as for the Expedited Forwarding Per-Hop Behavior [23], [24].

In this section we survey methods of deriving network service curves and end-to-end performance measures. Starting from line topologies with single-hop persistent cross-traffic the class of topologies will be expanded successively to feed-forward and non feed-forward networks thus making derivations more difficult. We assume that arrival envelopes of all flows that enter a network are known a priori at the network ingress only, e.g. specified by the parameters of leaky-bucket traffic regulators. Moreover, we require that a service curve characterization of each of the network’s links is known, e.g. the rate and the propagation delay of a link.
specify the parameters of a latency-rate service curve. To this end, bidirectional links are split up into two unidirectional links beforehand. In the sequel lower indices refer to links, respectively, systems and upper indices to flows, e.g. $A_i^j(t)$ are the arrivals of flow $i$ at system $j$ and $S_j^j(t)$ is the leftover service curve offered by system $j$ to flow $i$.

### A. Single-hop persistent cross-traffic

Fig. 5 shows a single flow that traverses a simple line topology of $n$ tandem systems labeled $i = 1\ldots n$ with service curves $S_j^i(t)$. The network as seen by the through flow can be efficiently characterized by a network service curve $S_{net}(t) = S_1^2 \otimes S_2^3 \otimes \cdots \otimes S_n^1(t)$. In the presence of cross-traffic, however, leftover service curves that are offered to the through flow have to be determined beforehand.

The leftover service curve models for priority scheduling, blind multiplexing (19), FIFO multiplexing (20), GPS (23), (24), and EDF (25), (27) depend on cross-traffic envelopes. Given these envelopes are known, topologies with single-hop persistent cross-traffic, as shown in Fig. 8, can be easily transformed into the simple line topology from Fig. 5. To this end, leftover service curves are derived for each system, e.g. from (19) or (20). Afterwards, the leftover service curves are convolved to derive a network service curve for the through flow. Note that the derivation of leftover service curves for the time being assumes that cross-traffic persists only one hop.

Compared to blind multiplexing the parameterized FIFO multiplexing model can be further optimized. The optimal choice of the parameter $\eta$ in (20) depends on the actual shape of the arrival envelopes and the service curve and may require an additional minimization step once performance bounds are derived. An example is the derivation of the minimal output envelope of a FIFO multiplexer in Sect. IV-B [15] that provides the optimal parameter $\eta$. The resulting leftover service curve can be easily convolved with leftover service curves of tandem systems under single-hop persistent cross-traffic.

The scenario is further investigated in [96] where Lenzini, Mingozzi, and Stea show that the FIFO service curve parametrization that achieves the minimal output envelope does not imply a service curve that leads to the best possible delay bound. Lenzini et al. characterize each of the systems in Fig. 8 using a family of service curves according to (20). After end-to-end convolution a multi-parameter set of leftover service curves $S_{net}^i(t, \eta_1, \eta_2, \ldots, \eta_n) = S_1^j(t, \eta_1) \otimes S_2^j(t, \eta_2) \otimes \cdots \otimes S_n^j(t, \eta_n)$ is derived for the through flow $A^1$. Since $S_i^1$ is a valid service curve for any $\eta_i \geq 0$ a delay bound can be derived for any parameter set. In a final minimization step the smallest delay bound is found. Interestingly, the minimal delay bound derived in [96] does not generally exhibit the well-known pay bursts only once phenomenon, see Sect. II-D.

### B. Feed-forward networks

Beyond tandem systems an iterative computation of network service curves from leftover service curves is possible if the network topology is acyclic, i.e. a feed-forward network. While in many networks flows can form cycles there exist important topologies, such as spanning trees or sink trees that are applied e.g. in multi-point-to-point Label Switched Paths [120] that are generally acyclic. Moreover, feed-forward networks are provably stable for all utilizations up to one [121], [122]. On this account, we will define the important feed-forward property and show an inductive method for derivation of network service curves for this class of networks that follows immediately from [2]. In following subsections we will report improved results for certain topologies and show an effect coined “pay multiplexing only once” that resembles the celebrated pay bursts only once phenomenon, see Sect. II-D, with regard to cross-traffic.

A network is feed-forward, if the links can be uniquely labeled in such a way that $i < j$ if data flows from link $i$ to link $j$ [123]. Fig. 9 shows two example networks with systems $S_1$ and $S_2$ and flows $A^1$ and $A^2$. Clearly, the two systems in Fig. 9(a) can be labeled with increasing numbers whenever data flows from one system to the other system, i.e. the feed-forward property is fulfilled. In contrast no such labeling exists for the systems in Fig. 9(b) such that the topology is non feed-forward. The important feature of feed-forward networks is that flows cannot form cycles as opposed to the cycle that is created by the two flows in the non feed-forward network displayed in Fig. 9(b). It is the absence of cycles that facilitates an iterative solution using leftover service curves.

Considering the common case of network topologies with bidirectional links it is generally possible to modify the routing in such a way that flows cannot create any cycles while maintaining reachability. To this end, a couple of algorithms have been proposed that break potential cycles. The simplest of these approaches is to construct a spanning tree for routing data flows. Using any of the links that do not belong to the spanning tree is prohibited. Clearly, this approach ensures the feed-forward property, however, at the cost of a potentially large number of unused links. A less drastic approach is to prohibit the use of certain turns instead of links, where a turn $(i, j, k)$ is the concatenation of two links $(i, j)$ and $(j, k)$ that connect nodes $i$, $j$, and $k$. The up-down routing approach developed by Schroeder et al. [124] as well as the turn-prohibition algorithm by Starobinski, Karpovsky, and Zakrevski [125] use this approach to break all cycles, where
the latter is proven to prohibit at most one third of all turns. Fidler and Einhoff show that Dijkstra’s shortest path algorithm may fail in networks with prohibited turns, a problem that is solved using a network transformation called Turnnet [126].

1) **Decomposition of cross-traffic:** Given a feed-forward network it is generally possible to decompose cross-traffic along the path of a through flow to obtain the model with single-hop persistent cross-traffic. To this end, cross-traffic flows that traverse two (or more) consecutive systems are virtually split into two flows, where the departure envelope from the first system is used as arrival envelope to the second system. Fig. 10 shows an example where cross-traffic flows with envelopes $E^1_1$ and $E^2_2$ each traverse two systems, see Fig. 10(a), before being decomposed into single-hop persistent flows, see Fig. 10(b). The respective departure envelopes $F^1_1$ and $F^2_2$ are the arrival envelopes $E^2_2$ and $E^3_3$, respectively, at downstream systems.

The following procedure generalizes the analysis of feed-forward networks using the network calculus [30], [97]. Systems are assumed to be labeled in feed-forward order. The procedure is executed in increasing order of the systems’ labels starting with the system that has the smallest label. Note that at the first system, i.e., the system with the smallest label, all arrival envelopes are known since the flows cannot have passed any other system before. For each flow, referred to as the tagged flow in the sequel, that enters the system currently under investigation the following steps are executed:

1) Sum up the envelopes of all other flows at the system
2) Compute the leftover service curve for the tagged flow
3) Compute the departure envelope of the tagged flow

Once these steps have been executed for each flow at a system all of the system’s per-flow departure envelopes have been computed. In a feed-forward network this implies that all arrival envelopes at the system with the next higher label are known and the procedure can continue until all systems have been visited.

Fig. 10 exemplifies the procedure. The network consists of three links, respectively, systems with service curves $S_1$, $S_2$, and $S_3$ that are traversed by two flows that are characterized by envelopes $E^1_1$ and $E^2_2$ at the ingress of the network. The scenario and the initially available information are shown in Fig. 10(a). Starting with system 1 the departure envelope of flow 1 denoted $F^1_1$ can be immediately derived, where $F^1_1$ equals the arrival envelope of flow 1 at system 2 denoted $E^2_2$. With this information the leftover service curve for flow 2 at system 2 and hence the departure envelope $F^2_2$ can be derived. Similarly the leftover service curve for flow 1 at system 3 and the departure envelope $F^2_2$ follow. Finally, the departure envelope $F^3_3$ is the arrival envelope of flow 2 at system 3 denoted $E^3_3$ from which the departure envelope $F^3_3$ follows readily. Fig. 10(b) shows the derived departure envelopes.

As a result the procedure provides per-flow envelopes at each of the systems of a feed-forward network. This yields a model with single-hop persistent traffic, as in Fig. 8, for which efficient techniques for derivation of performance bounds are known. Note, however, that the derived envelopes are not generally tight and may result in pessimistic bounds that could be further improved.

2) **Pay multiplexing only once:** In a feed-forward network it is generally possible to decompose cross-traffic flows resulting in a network model with single-hop persistent cross-traffic. It turned out, however, that performance bounds derived from this model are not tight and may be further improved, for a quantitative comparison see e.g. [30].

The initial finding for FIFO systems by Fidler [127] is that it is possible to derive network service curves where the multiplexing of cross-traffic flows is accounted for only once regardless of the number of systems that are shared by the through and cross-traffic flows. Compared to the cross-traffic decomposition approach the resulting effect can be viewed as an extension of the well-known pay bursts only once phenomenon, see Sect. II-D, for cross-traffic flows. The approach is also followed by Lenzini et al. [128], [129], Kim and Kim [130], Koubaa et al. [43] for tree topologies, Schmitt et al. in [55], [131] where the effect has been coined “pay multiplexing only once”, and Bouillard et al. [132] for optimal routing. The early derivation of a network service curve in [127] is confuted by [128] and revised in [30].

Fig. 11 shows a simple example topology consisting of two systems and two flows where the pay multiplexing only once principle applies. Let flow 1 be the tagged flow for which a network service curve has to be determined and assume FIFO multiplexing. Following the aforementioned iterative procedure the leftover service curve $S^1_1$ and the departure envelope $F^1_1 = E^1_1 \otimes S^1_1 \geq E^1_1$ of flow 2 at system 1 is derived yielding the arrival envelope $E^3_3 = F^3_3$ of flow 2 at system 2. A network service curve for flow 1 follows from concatenation of the leftover service curves of each of the two FIFO system applying first (20) to each system and (10) afterwards resulting in

$$S^1_{net}(t, \eta_1, \eta_2) = [S_1(t) - E^3_3(t - \eta_1)]^{+1}_{t \geq \eta_1} \otimes [S_2(t) - E^3_3(t - \eta_2)]^{+1}_{t \geq \eta_2}.$$ 

The alternative, as proposed in [127], is to derive first an end-to-end service curve for the aggregate of the two flows from
(10) and compute a leftover service curve for flow 1 using (20) afterwards such that 
\[ S_{\text{net}}^1(t, \eta) = [(S_1 \otimes S_2)(t) - E_1^2(t - \eta)]^+ 1(t \geq \eta). \]

The latter form has a clear advantage since effects that are due to FIFO multiplexing and in particular due to the burstiness of the cross-traffic flow, i.e. the flow with index 2, are "paid" for only once as opposed to twice above. Generalizing the concept, the service curves of tandem systems that are traversed by the same cross-traffic flow have first to be concatenated by min-plus convolution before the leftover service curve is derived.

It is straightforward to extend this approach to topologies with nested cross-traffic flows [30]. A topology is referred to as nested if any two cross-traffic flows do either not intersect or otherwise their paths can be nested into each other. Formally, consider a tagged flow that traverses a number of indexed systems in the order of increasing indices. Cross-traffic flows are marked with tuples \((i, j)\) denoting the system where the cross-traffic flow joins, respectively, leaves the path of the tagged flow. The topology is classified as nested, if there exists no pair of cross-traffic flows \((i, j)\) and \((k, l)\) such that \(i < k \leq j < l\) [129]. Clearly, the topology in Fig. 11 is nested whereas the topology in Fig. 10(a) is not, if we view both flows as cross-traffic flows.

Lenzini et al. [129] show that nested topologies can be conveniently characterized by trees that reproduce the respective nesting structure. Fig. 12(a) shows a nested topology consisting of five systems \(S_1\) through \(S_5\) and six cross-traffic flows that are labeled with tuples \((i, j)\) indicating the system where they join, respectively, leave the path. Fig. 12(b) shows the corresponding nesting tree (upside down), e.g. flow \((2, 2)\) is nested into the path of flow \((1, 2)\) which is nested into the path of flow \((1, 5)\) and so on. The nesting tree in Fig. 12(b) immediately reveals the order of operations suggested by the pay multiplexing only once model to derive the network service curve of the through flow denoted \(S_{\text{net}}^1\):

- the circled times symbol refers to min-plus convolution of the connected service curves to compute the service curve of the respective tandem according to (10),
- an oval labeled \(-((i, j))\) refers to computation of the FIFO leftover service curve by subtracting the amount of service that may at most be consumed by flow \((i, j)\) as in (20).

Using this method each time a FIFO leftover service curve is derived from (20) a new free parameter \(\eta\) is introduced. Fidler and Sander [30] use latency-rate service curves and leaky-bucket envelopes and set the parameters according to the values that have been proven to be optimal with respect to the output envelope in [15], see also Sect. IV-B. Instead, Lenzini et al. derive families of (pseudo-affine) multi-parameter service curves and minimize the expression of the delay to derive the least upper delay bound. For the special case of sink-tree networks, that are networks where flows are multiplexed but never de-multiplexed also known as multi-point-to-point networks, Lenzini et al. [128] derive a closed-form solution that provides tight delay bounds. Additive (and thus non-tight) delay bounds for similar scenarios have previously been derived in [29]. A corresponding closed form solution for the more general case of nested topologies is, however, still missing. Lenzini et al. [129] use linear programming to obtain individual solutions tailored to the specific scenario.

As already emphasized the order of systems that contribute to a network service curve according to (10) does not matter due to the commutativity and associativity of the min-plus convolution. Applying this result to the nesting tree in Fig. 12(b) it is possible to construct trees with equivalent network service curves, see [129].

The pay multiplexing only once approach above can, however, not be applied to non-nested cross-traffic flows. As an example consider the topology in Fig. 10(a). Here, it is not possible to construct a nesting tree and hence no good order of operations is known. Fidler [97] provides only a partial solution to this problem. Lenzini et al. [129] propose a heuristic that cuts the topology into nested sub-topologies to derive partial delay bounds that are summed up. The approach comes, however, at the cost of additive delay bounds that are known to be loose in many cases. Kim and Kim [130] propose an end-to-end delay bound for arbitrary topologies, however, without providing a stringent proof. To the best of our knowledge, a provably tight solution using the FIFO leftover service curve model is still unknown for non-nested topologies.

Schmitt et al. [133], [55], [131] investigate feed-forward topologies using leftover service curves under blind multiplexing (19). Compared to the FIFO model (20) the simpler blind multiplexing model is not parameterized. It causes, however, additional difficulty, since it builds on the assumption of strict service curves, a property that is not conserved by min-plus
convolution. To this end busy periods of tandem systems are cascaded in [131], [133] leading to a formulation of the network service curve as the solution of a minimization problem. Here, slack variables are defined that assign cross-traffic burstiness to systems in a worst-case manner, yet obeying the principle of paying multiplexing only once. The method applies also to non-nested topologies where it has been shown to provide tight bounds for a case study network using latency-rate service curves and leaky-bucket envelopes.

The derivation of network service curves investigated in this subsection generally assumes knowledge of the envelopes of cross-traffic flows $i,j$ at their multiplexing points $i$. If a cross-traffic flow has traversed one or more systems upstream of system $i$ its arrival envelope at system $i$ does not match its envelope at the ingress of the network due to known results on burstiness increase, for an example see Sect. IV-A. In this case, the same approach can be applied recursively to derive a service curve for the path from the ingress of the cross-traffic flow up to system $i$ to compute its arrival envelope at the multiplexing point beforehand, see e.g. the recursive implementations in [30], [31], [129].

C. Non feed-forward networks

In this section we report results for networks that may not be feed-forward. Yet, we still assume that networks are loop-free in the sense that a flow does not pass a system twice. For results on so-called re-entrant lines see e.g. the work by Winograd and Kumar [134].

As opposed to feed-forward networks no natural ordering of systems that would facilitate an inductive analysis exists in non feed-forward networks, i.e. there is no system to start the induction with. Moreover, it is not generally clear whether a non feed-forward network is stable or not, even if the utilization is smaller than one. The issue of stability has received considerable attention and certain sufficient conditions for stability are known today, see e.g. [2], [134], [135], [136], [137], yet a number of important questions are still open. A detailed survey on relevant works on stability of FIFO networks is beyond the scope of this work and is readily available provided by Cholvi and Echagüe [138].

1) Time stopping method: Cruz [2] introduced a method for analysis of non feed-forward networks referred to as time-stopping. The idea is to derive performance bounds for an interval $[0,\tau]$ only, i.e. arrival processes are virtually stopped at $\tau$. Since only a finite amount of data can enter the network in $[0,\tau]$ bounds exist generally. If it can be shown that the bounds obtained for the stopped sequence are independent of $\tau$ the solution holds also for the limit $\tau \to \infty$, i.e. for unstopped processes. Note that the existence of burstiness bounds implies stability [134]. For illustration see also [17], [16].

As an example consider the simple non feed-forward topology in Fig. 9(b). For notational simplicity assume rate service curves $S_i = R_it$ and leaky-bucket constrained arrivals with envelope $E_i^1(t) = \sigma_i^1 + \rho_it$ for flow $j$ at system $i$. A priori, the arrival envelopes $E_i^1$ and $E_i^2$ are assumed to be known, whereas the envelopes $E_2^1$ and $E_2^2$ are not. Time is stopped at $\tau$. Let $\sigma_1^1(\tau)$ and $\sigma_2^1(\tau)$ denote the unknown upper burstiness constraints for the interval $[0,\tau)$. Using the result on burstiness increase due to blind multiplexing from Sect. IV-A we have

$$\sigma_1^2(\tau) = \sigma_1^1 + \rho_1 \frac{\sigma_1^1(\tau)}{R_1 - \rho_1^2}$$

$$\sigma_2^1(\tau) = \sigma_2^1 + \rho_2 \frac{\sigma_1^1(\tau)}{R_2 - \rho_1^2}$$

The equation system can be solved numerically using an algorithm for fixed point iteration. Starting from some initial values, e.g. $\sigma_1^1(0) = \sigma_2^1(0) = 0$, the algorithm converges to the solution if it exists [135]. The CyNC network calculus toolbox by Schioler et al. [53] uses a related approach to obtain a solution from fixed point convergence.

If the equation system is solved analytically the goal is to find solutions for $\sigma_1^2(\tau)$ and $\sigma_2^1(\tau)$ that are independent of $\tau$ and thus hold for unstopped processes. The general approach is formulated in matrix notation in [2] where the respective eigenvalues allow specifying a sufficient condition for solvability and hence for stability. This method has for example been used in [139], [16] to recover an important result by Tassiulas and Georgiadis [136] proving the stability of ring topologies.

2) Topology-agnostic bounds: Charny and Le Boudec [140] use the concept of time-stopping to analyze a class of traffic that is served with priority, whereas FIFO applies within the class. The significant result is a closed-form representation of an end-to-end delay bound that does not depend on topology nor routing information. Merely information about the maximum path length called diameter is used. The bound is derived for arrivals that conform to leaky-bucket envelopes at the ingress of the network. Rate and burstiness of the flows are expressed by parameters $\alpha$ and $\tau$ where $\alpha$ is the maximum utilization of any of the links and $\tau$ is the maximum of the aggregated burstiness relative to the respective link capacities. In a simplified form (neglecting packetization and maximum link rates) the maximum delay in a network with diameter $h$ is bounded by

$$W_{\text{max}} \leq \frac{h\tau}{1-(h-1)\alpha}, \quad \text{if } \alpha < \frac{1}{h-1}. \quad (30)$$

Moreover, for any utilization $\alpha > (h-1)^{-1}$ it is shown in [140] that it is possible to construct a network topology where the worst-case delay can exceed any arbitrarily large value. This result implies that topology is key to determining a delay bound if it exists. Virtually identical results are also provided in [139], [113] where the latter also includes a delay bound for lossy systems with finite buffer.

Zhang, Duan, and Hou [141] extend the approach to EDF scheduling and devise a static as well as a dynamic deadline assignment algorithm. The authors recover the FIFO result and show a significant improvement for EDF where certain settings achieve stability for any utilization $\alpha < 1$ independently of the diameter $h$ as well as an end-to-end delay bound that scales linearly in $h$.

Jiang [142] derives a delay bound for the class of GR schedulers with FIFO aggregation. Paying specific attention to irregularities introduced by packets the delay bound in [142] extends the fluid-flow bound (30) by certain error terms that are due to packet scheduling. Moreover, Jiang applies the
notion of per-domain PSRG and derives the PSRG of networks with arbitrary topology from the delay bound [115], [114].

Wang et al. [31] derive flow-population insensitive delay bounds for networks with priority scheduling. The authors formulate an iterative solution for arbitrary topologies with potential circular dependencies. For a special case a delay bound that is independent of the network topology similar to (30) is derived. Moreover, the authors show a significantly improved delay bound that can be derived under the assumption of a feed-forward network.

3) Route interference number: Chlamtac et al. [143] propose an approach that makes use of flow route parameters to derive a source rate condition that achieves certain backlog and delay bounds. To this end, the authors devise the notion of Route Interference Number (RIN) that is the number of cross-traffic flows that are multiplexed along the path of a through flow, counting flows repeatedly that leave and join the path again. The source rate condition assumes that packets of a source are spaced out by a number of timeslots that is at least their RIN. Assuming time-slotted globally synchronized networks where all links have identical capacity, zero propagation delay, and employ FIFO scheduling finite backlog and delay bounds are shown where the delay bound coincides with the RIN. Key to these results are so-called delay chains that are used to prove that no two packets of the same flow join in a single backlogged period at any of the queues.

An improvement of the bounds in [143] and a proof of tightness for sink-tree topologies is provided by Zhang [144]. At the same time the result is derived by Le Boudec and Hébuterne [145] using the formal framework of the network calculus.

The discrete time model assumed in [143], [144], [145] corresponds to fixed sized cells as used in ATM networks. Otel and Le Boudec [146] use the network calculus to derive a RIN source rate condition for networks with variable sized packets and links with heterogeneous capacities and non-zero propagation delays. Flows are assumed to belong to a traffic class that is served with priority, whereas FIFO applies within the class. To this end strict service curves with latency-rate shape are used as the system model. Similarly, Yin and Poo [147] adapt the RIN approach to latency-rate lower and upper service curves. The authors analyze multi-class networks that can employ any scheduling algorithm that falls into the class of latency-rate servers.

Rizzo and Le Boudec [148], [137] investigate the more general case of latency-rate service curves and leaky-bucket constrained flows as opposed to the staircase functions specified by the strict source rate condition in [143]. The authors define operators that describe the evolution of certain network quantities over time. The recursion yields the super-additive closure of the respective operators for which the authors employ fixed-point solutions. Using this general approach and adopting the concept of delay chains from [143] a generalized RIN result is derived in [148]. In [137] the authors also recover the delay bound from [140].

VI. MEASUREMENT-BASED SERVICE CURVE ESTIMATION

The service curves that have been shown so far have been derived analytically using models, for example of buffered links, scheduling algorithms, and traffic regulators. The following section surveys methods that probe an unknown system or even a whole network to estimate its service curve from measurements of arrivals and departures, e.g. taken at the ingress and egress of a network. These methods dispense with detailed information about system internals, such as link capacities, cross-traffic intensities, and network topology, and merely assume that the system satisfies a certain definition of service curve.

Applications are manifold due to the fundamental challenges posed by the end-to-end principle of the Internet, where end-systems have to deal with minimal support from the network. To this end, cooperative hosts rely on measurements of their traffic arrivals and departures to estimate the state of the network and to adapt their transmission rate accordingly. A classic example is TCP congestion control.

The task of service curve estimation was first considered in the context of measurement-based admission control [34], [35], [36]. The approach is to analyze busy periods to estimate the offered service. The intuition behind these methods resorts to the strict service curve model. Recent studies focus on a formulation using the more general service curve model and concepts of the min-plus systems theory [86], [89], [49], [50], [149]. In this case the task of service curve estimation can be phrased as an inversion problem of the min-plus convolution.

The approaches can be further classified as passive or active. Passive methods resort to measurements of live traffic whereas active probing injects artificial probing traffic into the network. In case of active probing the traffic profile that is used for measurements can be controlled. This additional degree of freedom can be used to optimize the probing pattern.

An alternative approach is used by Jiang et al. [37] where cross-traffic is measured locally at a router. The method resorts to a Gaussian cross-traffic model whose parameters are estimated from measurements. With this model the leftover service curve that is offered to low priority traffic at a priority scheduler is computed. In this section we focus, however, on methods that rely on measurements taken at the network edge only.

A. Busy period analysis

The methods by Cetinkaya, Kanodia, and Knightly [34], [35] as well as by Valae [36] estimate the service that is provided during backlogged periods. The approach is closely related to the strict service curve model in (4). For clearness of exposition we phrase the underlying estimation problem of [34], [35], [36] as an optimization problem similar to [49]. The task is to find a maximal $S$ that fulfills the definition of strict service curve (4) for any pair of arrivals and departures $(A, D)$ that is

\[
\text{maximize } S
\]

subject to $D(t) \geq D(\tau) + S(t-\tau), \forall t \geq \tau \geq 0$ \hspace{1cm} (31)

assuming that the system is continuously backlogged in the interval $(0, t)$. Since strict service curves satisfy the general definition of service curve (3), a solution to the maximization problem (31) immediately provides a lower service curve.

The methods that will be discussed here use packet time stamps, that can be phrased elegantly in the max-plus algebra.
Along the same line the optimization problem in max-plus algebra becomes

\[
\begin{align*}
\text{minimize } & T_S \\
\text{subject to } & T_D(n) \leq T_D(\nu) + T_S(n - \nu), \forall n \geq \nu \geq 0
\end{align*}
\]

(32)

again assuming that the system is continuously backlogged. Here \(T_D(n)\) is the departure time of packet \(n\) and \(T_S(n)\) denotes the time that is required to serve \(n\) packets. For ease of exposition we assume that all packets have the same size. This assumption can be relaxed at the cost of additional notation, see e.g. [17]. Eventually, \(T_S(n)\) can be inverted to obtain the corresponding service curve \(S(t)\) in the min-plus algebra [17].

The above formulation requires that all possible sample paths of arrivals and departures, i.e., all pairs \((A, D)\), are considered whereas in practice only a limited set of measurements are available. The generalization of estimation results necessitates additional assumptions, such as that the system has a unique service curve that is not altered by the intensity of probing, respectively, through-traffic.

Equipped with a criterion for an optimal strict service curve we investigate the method in [34], [35] which uses passive measurements to estimate the service curve of a network. It uses time stamps of packet arrivals and departures \(T_A(n)\) and \(T_D(n)\) at the ingress and the egress of the network, respectively. The readings of arrivals and departures are disassembled into continuously backlogged periods. Given a single busy period an estimate of the service curve is computed as

\[
T_S(n) = \max_{\nu \geq 0} \{T_D(n + \nu) - T_A(\nu)\}.
\]

(33)

To verify that the estimate \(T_S\) satisfies the condition in (32) note that \(T_S(n) \geq T_D(n + \nu) - T_A(\nu)\) for all \(n, \nu \geq 0\) follows from (33). Moreover, \(T_A(\nu) \leq T_D(\nu)\) for causality such that \(T_D(n + \nu) \leq T_D(\nu) + T_S(n)\) satisfying the condition in (32). We note, however, that (33) does not attain the minimum in (32). The optimal solution is achieved by (33), if \(T_A(\nu)\) is replaced by \(T_D(\nu)\).

In [34], [35] estimated service curve samples are statistically aggregated to obtain mean service curves and their variance. These are used to formulate an admission control criterion. The approach is extended in [150], [151] where the scheduling algorithm that is employed by the network is estimated.

Related methods by Undheim, Jiang, and Emstad [152] as well as Alcuri, Barbera, and D’Acquisto [153] assume latency-rate service curves and seek to estimate the two parameters, i.e., latency and rate, from measurements obtained during a busy period. The basis of [152] are GRC and PSRG server models, respectively, see Sect. IV-E. The formulation uses max-plus algebra and the resulting parameter estimation problem is closely related to (32) whereas [153] builds on a formulation in min-plus algebra that resembles (31).

The work in [36] uses active probe packets with constant size. The probes are transmitted in a greedy fashion ensuring that the network is continuously backlogged. The method assumes that the network nodes implement separate queues for the probe traffic that are served in round robin order. While the round robin scheduler yields a service estimate that is related to the fair share, using a priority scheduler can provide an estimate of the remaining, unused service. A service estimate is computed from the packet departures as

\[
\hat{T}_S(n) = T_D(n + \nu) - T_D(\nu) - n T_t
\]

(34)

where \(T_t\) denotes the transmission time of a packet that is computed beforehand (assuming certain characteristics of a wireless link). Thus, only waiting times due to queuing are considered. Under the assumption that the estimates \(\hat{T}_S(n)\) are statistically independent of \(\nu\) the method computes percentiles of the service times, i.e., the resulting estimate \(\hat{T}_S(n)\) is violated at most with probability \(\varepsilon\). Letting \(\varepsilon = 0\) and neglecting the additional term \(n T_t\) (34) is a solution of the minimization problem (32). In a final inversion step a service curve is obtained in min-plus algebra.

B. Inversion of min-plus convolution

The estimation of service curves from an analysis of backlogged periods is intuitive and makes the use of the definition of strict service curves (4) appealing. The concept has, however, significant limitations that call the a priori assumption of strict service curves into question. To begin with, there exist systems that implement a service curve without fulfilling the definition of strict service curves, such as delay elements. Moreover, the strict service curve property does not naturally extend to tandem systems as the service curve property does, making the a priori assumption of strict service curves for networks questionable.

A formulation of the task of service curve estimation as an optimization problem is developed by Liebeherr, Fidler, and Valae [49]. The goal is to find a maximal lower service curve that fulfills the definition (3) for any pair of arrivals and departures \((A, D)\) that is

\[
\begin{align*}
\text{maximize } & S \\
\text{subject to } & D(t) \geq \inf_{\tau \in [0, t]} \{A(\tau) + S(t - \tau)\}, \forall t \geq 0.
\end{align*}
\]

(35)

Compared to the formulation using strict service curves this problem exhibits a number of important differences. First of all, there is no need to determine backlogged periods. Secondly, (35) makes use of min-plus convolution that has an important equivalent representation

\[
\begin{align*}
D(t) & \geq \inf_{\tau \in [0, t]} \{A(\tau) + S(t - \tau)\} \\
\Leftrightarrow & \exists \tau \in [0, t] : D(t) \geq A(\tau) + S(t - \tau).
\end{align*}
\]

Instead the constraint in (31) is required to hold for all \(\tau \in [0, t]\) which can be rephrased as

\[
\begin{align*}
D(t) & \geq D(\tau) + S(t - \tau) \quad \forall \tau \in [0, t] \\
\Leftrightarrow & D(t) \geq \sup_{\tau \in [0, t]} \{D(\tau) + S(t - \tau)\}.
\end{align*}
\]

Finally, (35) uses \(A(\tau)\) instead of \(D(\tau)\) in (31). Generally, \(A(\tau) \geq D(\tau)\) and \(A(\tau) = D(\tau)\) if the system is empty, e.g., at \(\tau = 0\). Hence, the constraint in (35) is generally weaker than the one in (31) resulting in better service curve estimates.

Similar to (31) the challenge of (35) is to find a service curve that holds for all possible pairs of arrivals and departures \((A, D)\) given only a limited set of measurements. A solution is assuming the existence of an exact service curve (12) that
reduces the estimation problem to an inversion of $D = A \otimes S$ for $S$ given readings of $(A, D)$. This assumption is justified for certain probes by arguments in min-plus linear systems theory and corresponding test criteria in [49], [50].

A passive measurement method for systems that implement an exact service curve is derived in [49] using the duality of min-plus convolution and de-convolution (14). Given a traffic trace of arrivals and departures $(A, D)$ the service curve estimate is computed as $\tilde{S}(t) = D \otimes A(t)$. It is proven in [49] that $\tilde{S}(t)$ can perfectly reconstruct $D(t)$ from $A(t)$. It is the best possible estimate that is justified by a given traffic trace.

The straightforward active probing solution to solve $D = A \otimes S$ for $S$ is the impulse function $\delta(t)$, see Sect. III, that is sending an infinite burst of data into the network. The burst impulse is the neutral element of min-plus convolution, such that measuring the departures $D(t) = \delta \otimes S(t) = S(t)$ immediately reveals the service curve. It is, however, shown in [49] that the approach violates the assumption of an exact service curve for certain systems, such as FIFO multiplexers.

In the literature two less intrusive methods have been reported that are based on the Legendre transform [49], [50], [86], [89]. The first approach that we mention here has been coined rate scanning in [49] as it scans a system using constant rate arrivals $A(t) = rt$ to estimate the maximum backlog $B_{\text{max}}$ at different rates $r$. The method uses the relation between the maximum backlog of a system and its service curve that is established by the Legendre transform (16) to obtain the service curve estimate

$$\tilde{S}(t) = L_B(t).$$

Alternatively, the Legendre transform facilitates the intended inversion of $D = A \otimes S$ in the Legendre domain where the min-plus convolution becomes an addition. Taking the inverse transform a service curve estimate is computed from readings of $A$ and $D$ as

$$\tilde{S}(t) = L(A(t) - L_A(t)).$$

A difficulty using this method is that both $L_A$ and $L_D$ can become infinite such that the difference is not defined. To this end, [49] uses specific probes, called rate chirps, that ensure finite $L_A$. The impulse function $\delta(t)$ can be viewed as a degenerate chirp having $L_A(r) = 0$ for all $r \geq 0$. For further details on the methods we refer to [49], [50].

VII. TIME-VARYING SYSTEMS

In Sect. III it is shown that systems have an exact service curve (4) if and only if they are time-invariant and min-plus linear. Otherwise, a system generally has non-identical lower and upper service curves. The assumption of time-invariance is relaxed in works by Chang, Cruz, Le Boudec, and Thiran [74], [76], as well as Agrawal, Baccelli, and Rajan [154] where a corresponding time-varying min-plus systems theory is developed. A compilation of the results can also be found in the textbook [17]. The framework can be extended to provide probabilistic guarantees [155], [156]. It is adopted as the basis of a stochastic network calculus in [17], [157].

The time-varying network calculus uses non-negative, non-decreasing bivariate functions to model arrivals, departures, and service guarantees. Thus, the respective functions are generally in

$$G_0 = \{g : g(\tau, t) \geq g(\tau, \vartheta) \geq 0 \ \forall t \geq \vartheta \geq \tau, \ g(t, t) = 0 \ \forall t\}.$$ 

Clearly the arrivals and departures of a system $A(\tau, t)$ and $D(\tau, t)$, respectively, are in $G_0$.

A system has a time-varying lower service curve $S(\tau, t)$ if it holds for all $t \geq 0$ that

$$D(t) \geq \sup_{\tau \in [0, t]} \{A(\tau) + S(\tau, t)\} =: A \otimes S(t)$$

where $\otimes$ is the accordingly extended definition of min-plus convolution. A function $\tilde{S}(t)$ that satisfies $S(t - \tau) \leq S(\tau, t)$ for all $t \geq \tau \geq 0$ is a univariate lower service curve of the system. A time-varying system has an exact service curve, i.e. (36) holds with equality, if and only if the system is min-plus linear. An example that is provided in [74], [17], [76] are work-conserving links with a time-varying capacity.

As in the time-invariant case the min-plus convolution is associative facilitating the concatenation of tandem systems into a single equivalent system. Hence, two systems with service curves $S_1(\tau, t)$ and $S_2(\tau, t)$ in series behave as a single system with service curve

$$S(\tau, t) = \inf_{\vartheta \in [\tau, t]} \{S_1(\tau, \vartheta) + S_2(\vartheta, t)\} = S_1 \otimes S_2(\tau, t)$$

and by iterative application any number of tandem systems can be collapsed into a single system. Note, however, that the convolution of bivariate functions is not commutative. For an overview on the algebraic properties see [74], [17], [76].

Mimicking the definition of envelope, as used for example in [157]. A bivariate upper envelope $E(\tau, t)$ for all $t \geq \tau$ that

$$A(\tau, t) \leq E(\tau, t).$$

Note that $A(\tau, t)$ itself satisfies the definition of envelope, as used for example in [157].

A bivariate envelope of the departures of a time-varying system can be derived as

$$F(\tau, t) = \sup_{\vartheta \in [0, \tau]} \{E(\vartheta, t) - S(\vartheta, \tau)\} =: E \otimes S(\tau, t)$$

where $\otimes$ denotes the accordingly extended definition of min-plus de-convolution. The backlog at time $t$ is bounded as

$$B(t) \leq E \otimes S(t, t)$$

and in case of FIFO scheduling the delay of data arriving at time $t$ is bounded as

$$W(t) \leq \inf \{w \geq 0 : E \otimes S(t + w, t) \leq 0\}.$$ 

Bounds for the maximum backlog $B_{\text{max}}$ and the maximum delay $W_{\text{max}}$ follow by taking the supremum over all $t$.

The works in [74], [17], [76] solve the problem of traffic regulation under backlog and delay constraints in the time-varying systems theory. The authors derive an optimal regulator in the sense that it minimizes the number of discarded packets. The regulator consists of a bufferless clipper, which discards packets if necessary, and a buffered traffic regulator. Further on, the authors phrase window flow control systems as well as SCED schedulers in the time-varying systems theory.
In [157] it is shown that the leftover service curve model under blind multiplexing can be extended to the time-varying case. Through traffic at a work-conserving server with time-varying capacity denoted $S(\tau, t)$ and cross-traffic with envelope $E(\tau, t)$ sees a leftover service curve

$$S^1(\tau, t) = [S(\tau, t) - E(\tau, t)]^+.$$  

(VIII. STOCHASTIC NETWORK CALCULUS)

The deterministic network calculus provides an intuitive framework that has outstanding qualities with respect to system modeling as well as concatenation of tandem systems. A significant shortcoming is, however, the worst-case view that frequently results in overly pessimistic performance bounds. Even if certain bounds are provably tight the worst case may be attained rarely. Most applications on the other hand tolerate occasional quality of service violations, e.g. audio and video transmissions achieve an acceptable quality even if a certain fraction of data are lost or delayed in the network. See [158] for a discussion of client requirements for real-time communications.

Moreover, the deterministic envelope model is additive, see Sect. II-C, i.e. the envelope of an aggregate of flows increases linearly with the number of flows that are multiplexed, unless specific information about the phase of flows is known [159]. In contrast, packet switching, that is the basic technology of the Internet, saves significant amounts of resources due to statistical effects when multiplexing variable bit rate flows. This important statistical multiplexing gain cannot be utilized by deterministic methods. It can, however, be exploited efficiently using statistical envelopes as presented in the sequel, see e.g. [160].

Lastly, numerous systems exist which cannot provide deterministic service guarantees as assumed by the network calculus. Deterministic leftover service curves as derived in Sect. IV assume that a worst-case envelope exists for cross-traffic, whereas the actual leftover service is random as a consequence of the variability of cross-traffic. Likewise, the service offered by a radio channel is random due to fading and interference as well as randomized multiple-access protocols.

Stochastic extensions of the network calculus have been of significant interest to overcome the limitations of the deterministic framework and to utilize the statistical multiplexing gain. The service guarantees that are typically provided are percentiles, i.e. statistical bounds with a certain violation probability $\varepsilon$, e.g. [161] considers

$$\text{P}[\text{delay} > x] \leq \varepsilon \quad \text{or} \quad \text{P}[\text{loss}] \leq \varepsilon.$$  

Usually a small $\varepsilon$ that excludes only certain rare cases, e.g. in the order of $10^{-6}$, effectuates a large statistical gain and has the potential to improve Internet performance bounds in the order of magnitudes.

The remainder of this section is structured as follows. In Sect. VIII-A we briefly provide the required background on statistical envelopes, for a detailed overview see [19], [66], and show how performance bounds for single systems can be derived. Sect. VIII-B deals with stochastic service curves and in Sect. VIII-C we discuss the difficulty of statistical end-to-end analysis of tandem systems and survey recent solutions. We conclude with relevant open issues in Sect. VIII-D.

A. Envelopes and single system performance bounds

The concept of statistical envelopes dates back to the seminal works by Kurose [162] and Chang [135]. Kurose uses stochastic bounds for arrivals seen in time intervals of different duration. Chang formulates a stochastic extension of Cruz’s $(\sigma, \rho)$ traffic model [1] using affine bounds $(\sigma(\theta), \rho(\theta))$ for the moment generating function of traffic arrivals. Since then, numerous statistical traffic envelope models as well as methods for their construction have been developed, including self-similar [163] and multi-fractal processes [164], see the recent survey of envelopes by Mao and Panwar [19].

In a recent work, Ciuca [66] distinguishes statistical envelopes as random processes e.g. [162], [135], [165] and as non-random functions e.g. [166], [161]. The view of envelopes as random processes matches the definition of bivariate envelopes in the time-varying systems theory, see Sect. VII. In this case, the envelope $E(\tau, t)$ is a random process and the smaller equal relation in its definition (37) is treated either as stochastic ordering or as almost surely ordering [66]. In this sense, the arrivals $A(\tau, t)$ itself are an envelope.

The notion of statistical envelopes as non-random functions has been developed starting with the Exponentially Bounded Burstiness (EBB) model by Yaron and Sidi [166], [167] and the subsequent Stochastically Bounded Burstiness (SBB) model by Starobinski and Sidi [168], as well as effective envelopes by Boorstyn et al. [161]. A powerful model that became evident since then is defined as follows, see [66], [18] for further discussion: Arrivals $A(t)$ have a statistical envelope $E(t)$ if for all $t \geq \tau \geq 0$ it holds that

$$\text{P}[A(\tau, t) > E(t - \tau) + \sigma] \leq \varepsilon(\sigma)$$  

where the parameterized violation probability $\varepsilon(\sigma)$ is referred to as overflow profile or error function. Letting $\sigma = 0$ the earlier definition used in [161] that is an immediate statistical extension of (6) is recovered. The parameter $\sigma$ plays, however, an important role for the construction of performance bounds later on.

The statistical envelope (39) includes, respectively, builds on numerous important traffic models that have been developed previously. For $E(t) = \rho t$ and $\varepsilon(\sigma) = \alpha e^{-\sigma}$ the basic Exponentially Bounded Burstiness (EBB) model

$$\text{P}[A(\tau, t) > \rho(t - \tau) + \sigma] \leq \alpha e^{-\sigma}$$  

by Yaron and Sidi [166] is recovered. Linked together by the use of Chernoff’s bound\(^3\), Chang’s $(\sigma(\theta), \rho(\theta))$ traffic model [170] is also EBB and the converse results from [92]. Similarly, the related Stochastically Bounded Burstiness (SBB) model by Starobinski and Sidi [168] follows from (39) for $E(t) = \rho t$ under the additional condition that the error function $\varepsilon(\sigma)$ is $n$-fold integrable. Compared to the EBB model the generalization of SBB includes for example fBm traffic with Hurst parameter $1/2 < H < 1$, i.e. traffic with long range dependence. An envelope (39) for fBm is defined in [163]. The relation between the different envelope models is elaborated in [66], [18].

\(^3\)Given a random variable $X$ with moment generating function $M_X(\theta) = E[e^{\theta X}]$ Chernoff’s bound states that $\text{P}[X \geq x] \leq e^{-\theta x} M_X(\theta)$ for all $\theta \geq 0$, i.e. the right hand side can be minimized for $\theta$, see e.g. [169].
Different approaches to derive statistical envelopes are known today. Originating from the use of leaky-bucket shapers, statistical envelopes for aggregates of regulated flows have created considerable interest. The scenario is investigated by Kesidis and Konstantopoulos [171], [172], Knightly and Wu [173], [174], [175], Reisslein, Rajagopal, and Ross [176], [177], [178], [179], Boorstyn and Liebeherr et al. [161], [160], as well as Vojnović and Le Boudec [180].

Here, we only briefly mention two general methods for construction of statistical envelopes, see [19] for a broader overview. One option is rate-variance envelopes defined by Knightly et al. [174], [181], [182] where the central limit theorem can be used to determine statistical envelopes as well as statistical performance bounds for aggregates of traffic flows, see also [161] and [183].

Alternatively, envelopes can be computed from the moment generating function of traffic arrivals defined as $M_A(\theta, t) = \mathbb{E}[e^{\theta A(t)}]$. A statistical envelope with free parameter $\theta \geq 0$ follows from Chernoff’s bound as $|P[A(\tau, \tau+t) > E(t)]| \leq e^{-\theta E(t)} M_A(\theta, t)$.

Assuming stationarity the envelope applies for all $\tau \geq 0$. Moment generating functions of numerous traffic models including regulated, Markov on-off, periodic, and fBm sources as well as traffic traces are known from the theory of effective bandwidths, see e.g. [17], [184], [185], [186], [187] and references therein. A method for conversion of effective bandwidths into statistical envelopes and vice versa is provided by Li, Burchard, and Liebeherr [92].

Given a statistical envelope for traffic arrivals at a system with deterministic service curve $S(t)$ the task is to derive statistical performance bounds. For brevity we restrict the exposition to backlog bounds to show the general approach. The computation of delay bounds and output envelopes, see Sect. II-C, faces essentially the same difficulties and follows analogously. The backlog is defined as $B(t) = A(t) - D(t)$ and by insertion of the definition of service curve $D(t) \geq A \otimes S(t)$ it follows that

$$B(t) \leq \sup_{\tau \in [0,t]} \{A(\tau, t) - S(t-\tau)\}.$$ 

Next, the definition of statistical envelope (39) has to be inserted to determine the violation probability $P[B(t) > b]$ of a backlog bound $b$. As opposed to the deterministic case this is, however, not straightforward since the desirable simplification

$$P\left[\sup_{\tau \in [0,t]} \{A(\tau, t) - S(t-\tau)\} > b\right] \geq \sup_{\tau \in [0,t]} P[A(\tau, t) - S(t-\tau) > b]$$

provides only a lower bound for the violation probability. The lower bound may be justified as an acceptable approximation in certain carefully selected cases, see e.g. [183] and references therein. A rigorous upper bound, on the other hand, follows by taking the union over all $\tau$. For further discussion see “What makes statistical network calculus hard?” in [100]. With Boole’s inequality if follows that

$$P\left[\sup_{\tau \in [0,t]} \{A(\tau, t) - S(t-\tau)\} > b\right] \leq \sum_{\tau = 0}^{t} P[A(\tau, t) - S(t-\tau) > b].$$

(41)

The last step implies that time is discrete. Otherwise, continuous-time functions have to be sampled. Since functions in the network calculus are non-decreasing, it is generally possible to lower or upper bound the function values in an interval of width $\tau_0$ using a sample taken at the left, respectively, right border of the interval. Sampling is employed in the statistical network calculus by Ciucu, Burchard, and Liebeherr [65] where the gain from optimizing the width of the sampling interval $\tau_0$ is utilized.

Inserting the definition of statistical envelope (39), e.g. $\sigma = 0$ into (41), it follows from Boole’s inequality that the backlog bound $b = E \otimes S(0)$ is violated at most with probability $\varepsilon$.

Clearly, the violation probability increases with $t$ and becomes unbounded for $t \to \infty$. This problem can be circumvented assuming finite time-scales $t \leq T$.

One approach is to restrict the duration of busy periods, e.g. given an arrival envelope $E(t)$ and a strict service curve $S(t)$ the maximum busy period is bounded by $T$ if $E(t) \leq S(t)$ for all $t \geq T$. Busy period bounds have been used by Knightly [173], Li, Burchard, and Liebeherr [92], Liebeherr, Patek and Burchard [188], Vojnović and Le Boudec [189], and recently by Angrishi and Killat [190]. The derivation of such time-scale bounds for multi-node networks is, however, nontrivial.

Busy period bounds for downstream nodes are shown in [92], [188]. Another option is to derive time-scale bounds under the assumption that nodes drop excess traffic that violates a buffer limit or a delay constraint as used in [77], [78], [79].

Much research has been devoted to avoiding the necessity of such time-scale bounds entirely. Burchard, Liebeherr, and Patek [191] show a solution that uses a stochastic extension of the adaptive service curve model (5). An alternative solution is the use of decaying violation probabilities $\varepsilon(\sigma)$ that satisfy certain integrability conditions [166], [135], [168], [157]. A solution for the EBB traffic model was already shown in [166] by Yaron and Sidi. Assuming EBB arrivals at a constant rate server $S(t) = C t$ (41) can be rewritten as

$$\sum_{\tau = 0}^{t} P[A(\tau, t) - C(t-\tau) > b] \leq \sum_{\tau = 0}^{\infty} \alpha e^{-\theta b + (C-\rho)\tau}$$

where the EBB model (40) is instantiated with $\sigma = b + (C-\rho)(t-\tau)$. If $C > \rho$ for stability a solution is obtained using the geometric sum resulting in the exponentially decaying backlog bound

$$P[B(t) > b] \leq \frac{\alpha e^{-\theta b}}{1 - e^{-\theta (C-\rho)}}.$$ 

(42)

Conversely, if an exponential bound on the backlog is known the traffic is EBB [166]. Similar results can be derived for SBB traffic owing to the integrability condition of the SBB model [168]. Concluding, the addition of $\sigma$ to the definition of statistical envelope (39) and the integrability of $\varepsilon(\sigma)$ resolve the necessity of time-scale bounds. Intuitively, the envelope that is used has a violation probability that decays with $t$ to achieve convergence of the geometric sum.

Using the arrival function as an envelope a related result is derived by Chang [135] for the $(\sigma(\theta), \rho(\theta))$ traffic model.

The geometric sum is $\sum_{n=0}^{\infty} x^n = (1 - x)^{-1}$ for $|x| < 1$. Owing to the non-decreasing nature of functions in the network calculus it is usually possible to estimate sums by integrals that may be more convenient to solve.
From (41) it follows with Chernoff’s bound that
\[
\sum_{\tau=0}^{t} \mathbb{P}[A(\tau, t) - C(t-\tau) > b] \leq \sum_{\tau=0}^{t} e^{-\theta_B} \mathbb{E}\left[ e^{\theta(A(\tau, t) - C(t-\tau))} \right].
\]
Bounding \( \mathbb{E}[e^{\theta A(\tau, t)}] \leq e^{\theta (\sigma + \rho(t))} \), assuming \( C > \rho \) for stability, and using the geometric sum yields
\[
\mathbb{P}[B(t) > b] \leq \frac{e^{\theta \sigma} e^{-\theta B}}{1 - e^{-\theta (C + \rho)}}.
\]
The relation to the EBB result (42) is explained by the fact that \((\sigma, \rho)\) is EBB (40) with \( \alpha = e^{\theta A} \) [66].

The EBB approach has motivated significant research that led to the notion of sample path envelopes developed by Cruz [156], Ayayorg and Feng [192], Yin, Jiang, Jiang, and Kong [193], Burchard, Liebeherr, and Patek [191], Ciucu, Burchard, and Liebeherr [65], and Jiang [194]. Compared to the definition of envelope (39) that is subject to a point-wise violation probability, sample path envelopes are subject to a path-wise violation probability. Such envelopes largely simplify the computation of performance bounds. The sample path envelope that corresponds directly to (39) is defined for all \( t \) as [156], [192], [66], [18]
\[
\mathbb{P}\left[ \exists \tau \in [0, t] : A(\tau, t) > E(t-\tau) + \sigma \right] \leq \varepsilon(\sigma). \tag{43}
\]
Note that the formulation \( \forall \tau \in [0, t] : A(\tau, t) > E(t-\tau) + \sigma \) in (43) is equivalent to \( \sup_{\tau \in [0, t]} \{ A(\tau, t) - E(t-\tau) \} > \sigma \) as used in [66] and for the special case \( \sigma = 0 \) in [191]. The distinction of sample path envelopes from point-wise envelopes does not exist in the deterministic case (6).

Closely related is the generalized SBB (gSBB) model by Yin, Jiang, Jiang, and Kong [193] that is a sample path extension of SBB. Letting \( E(t-\tau) = \rho \cdot (t - \tau) \) the gSBB model follows as a special case of (43). A method for construction of gSBB envelopes from the SBB model is developed in [193]. Moreover, it is shown therein that gSBB envelopes follow immediately from a known backlog bound. Further stochastic network calculus results for gSBB traffic are also provided in [195].

Sample path envelopes (43) can be constructed from the definition of envelope (39) (in the same way as gSBB envelopes can be constructed from SBB) using Boole’s inequality, i.e.
\[
\mathbb{P}\left[ \sup_{\tau \in [0, t]} \{ A(\tau, t) - E(t-\tau) \} > \sigma \right] \leq \sum_{\tau=0}^{t} \mathbb{P}[A(\tau, t) - E(t-\tau) > \sigma].
\]
The method closely resembles the derivation of the backlog bound (41). It assumes similar conditions on the integrability of \( \varepsilon(\sigma) \). In case of the backlog bound it implies excess capacity \( x > 0 \) where \( x = C - \rho \). The same condition applies for construction of sample path envelopes resulting in a slackening of the sample path envelope by some rate \( x > 0 \), i.e. letting the sample path envelope \( E(t) = E_{pw}(t) + xt \) where \( E_{pw}(t) \) is the envelope with point-wise violation probability \( \varepsilon_{pw}(\sigma) \) as defined in (39). By insertion the violation probability of the sample path envelope (43) is bounded by \( \varepsilon(\sigma) = \sum_{\tau=0}^{\infty} \varepsilon_{pw}(\sigma + \tau x) \), see [66], [18] for details.

Note that the parameter \( \sigma \) in the EBB model and in the \((\sigma, \rho)\) model need not be identical.

Given arrivals with a sample path envelope at a system with deterministic service curve it is straightforward to compute performance bounds from (deterministic) sample path arguments. In case of the backlog it follows that
\[
\mathbb{P}[B(t) > \sigma + E \sup S(0)] \leq \varepsilon(\sigma).
\]
For \( \varepsilon(\sigma) = 0 \) the deterministic case is recovered. Delay and output bounds follow similarly [66], [18].

Considering the above construction for the case of EBB arrivals the sample path envelope \( E(t) = (\rho + x)t \) where \( x > 0 \) has violation probability
\[
\varepsilon(\sigma) = \sum_{\tau=0}^{\infty} \alpha e^{-\theta(\sigma + \tau x)} = \frac{\alpha e^{-\theta \sigma}}{1 - e^{-\theta x}}.
\]
Assuming a constant rate server with capacity \( C \) and letting \( x = C - \rho \) results in \( E \sup S(0) = 0 \). Substituting \( b \) for \( \sigma \) the previous backlog bound (42) is recovered.

The close relation of sample path envelopes and backlog bounds becomes most apparent in case of the gSBB model. Letting \( E(t) = \rho t \) in (43) gives
\[
\mathbb{P}\left[ \sup_{\tau \in [0, t]} \{ A(\tau, t) - \rho(t-\tau) \} > \sigma \right] \leq \varepsilon(\sigma)
\]
where \( \sup_{\tau \in [0, t]} \{ A(\tau, t) - \rho(t-\tau) \} \) equals the backlog \( B(t) \) at a constant rate server with rate \( \rho \). Hence, the gSBB traffic characterization follows immediately from a statistical backlog bound [193]. Consequently, [194] classifies the envelope (39) as traffic amount centric and the sample path envelope (43) as virtual backlog centric. Taking an additional supremum [194] defines a maximum virtual backlog centric envelope, e.g. \( \mathbb{P}[\sup_{\tau \geq 0} B(t)] > \sigma \). Under the assumption of stationarity and ergodicity [191] shows, however, that the definition of maximum virtual backlog centric envelopes is deterministic.

B. Stochastic service curves

Frequently, not only the arrivals but also the service provided by links or networks is stochastic. This randomness is an inherent characteristic of the service left over by variable bitrate cross-traffic [98], [99], [188], [92], [157], [194]. Besides, different types of random impairment may occur [156], [196], e.g. due to the variability of the service provided by radio channels [197], [198] or due to CSMA/CA [199] and random access control. Stochastic service curves can efficiently model these effects. The concept mimics certain properties of statistical envelopes. Service curves exhibit, however, important differences that will be discussed in the sequel.

As for envelopes Ciucu [66] distinguishes stochastic service curves as random process and as non-random functions. The former can be viewed as a stochastic interpretation of time-varying service curve (36) introduced by Chang [135], [155], [17]. An example is the leftover service curve model (38) [157] that applies in case of random cross-traffic. The approach by Chang [17] and Fidler [157] uses bounds of moment generating functions to model stochastic arrival envelopes and service curves. Assuming stationarity we write \( M_A(\theta, t) = \mathbb{E}[e^{\theta A(\tau, t+t)}] \) and \( \mathbb{M}_S(\theta, t) = \mathbb{E}[e^{\theta S(\tau, t+t)}] \) for all \( \tau \geq 0 \) to denote the moment generating function,
respectively. Laplace transform of arrivals and service. Under the assumption of statistical independence of arrivals and service a stochastic backlog bound follows by application of Boole’s inequality and Chernoff’s bound as

\[ P[B(t) > b] \leq e^{-\theta b} \sum_{\tau=0}^{\infty} M_A(\theta, \tau) M_S(\theta, \tau) \]  \tag{44} \]

Performance bounds that do not require assumptions about time-scale bounds are derived in [157] for the class of \((\sigma(\theta), \rho(\theta))\)-constrained through and cross-traffic. The derivations of output envelopes and delay bounds follow similarly and are omitted here, see [17], [157] for details. The advantage of the approach is that it can make efficient use of statistical independence. If this assumption does not hold the computation of bounds becomes significantly more complicated, as outlined in [157].

Ciucu [200], [201], [202] investigates the special case of processes with stationary and independent increments and shows conditions where the network calculus recovers exact solutions from queuing theory, e.g. M|M|1 and M|D|1, with reasonable accuracy. The general approach is also used by Angrishi and Killat [203], [204] and used with time-scale bounds in [190].

In analogy to the EBB traffic characterization Lee [205] presents an Exponentially Bounded Fluctuation (EBF) model for stochastic servers. A fundamental definition of stochastic service curve as a non-random function is presented by Cruz [156] and adopted with modifications in [188], [92], [65], [66], [194], [18], [38], [195]. A system with arrivals \(A(t)\) and departures \(D(t)\) is said to offer a stochastic service curve \(S(t)\) if it holds for all \(t \geq 0\) that

\[ P[D(t) < A \otimes S(t) - \sigma] \leq \varepsilon(\sigma) \]  \tag{45} \]

where \(\varepsilon(\sigma)\) is referred to as deficit profile or error function. Note that the definition (45) already implies a sample path argument that is due to the use of min-plus convolution. Similarly, the sample path envelope (43) can be rewritten as

\[ P[A(t) > A \otimes E(t) + \sigma] \leq \varepsilon(\sigma). \]

The works [188], [92] use (45) for \(\sigma = 0\) in a framework with time-scale bounds. A definition of stochastic service curve that is based on the concept of adaptive service guarantees (5) is presented in [191]. A variation of (45) is presented in [65], [66] where a positivity constraint \([S(t) - \sigma]^+\) is added to the definition. The papers [194], [18], [38], [195] make the same definition, whereby [194], [18] refer to the definition as weak stochastic service curve and propose a stronger definition that assumes an a priori sample path bound for the departures of the form \(P[\exists t \geq 0: D(t) < A \otimes S(t) - \sigma] \leq \varepsilon(\sigma)\). Assuming stationarity and ergodicity it is, however, argued in [191] that this definition essentially is deterministic, i.e. \(\varepsilon(\sigma)\) is either 0 or 1.

The definition of stochastic service curve (45) facilitates deriving statistical single node performance bounds. As before we show only the backlog bound that was first presented in [156]. Given arrivals with sample path envelope \(E(t)\) and overflow profile \(e_E(\sigma)\) according to (43) at a system with stochastic service curve \(S(t)\) and deficit profile \(e_S(\sigma)\) according to (45) it follows for all \(t \geq 0\) that

\[ P[B(t) > E \otimes S(0) + \sigma] \leq \varepsilon_B(\sigma) \text{ where } \varepsilon_B(\sigma) = e_E \otimes e_S(\sigma). \]

As opposed to (44) the derivation does not require any assumptions on statistical independence of arrivals and service. Delay bounds and output envelopes follow similarly and are presented e.g. in [156], [65], [66], [195], [18].

An important field of application of the stochastic network calculus are traffic scheduling algorithms and leftover service curves under cross-traffic as discussed for the deterministic case in Sect. IV. Here, we restrict the exposition of stochastic leftover service curves to the general blind multiplexing model. Given a system with capacity \(C\), i.e. \(S(t) = Ct\) and cross-traffic arrivals with sample path envelope \(E^c(t)\) and overflow profile \(\varepsilon(\sigma)\) according to (43) the system offers a leftover stochastic service curve \(S^c(t) = [S(t) - E^c(t)]^+\) with deficit profile \(\varepsilon(\sigma)\) according to (45) to through-traffic [65].

Stochastic results for GPS, EDF, and SP that extend the deterministic models in Sect. IV are presented by Qiu, Knightly, Cetinkaya, and Li in [98], [99], by Li, Burchard, and Liebeherr in [100], [92], and for GPS, SP, and FIFO by Jiang, Yin, Liu, and Jiang in [206]. A finding in [100], [92] is that in most practical cases the gain of scheduling algorithms is small compared to the gain of statistical over deterministic service guarantees. A comparison of FIFO multiplexing and blind multiplexing [202] provides stochastic bounds on the output burstiness of through-traffic in the presence of cross-traffic. For EBB flows it is found that statistical output envelopes converge exponentially fast to the input envelopes supporting a decomposability of FIFO networks.

While most of the work on stochastic service curves considers either fluid systems (in continuous time) or systems with fixed packet size (in discrete time) Burchard, Liebeherr, and Ciucu [207] analyze the effects of variable-sized packets that follow a given random distribution, e.g. exponential or Pareto, on network performance. In [66] Ciucu extends the concept of packetizer to random packet sizes and phrases the packetizer as a stochastic service curve with a defined deficit profile.

C. End-to-end concatenation of tandem systems

Like the theory of effective bandwidths the stochastic network calculus can effectively utilize the statistical multiplexing gain of independent flows. A significant advantage of the network calculus are leftover service curve models for scheduling algorithms that facilitate the analysis of networks with cross-traffic. Using the concept of (leftover) service curves the main goal of recent research is to recover the concatenation theorem (10) in the stochastic network calculus to derive statistical end-to-end performance bounds.

Let us recall the main results of the deterministic end-to-end analysis from Sect. II-D. Given arrivals with envelope \(E(t)\) at a series of systems indexed \(i = 1 \ldots n\) with service curves \(S_i(t)\) as shown in Fig. 5. Two options are known to compute an end-to-end delay bound. Either per-node delay bounds are summed up, which requires computation of output envelopes (7) at all systems \(i = 1 \ldots n-1\) that are the input envelope of the subsequent systems \(i + 1\), respectively. Alternatively, the service curves are convolved (10) to identify an equivalent single server system for which a delay bound is computed in one step. Given \(n\) systems in tandem, the additive bound grows with \(O(n^2)\). In contrast the pay bursts only once result from end-to-end convolution is in \(O(n)\), see Sect. II-D.
Basically the same two options exist in the stochastic network calculus, however, obtaining an end-to-end convolution form has been a long-standing problem. Consequently, numerous earlier works [162], [166], [168], [135], [208] but also recent papers, e.g. [209], use the node-by-node approach. This comes at the cost of loose end-to-end bounds. Recently, Ciucu, Liebeherr, and Burchard [65] showed that additive delay bounds from the EBB model actually scale in $O(n^3)$.

Before going into details of deriving a stochastic network service curve we recapitulate the difficulty that has to be solved [100], [191]. Given two systems with stochastic service curves (45) $S_1(t)$ and $S_2(t)$ in tandem. Denote $A_i(t), D_i(t)$ the arrivals and departures, respectively, of system $i$ where $A_i(t) = D_i(t)$. Letting $\sigma_i = 0$ it holds that

\[
\mathbb{P} \left[ D_2(t) < \inf_{\tau \in [0,t]} \{ D_1(\tau) + S_2(t - \tau) \} \right] \leq \varepsilon_2
\]

\[
\mathbb{P} \left[ D_1(\tau) < \inf_{\delta \in [0,\tau]} \{ A_1(\delta) + S_1(t - \delta) \} \right] \leq \varepsilon_1.
\]

As pointed out in [100], [191] $D_1(\tau)$ in the first line can not be substituted for the second line as in the deterministic case since $\tau$ is a random variable that cannot be easily fixed. Instead, a sample path bound for $D_1(\tau)$ is required, i.e. a bound that holds for all $\tau \in [0, t]$. Invoking Boole’s inequality the violation probability of such a sample path bound can be estimated as $t \varepsilon_1$. Clearly, without further assumptions this probability is unbounded as $t$ grows to infinity.

A solution to the problem of unbounded sample path violation probabilities is to bound the relevant time-scale using busy period bounds as proposed by Li, Burchard, and Liebeherr [92]. Given a time-scale $T$ the sample path violation probability from Boole’s inequality in the above example remains bounded by $T \varepsilon_1$. Time-scale bounds have later on been adopted e.g. in [190], [210]. As discussed in Sect. VIII-A the assumption of a priori time-scale bounds is, however, difficult to deal with.

An option that avoids the assumption of a priori bounds is proposed by Burchard, Liebeherr, and Patek [191] using a stochastic extension of the concept of adaptive service guarantees (5). The accordingly modified convolution operator is interval-based and yields the desired end-to-end convolution form with bounded violation probability.

A further option is the definition of service curves with loss by Ayyorgun and Cruz [79]. The authors construct a composable service curve model for end-to-end analysis under the assumption that traffic is discarded if it violates a deadline. The discard policy requires, however, a specific scheduler.

Recently, Ciucu, Burchard, and Liebeherr [65] derive sample path bounds for the departures of a stochastic service curve element that do not depend on time-scale bounds. A basic version of the sample path stochastic service curve can be formulated as

\[
\mathbb{P} \left[ \exists \tau \in [0, t] : D(\tau) < A \otimes S(\tau) - g(t - \tau) - \sigma \leq \varepsilon^c(\sigma) \right]. \tag{46}
\]

for all $t \geq 0$ and any choice of $g > 0$. The construction of sample path service curves from the definition of stochastic service curves (45) uses Boole’s inequality. The sample path service curve (46) can be rewritten and bounded as

\[
\mathbb{P} \left[ \sup_{\tau \in [0, t]} \{ A \otimes S(\tau) - g(t - \tau) - \sigma - D(\tau) \} > 0 \right] \leq \sum_{\tau = 0}^{t} \mathbb{P}[D(\tau) < A \otimes S(\tau) - g(t - \tau) - \sigma]
\]

Inserting the definition of stochastic service curve from (45) with violation probability $\varepsilon(\sigma)$ and letting $t \to \infty$ yields a service curve with sample-path violation probability $\varepsilon^c(\sigma) = \sum_{\tau = 0}^{\infty} \varepsilon(\sigma + g\tau)$. As in case of statistical envelopes integrability of the error functions is assumed. By recursive insertion a network service curve for tandem systems can be derived from the definition of sample path service curves as

\[
S_{\text{net}}(t) = S_1 \otimes S_2^\infty \otimes \cdots \otimes S_n^{(n-1)}(t)
\]

where $S^{-\infty}(t) = S(t) - gt$. The network service curve has deficit profile [65]

\[
\varepsilon_{\text{net}}(b) = \varepsilon_1^b \otimes \varepsilon_2^b \otimes \cdots \otimes \varepsilon_n^{b-1} \otimes \varepsilon_n(b).
\]

Ciucu, Burchard, and Liebeherr [65] investigate line topologies with single-hop persistent cross-traffic, see Fig. 8, and derive statistical delay bounds for EBB traffic from network service curves. They improve the additive EBB bound in $O(n^3)$ significantly and derive statistical delay bounds from end-to-end convolution that grow in $O(n \log n)$. In consecutive work Burchard, Liebeherr, and Ciucu [207] show that $\Omega(n \log n)$ is also a lower bound for packetized EBB traffic and hence the result is provably tight.

Recent studies [211] and [212] seek to solve the problem using expectations instead of probabilities. The approach is inviting since the expected value is a linear operation in systems theory [169] such that the order of expectation and convolution can be exchanged. Assuming statistical independence of $S_1$ and $S_2$ it holds that

\[
\mathbb{E}[S_1 \ast S_2(t)] = \mathbb{E}[S_1] \ast \mathbb{E}[S_2](t)
\]

where $\ast$ is the convolution in classical algebra. Under the min-plus algebra we obtain, however, only a (not useful) upper bound

\[
\mathbb{E}[S_1 \otimes S_2(t)] \leq \mathbb{E}[S_1] \otimes \mathbb{E}[S_2](t).
\]

This unfortunate result is due to the fact that the expectation is a linear operation in classical algebra but not in min-plus algebra. Hence, the order of operations cannot be exchanged and the approach does not solve the problem of stochastic network service curves by itself.

Regarding envelopes and service curves as random functions that are modeled by their moment generating function, respectively, Laplace transform Chang proposes an approach for end-to-end convolution [17] that is elaborated by Fidler [157]. The idea goes back to earlier work by Chang [155] where it is observed that the exponential transformation allows bounding max-plus expressions in classical algebra. Brought forward to the min-plus network calculus it follows that the exponential transformation of the min-plus convolution is bounded by a classical convolution. This change of algebra is due to the fact that the exponential takes the addition to a multiplication and the infimum is bounded by a sum using
Boole’s inequality. It permits inverting the order of expectation and convolution since both are linear operations in classical algebra. Hence, the end-to-end convolution of bivariate service curves can (in case of statistical independence) be written as

\[ E[e^{-\theta(S_1 \otimes S_2)}] \leq E[e^{-\theta S_1} \cdot e^{-\theta S_2}] = E[e^{-\theta S_1}] \cdot E[e^{-\theta S_2}]. \]

Using the earlier notation for moment generating functions we have in short form that \( M_{S_1 \otimes S_2} \leq M_{S_1} \cdot M_{S_2}. \) Note that due to the minus sign in the exponent the upper bound shown here provides in fact a lower bound for the end-to-end service curve.

Fidler [157] uses this approach to derive closed-form end-to-end performance bounds for line topologies with single-hop persistent \((\sigma(\theta), \rho(\theta))\) cross-traffic as shown in Fig. 8. The finding in [157] are statistical delay bounds that scale in \( O(n) \). Compared to the result in \( O(n log n) \) [65] the improvement is due to the efficient utilization of statistical independence if service curves are defined as random processes and modeled by their moment generating function. In contrast service curves as non-random functions dispense with this assumption. Consequently, they are also applicable if the service, respectively, cross-traffic at different systems is correlated. In this case \( \Theta(n log n) \) is shown to be an upper as well as a lower bound in [207]. For further details on the scaling properties in the stochastic network calculus see the recent dissertation by Ciucu [66].

D. Open research challenges

Remarkable progress on traffic envelopes and stochastic service curves using sample path bounds has led to the recent formulation of the stochastic network calculus. The potential of the approach is documented clearly by closed-form solutions and explicit results on the scaling of end-to-end statistical performance bounds for tandem systems with random cross-traffic. Yet, the stochastic network calculus turned out to be significantly more difficult than its deterministic counterpart and numerous deterministic results have yet to be mirrored in the probabilistic framework. While the use of stochastic service curves has largely been motivated by the analysis of schedulers with random cross-traffic, the concept is quite universal and can be applied to many other random systems, as demonstrated by first results for noisy radio channels and multiple access protocols. Among the most challenging systems are feedback loops, e.g. window-based flow control and congestion control as implemented by TCP, see Sect. II-E3, where stochastic models could significantly benefit the design of future protocols. Another important issue is the analysis of networks beyond line or tree topologies, see Sect. V. Stochastic results that resemble e.g. the topology-agnostic bounds in Sect. V-C2 are highly desirable as they could shed light on actual network performance and stability. Similar to simple product-form queuing networks with Poisson traffic, “convolution-form networks” [66] in the stochastic network calculus could provide bounds for many other relevant traffic types. While results for a number of traffic models are available today difficult problems are still posed by the analysis of certain unfriendly traffic types in particular by heavy-tailed traffic. A discussion of relevant open research questions can also be found in [66], [18].

IX. CONCLUDING REMARKS

We provided a survey on deterministic and stochastic service curve models and their applications in the network calculus. We covered important network calculus basics and elaborated on the view of network calculus as a systems theory under the min-plus algebra. This analogy has proven very powerful since among other properties it implies the concatenation theorem for tandem systems.

We reviewed elementary models for different scheduling disciplines and discussed unifying abstractions and their equivalent max-plus and min-plus representations. These models have significant potential and led to numerous important applications, such as in scheduler design, network configuration, and quality of service. Based on the concept of a scheduler’s leftover service curve we discussed the application of network calculus to different classes of topologies in detail. This area still offers various open challenges, yet, significant progress has been made during the past years. Tight bounds are known today for sink-tree networks. For general possibly non-feed-forward topologies a number of significant results are available.

We reported recent measurement techniques for service curve estimation and identification. These methods complement the previous theory as they can be used for experimental verification of service curve models and act as a benchmark. Moreover, the measurement-based approach can provide service curve estimates for systems that are not solved today.

We elaborated on time-varying and in particular stochastic systems, which received much attention during recent years. Although challenging, the stochastic network calculus shows tremendous potential. Significant progress towards a comprehensive and capable theory has been made lately that can provide important insights, such as the current scaling results for tandem systems.

REFERENCES


Markus Fidler (M’04, SM’08) graduated in electrical engineering from RWTH Aachen University in 1997 and in business economics from University of Hagen in 2001. During the years 1997 until 2000 he was a GSM/GPRS systems engineer at Hagemuk Telecom and at Alcatel SEL, respectively. In 2001 he joined RWTH Aachen University where he received his doctoral degree in computer engineering with distinction beginning of 2004. He was a postdoctoral fellow of the institute Mittag-Leffler in Stockholm in 2004, INN University in 2005, and the University of Toronto in 2006. During the years 2007 and 2008 he was an Emmy Noether research group leader at the Multimedia Communications Lab, TU Darmstadt, where he received his habilitation degree in communications networks end of 2008. Since 2009 he is a full professor heading the Communications Networks Group at the Institute of Communications Technology, Leibniz Universität Hannover.