Of Kernels and Queues: when network calculus meets analytic combinatorics

Anne Bouillard*, Céline Comte*†, Élie de Panafieu*, Fabien Mathieu*
*Nokia Bell Labs, France
†Télécom ParisTech, Paris-Saclay University, France

***Abstract*—Stochastic network calculus is a tool for computing error bounds on the performance of queueing systems. However, deriving accurate bounds for networks consisting of several queues or subject to non-independent traffic inputs is challenging. In this paper, we investigate the relevance of the tools from analytic combinatorics, especially the *kernel method*, to tackle this problem. Applying the kernel method allows us to compute the generating functions of the queue state distributions in the stationary regime of the network. As a consequence, error bounds with an arbitrary precision can be computed. In this preliminary work, we focus on simple examples which are representative of the difficulties that the kernel method allows us to overcome.**

I. Introduction

The development of new wireless communication technologies (5G) shed a new light on queueing theory, as the strong requirements on buffer occupancy, latencies, and reliability, bring the need for accurate dimensioning rules. In many scenarios, data packets arrive by batches and are processed by a server that can deal with a fixed number of packets per time slot [1]. The $G/D/1$ queue is thus a natural model.

A powerful tool to analyze such queues is Stochastic Network Calculus (SNC) [2]. The aim of SNC is to derive precise error bounds on the performance of systems, combining deterministic network calculus and probabilistic tools.

Among the techniques developed so far, the Tailbounded approach [3] introduces a violation probability in the parameters of the deterministic setting. It makes possible the computation of error bounds in networks, like in [4], but these bounds are usually loose. A second technique, introduced in [5], relies on moment generating functions. It can be very accurate for one queue. For example, in [6], [7], the authors obtain tight upper and lower bounds for the single-server case under various service policies and arrival processes, using martingales and Doob's inequality. However, for more general topologies, the method becomes non applicable due to interdependencies between the processes. Recently, some (looser) bounds have been computed using Hölder's inequality [8], [9].

The use of generating functions to investigate random processes is the core principle of *analytic combinatorics*, a subfield of combinatorics (see [10]). This community developed mathematical tools to study random walks [11], such as the *kernel method* [12], [13], described later. The link between random walks and queueing theory is known and results on the former were transferred to the latter [14].

In this article, we show how generating functions and the kernel method can be applied to derive precise results on queueing systems.

In §II, we first recall the main definitions and notations of generating functions. The main contribution of the paper is given in §III, where we show in detail how to apply the kernel method to study the $GI/D/1$ queue. Although the result itself is well-known (we retrieve the Pollaczek-Khinchine formula), the interest of the analysis is that it contains all the pieces for further extensions, such as several flows of packets, several queues, or non i.i.d. arrivals. Some of these extensions are developed in §IV: random service, multi-flow and multi-queue. Finally, we confront our results with simulations in §V.

II. Generating functions

In this section, we recall some basics of generating functions. Let $(a_n)_{n\geq 0}$ be a sequence of non-negative numbers. Its generating function is the formal series

$$A(u) = \sum_{n\geq 0} a_n u^n.$$

The n-th monomial a_n will also be denoted by $[u^n]A(u)$. In combinatorics, a_n is often the number of objects of size n within a given family. In probability, a_n is usually the probability that a random variable $\mathbf{A}$ with values in $\mathbb{N}$ is equal to n:

$$A(u) = \sum_{n\geq 0} \mathbb{P}(\mathbf{A} = n)u^n.$$

In that case, we write $\mathbf{A} \sim A$; the convergence radius ρ of the function A is at least 1, $A(1) = 1$, $A'(1) = \mathbb{E}[\mathbf{A}]$, and we assume $\lim_{u\to\rho} A(u) = +\infty$ to simplify the asymptotic analyses.

Two elementary operations can be performed on generating functions. Suppose that A and B are the generating functions of two random variables $\mathbf{A}$ and $\mathbf{B}$, respectively.

1) If the events $\{\mathbf{A} = n\}$ and $\{\mathbf{B} = n\}$ are disjoint for each $n \in \mathbb{N}$, then
$A(u) + B(u) = \sum_{n\geq 0} \mathbb{P}(\{\mathbf{A} = n\} \cup \{\mathbf{B} = n\})u^n$.
2) If $\mathbf{A}$ and $\mathbf{B}$ are independent, then $A(u)B(u)$ is the generating function of the r.v. $\mathbf{A} + \mathbf{B}$.

Consider the example of a Galton-Watson tree, which is a branching process where the number of children of each node

978-0-9883045-5-0/18/$31.00 ©2018 ITC
DOI 10.1109/ITC30.2018.10061

is i.i.d. with distribution given by the generating function A. The number of nodes of the tree is

$$\mathbf{X} = 1 + \sum_{k=1}^{\mathbf{A}} \mathbf{X}_k,$$

where $\mathbf{A} \sim A$ is the number of children of the root and $\mathbf{X}_k$ is the number of nodes in the subtree rooted at the k-th child of the root. $\mathbf{X}_k$ has the same distribution as $\mathbf{X}$, hence the same generating function, denoted by T_A. Therefore, we obtain

$$T_A(u) = uA(T_A(u)). \tag{1}$$

This equation characterizes T_A. Fig. 1 shows how $T_A(u)$ is computed. Being the generating function of a probability distribution, $T_A(u)$ must be a solution of Eq. (1) that is analytic at 0, also known as a *small root* of the equation. $T_A(u)$ is then the abscissa coordinate of the first intersection of $A(x)$ with the line x/u. There is a maximal value $\rho_{T_A} > 1$ of u for which a root exists.

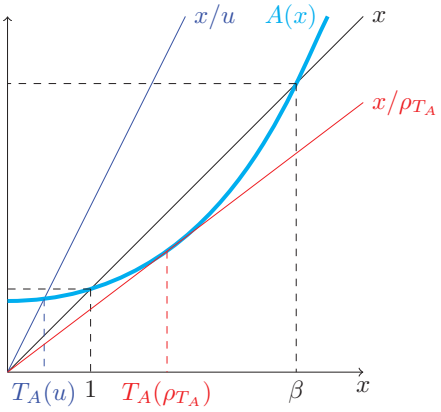


Fig. 1. Equation $T_A(u) = uA(T_A(u))$.

By deriving both sides at $u = 1$, one gets that $\mathbb{E}[\mathbf{X}] = 1 + \mathbb{E}[\mathbf{X}]\mathbb{E}[\mathbf{A}]$, so that $\mathbb{E}[\mathbf{X}] = (1 - \mathbb{E}[\mathbf{A}])^{-1}$ if $\mathbb{E}[\mathbf{A}] < 1$.

Adopting a combinatorics viewpoint allows us to consider generating functions that do not represent a distribution. For example, T_A^k is the generating function of the distribution of the total size of k independent Galton-Watson trees, and $\frac{1}{1-T_A} = \sum_{k\geq 0} T_A^k$ is the sum of the distributions for all possible k. Let $(\mathbf{X}_j)_{j\geq 0}$ denote the sequence of the sizes of i.i.d. Galton-Watson trees, then

$$[u^n]\frac{1}{1 - T_A(u)} = \mathbb{P}(\exists k,\ \mathbf{X}_1 + \cdots + \mathbf{X}_k = n).$$

Studying the behavior of this series will prove useful to derive the asymptotic probability that an arbitrary number of trees has a given total size. As $T_A(u) < 1$ for all $0 \leq u < 1$ and $T_A(1) = 1$, we can apply the result of [10, p. 294, Th. V.1]:

$$[u^n]\frac{1}{1 - T_A(u)} \underset{n\to\infty}{\sim} \frac{1}{T_A'(1)} = 1 - \mathbb{E}[\mathbf{A}]. \tag{2}$$

All the definitions can be extended to the multivariate case.

III. The single-server queue

In this section, we present the simple example of a single-server queue with one flow of packets, as depicted in Fig. 2. The results presented here are not new (we eventually re-discover the Pollaczek-Khinchine formula, and apply tools developed by [12]), but our aim is to present the method that will be generalized later.

A. Queueing model

The queue is initially empty. At each time slot $t \geq 1$, one packet (if any) is served and then $\mathbf{A}_t$ packets arrive. The sequence $(\mathbf{A}_t)_{t\geq 1}$ is i.i.d. with generating function A and mean $\lambda < 1$. We let $\mathbf{X}_t$ denote the number of packets in the queue at the end of time slot t. The system is driven by the equations

$$\mathbf{X}_0 = 0 \text{ and } \mathbf{X}_{t+1} = (\mathbf{X}_t - 1)_+ + \mathbf{A}_{t+1}, \quad \forall t \geq 0, \tag{3}$$

where $(\cdot)_+ = \max(\cdot, 0)$.

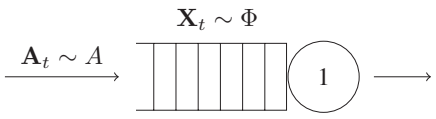


Fig. 2. A single-server queue crossed by a single flow. The server processes one packet during each time slot.

B. Generating function

We define the generating function of the queue state as

$$\Phi(u, z) = \sum_{n\geq 0}\sum_{t\geq 0} \mathbb{P}(\mathbf{X}_t = n)u^n z^t. \tag{4}$$

For each $t \in \mathbb{N}$, taking the coefficient of z^t yields

$$[z^t]\,\Phi(u, z) = \sum_{n\geq 0} \mathbb{P}(\mathbf{X}_t = n)u^n,$$

which is the generating function of $\mathbf{X}_t$. We will show that

Lemma 1:

$$\Phi(u, z) = 1 + zA(u)\left[(\Phi(u, z) - \Phi(0, z))u^{-1} + \Phi(0, z)\right]. \tag{5}$$

Sketch of proof: Eq. (3) implies $\mathbf{X}_0 = 0$ and for any $t \geq 1$, with the notation $a_n = \mathbb{P}(\mathbf{A} = n)$,

$$\mathbb{P}(\mathbf{X}_t = n - 1) = a_{n-1}(\mathbb{P}(\mathbf{X}_{t-1} = 0) + \mathbb{P}(\mathbf{X}_{t-1} = 1)) + \sum_{m=2}^{n} a_{n-m}\mathbb{P}(\mathbf{X}_{t-1} = m).$$

Multiplying this relation by $u^n z^t$, summing over n, t and dividing both sides by u leads to the equation of the lemma. The formula can also be directly derived using the *Symbolic Method* [10]. ∎

Eq. (5) can be rewritten as

$$\Phi(u, z)\left[1 - zA(u)u^{-1}\right] = 1 - \Phi(0, z)zA(u)\left[u^{-1} - 1\right]. \tag{6}$$

Although Eq. (6) completely characterizes $\Phi(u, z)$, it is not straightforward to derive an explicit formula for $\Phi(u, z)$ from it, as we would need an expression for $\Phi(0, z)$. This expression will be obtained with the kernel method.

C. Kernel method

When the left-hand side of Eq. (6) cancels, so does the right-hand side. The kernel method [12], [13] consists in taking $u = U(z)$ such that the second factor of the left-hand side cancels. Here, $U(z)$ is implicitly defined by the equality $U(z) = zA(U(z))$, and we recognize from Eq. (1) the size distribution of a Galton-Watson tree where the offspring distribution has the generating function A. Therefore, we have $U = T_A$.

Injecting $T_A(z)$ in Eq. (6) cancels its left-hand side, and its right-hand side can be rewritten as

$$\Phi(0, z) = \frac{1}{1 - T_A(z)}.$$

Going back to Eq. (6), we finally obtain

$$\Phi(u, z) = \frac{1 + \frac{1}{1-T_A(z)} zA(u)\left(1 - u^{-1}\right)}{1 - zA(u)u^{-1}}. \tag{7}$$

The kernel method has the following interpretation in terms of the queue sample paths. The generating function $\Phi(0, z) = \sum_{t\geq 0} \mathbb{P}(\mathbf{X}_t = 0)z^t$ is associated with the probability of having an empty queue. Consider the duration between two consecutive instants when the queue is empty, which we call an *inter-empty period*. It was showed in [15] that we can build a Galton-Watson tree with offspring distribution A from an inter-empty period: each node represents a time slot; its children are the time slots when the packets arrived during this time slot are served. Having an empty queue at time t means that the realization between times 0 and t is made up of an arbitrary number of inter-empty periods. This corresponds exactly to $\frac{1}{1-T_A(z)}$, where T_A is as defined in Eq. (1).

D. Asymptotic performance

In this paragraph, our aim is to bound the probability that $\mathbf{X}_t$ exceeds some value R in stationary regime. Note that, by monotony, this will also be an upper bound for the initially empty queue. We proceed in two steps. We first compute Π, the generating function of the stationary distribution of $(\mathbf{X}_t)$, and then we derive the asymptotic behavior of Π.

a) Computation of Π*:* We know that, under the stability condition $A'(1) = \lambda < 1$, the distribution of $\mathbf{X}_t$ converges to a stationary distribution π as t tends to $+\infty$. The first step of our analysis consists in finding the generating function Π of this distribution π. Recall that, for each $t \in \mathbb{N}$, the generating function of $\mathbf{X}_t$ is $\Pi_t(u) = [z^t]\Phi(u, z)$. By [10, p. 624], it suffices to study the limit of $\Pi_t(u)$ as t tends to $+\infty$, when u is fixed. The obtained limit is exactly $\Pi(u)$.

Let us fix $u = u_0$. We see in Eq. (7) that $\Phi(u_0, z)$ has two potential poles, 1 and $u_0/A(u_0)$. It can be checked that $T_A(\frac{u_0}{A(u_0)}) = u_0$, so that $u_0/A(u_0)$ is actually not a pole. In order to derive the asymptotic behavior of $\Pi_t(u_0)$ as t tends to $+\infty$, we first compute a simpler equivalent of $\Phi(u_0, z)$ in the neighborhood of its pole $z = 1$. After some rewriting, we obtain

$$\Phi(u_0, z) = \frac{u_0}{u_0 - zA(u_0)} + \frac{1}{1 - T_A(z)} \frac{zA(u_0)(u_0 - 1)}{u_0 - zA(u_0)}.$$

As a consequence,

$$\Phi(u_0, z) \underset{z\to 1}{\sim} \frac{u_0}{u_0 - A(u_0)} + \frac{1}{1 - T_A(z)} \frac{A(u_0)(u_0 - 1)}{u_0 - A(u_0)},$$

and from Eq. (2), the terms are equivalent to

$$[z^t]\Phi(u_0, z) \underset{t\to\infty}{\sim} (1 - \lambda) \frac{A(u_0)(u_0 - 1)}{u_0 - A(u_0)}.$$

Therefore, the generating function of π is equal to the one given by the Pollaczek-Khinchine formula

$$\Pi(u) = (1 - \lambda) \frac{A(u)(u - 1)}{u - A(u)}.$$

b) Performance: The second solution β of the equation $u = A(u)$ is the convergence radius of the function Π (with $\beta = +\infty$ in the degenerate case where $A(u)$ is linear). The error bound, *i.e.* the probability that the buffer occupancy is at least R, is $\sum_{n\geq R} \pi(n)$. Its generating function is

$$E(u) = \sum_{R\geq 0} \left(\sum_{n\geq R} \pi(n) \right) u^R = \frac{1 - u\Pi(u)}{1 - u}. \tag{8}$$

The asymptotic analysis of this generating function yields

Theorem 1: With $\mathbf{X} \sim \Pi$,

$$\mathbb{P}(\mathbf{X} \geq R) \underset{R\to\infty}{\sim} (1 - \lambda) \frac{\beta}{A'(\beta) - 1} \beta^{-R}. \tag{9}$$

IV. Extensions of the single-server queue

The analysis in the previous section shows that deriving an equation satisfied by the generating function from the system dynamics is the easy step; solving this equation is harder. We now consider a few simple extensions of the model of §III-A, where the kernel method allows us to perform the analysis and derive explicit formulas for the performance metrics.

A. Random service

We consider a first extension of the model of §III-A where the service is random. Specifically, at each time slot $t \geq 1$, the server processes one packet (if any) with some probability $p > \lambda$, and zero packet otherwise. The system is driven by the equations

$$\mathbf{X}_0 = 0 \text{ and } \mathbf{X}_{t+1} = (\mathbf{X}_t - \mathbf{S}_t)_+ + \mathbf{A}_{t+1}, \quad \forall t \geq 0,$$

where $(\mathbf{S}_t)_{t\in\mathbb{N}}$ is a sequence of independent, Bernoulli distributed random variables with parameter p. The corresponding generating function is $S(u) = 1 - p + pu$.

The generating function Φ of the system state is as defined in (4). The equation satisfied by Φ is a rewriting of Eq. (6), where u^{-1} is replaced by $S(u^{-1})$:

$$\Phi(u, z)[1 - zA(u)S(u^{-1})] = 1 - \Phi(0, z)zA(u)[S(u^{-1}) - 1].$$

Applying the kernel method consists in choosing $u = U(z)$ such that $zA(U(z))S(U(z)^{-1}) = 1$. We can rewrite this as $U(z) = G(zA(U(z)))$, where G is the generating series of the geometric distribution (defined on the set of positive integers) with parameter p:

$$G(s) = \frac{ps}{1 - (1 - p)s}.$$

In much the same way as in §III-C, we obtain

$$\Phi(0,z) = \frac{1}{1 - zA(U(z))} = \frac{1}{1 - T_{A\circ G}(z)}.$$

The second equality holds because $\bar{U}(z) = zA(U(z))$ satisfies the equation $\bar{U}(z) = zA(G(\bar{U}(z)))$, so that $\bar{U}$ is also the generating function of the size of a Galton-Watson tree with offspring distribution $A \circ G$. Finally, we obtain

$$\Phi(u,z) = \frac{1 + \frac{1}{1-T_{A\circ G}(z)} zA(u)(1 - S(u^{-1}))}{1 - zA(u)S(u^{-1})}.$$

The interpretation is similar to that of §III-C, except that the number of time slots dedicated to a given customer is now geometrically distributed with parameter p.

The stationary distribution has the generating function

$$\Pi(u) = \left(1 - \frac{\lambda}{p}\right) \frac{A(u)\left(1 - S(u^{-1})\right)}{1 - A(u)S(u^{-1})}.$$

Let γ be the largest solution of the equation $A(u)S(u^{-1}) = 1$, or, equivalently, $u = G(A(u))$. Similarly to §III-D, we obtain

Theorem 2: For $\mathbf{X} \sim \Pi$,

$$\mathbb{P}(\mathbf{X} \geq R) \underset{R\to\infty}{\sim} \frac{(1 - \frac{\lambda}{p})(A(\gamma) - 1)}{(\gamma - 1)\big(A'(\gamma)S(\gamma^{-1}) - A(\gamma)p\gamma^{-2}\big)} \gamma^{-R}.$$

The kernel method is also applicable to the case where the server processes up to c packets at each time slot, for some integer $c \geq 1$. Assume that the distribution of the number of served packets has generating function $S(u)$ if the queue contains at least c packets, and $S_k(u)$ if the queue contains exactly k packets, for each $0 \leq k < c$. The equivalent of Eq. (6) is now

$$\Phi(u,z)[1 - zA(u)S(u^{-1})] = 1 - \sum_{k=0}^{c-1} zA(u)(S(u^{-1}) - S_k(u^{-1})) \frac{u^k}{k!} \frac{\partial^k}{(\partial u)^k} \Phi(0,z). \quad (10)$$

We refer the reader to [10, p. 508] or [12] for a detailed analysis of this equation, and provide here a short version. There are c independent functions $(U_k(z))_{0\leq k<c}$, analytic at 0, that cancel the second term of the left hand-side, because S is a degree c polynomial. Thus, we obtain c equations for the c unknowns $\frac{\partial^k}{(\partial u)^k}\Phi(0,z)$ for $0 \leq k < c$. Solving this system of equations and injecting the solution in Eq. (10) leads to the expression of $\Phi(u,z)$.

B. Several flows with priorities

Consider the system in Fig. 3. As in §III-A, the queue is initially empty and the server processes one packet at each time slot. There are two flows of packets. Flow 1 has priority over flow 2, so that a packet from flow 1 is served whenever the queue contains at least one packet from this flow at the beginning of this time slot. At each time slot $t \geq 1$, $\mathbf{A}_t$ packets from flow 1 and $\mathbf{B}_t$ packets from flow 2 arrive. The sequences $(\mathbf{A}_t)_{t\geq 1}$ and $(\mathbf{B}_t)_{t\geq 1}$ are independent and i.i.d. with generating function A and B and mean λ_A and λ_B, respectively, such that $\lambda_A + \lambda_B < 1$. We denote by $\mathbf{X}_t$ and $\mathbf{Y}_t$ the respective numbers of packets from flows 1 and 2 in the queue at the end of time slot t. The system is then driven by the equations $\mathbf{X}_0 = 0$, $\mathbf{Y}_0 = 0$ and

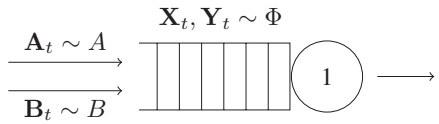


Fig. 3. A single-server queue crossed by two flows.

$$\begin{cases} \mathbf{X}_{t+1} = (\mathbf{X}_t - 1)_+ + \mathbf{A}_{t+1}, \\ \mathbf{Y}_{t+1} = (\mathbf{Y}_t - 1_{\{\mathbf{X}_t=0\}})_+ + \mathbf{B}_{t+1}, \quad \forall t \geq 0. \end{cases} \quad (11)$$

We define a generating function for the state $(\mathbf{X}_t, \mathbf{Y}_t)_{t\geq 0}$, with three variables u, v, and z, respectively representing the numbers of packets from flows 1 and 2 and the time:

$$\Phi(u,v,z) = \sum_{n\geq 0}\sum_{m\geq 0}\sum_{t\geq 0} \mathbb{P}(\mathbf{X}_t = n, \mathbf{Y}_t = m) u^n v^m z^t.$$

The equation satisfied by Φ follows from Eq. (11):

$$\Phi(u,v,z) = 1 + zA(u)B(v)[(\Phi(u,v,z) - \Phi(0,v,z))u^{-1} + (\Phi(0,v,z) - \Phi(0,0,z))v^{-1} + \Phi(0,0,z)].$$

Here $(\Phi(u,v,z) - \Phi(0,v,z))u^{-1}$ represents the service of a packet from flow 1 (if any) and $(\Phi(0,v,z) - \Phi(0,0,z))v^{-1}$ the service of a packet from flow 2 (if any, and when there is no packet from flow 1). This equation can be rewritten as

$$\Phi(u,v,z)\left[1 - zA(u)B(v)u^{-1}\right] = 1 - zA(u)B(v)\times \left[\Phi(0,v,z)\left(u^{-1} - v^{-1}\right) + \Phi(0,0,z)\left(v^{-1} - 1\right)\right]. \quad (12)$$

We could find an expression for $\Phi(0,0,z)$ by applying twice the kernel method on this equation, but we prefer a more intuitive approach. The generating function $\Phi(0,0,z)$ is associated with the probability that the queue is empty. This probability only depends on the global arrival process $(\mathbf{A}_t + \mathbf{B}_t)_{t\geq 1}$, regardless of the division of packets into flows. Therefore, we know from §III-C that

$$\Phi(0,0,z) = \frac{1}{1 - T_{AB}(z)},$$

where T_{AB} is the generating function of the size of a Galton-Watson tree with offspring distribution AB.

We apply the kernel method to derive the expression of $\Phi(0,v,z)$. Let us take $u = U(v,z)$ such that $U(v,z) = zB(v)A(U(v,z))$, in order to cancel the left-hand side of Eq. (12). We obtain $U(v,z) = T_A(zB(v))$, which is again strongly related to a Galton-Watson tree. The interpretation is similar to that of §III-C, except that $U(v,z)$ is now the generating function of the number of time slots passed and flow-2 packets arrived during an inter-empty period of flow 1. The priority of flow 1 ensures that no packet from flow 2 is served in the meantime. After simplifications, we get

$$\Phi(0,v,z) = \frac{v + \frac{1}{1-T_{AB}(z)} T_A(zB(v))(v-1)}{v - T_A(zB(v))},$$

and the expression for $\Phi(u,v,z)$ immediately follows. It is not difficult to see that this method can be generalized to queues with more than two flows with a total order on priority levels.

Suppose that we focus on the number of packets from flow 2 in the stationary state. We are then interested in the generating function $\Phi(1,v,z)$. The same approach as in §III-D can be used to obtain the following result.

Theorem 3: The limit distribution of the number of packets of flow 2 is given by the generating series

$$\Pi(v) = (1-\lambda_A-\lambda_B)\frac{B(v)(1-v)(T_A(B(v))-1)}{(1-B(v))(v-T_A(B(v))}.$$

Let δ be the largest solution of the equation $v = T_A(B(v))$ (as T_A and B are convex, there are exactly 2 solutions, and the smallest is 1). Similarly to §III-D,

Theorem 4: For $\mathbf{Y} \sim \Pi$,

$$\mathbb{P}(\mathbf{Y} \geq R) \underset{R\to\infty}{\sim} \frac{(1-\lambda_A-\lambda_B)B(\delta)(\delta-1)}{(1-B(\delta))(1-(T_A\circ B)'(\delta))}\delta^{-R}. \quad (13)$$

C. Several queues

We can also use generating functions to describe the dynamics of networks of queues. As an example, consider the network of Fig. 4, consisting of two single-server queues. At each time slot $t \geq 1$, $\mathbf{A}_t$ packets arrive at queue 1 and $\mathbf{B}_t$ packets arrive at queue 2. As before, the sequences $(\mathbf{A}_t)_{t\geq 1}$ and $(\mathbf{B}_t)_{t\geq 1}$ are independent and i.i.d. with generating function A and B and mean λ_A and λ_B, respectively, such that $\lambda_A + \lambda_B < 1$. Additionally, the packets served at queue 1 are subsequently forwarded to queue 2 for service. We let $\mathbf{X}_t$ and $\mathbf{Y}_t$ denote the numbers of packets at queues 1 and 2, respectively, at time t.

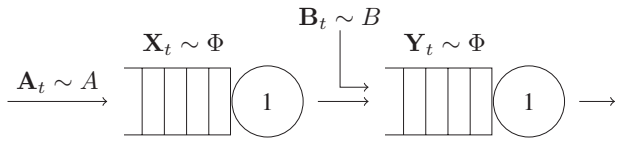


Fig. 4. A network of two queues crossed by two flows.

The dynamics of the system, which is initially empty, are driven by the equations $\mathbf{X}_0 = 0$, $\mathbf{Y}_0 = 0$ and

$$\begin{cases} \mathbf{X}_{t+1} &= (\mathbf{X}_t - 1)_+ + \mathbf{A}_{t+1}, \\ \mathbf{Y}_{t+1} &= (\mathbf{Y}_t - 1)_+ + \mathbf{B}_{t+1} + 1_{\{\mathbf{X}_t>0\}}, \quad \forall t \geq 0. \end{cases} \quad (14)$$

We define a generating function for the state $(\mathbf{X}_t, \mathbf{Y}_t)_{t\geq 0}$, with three variables u, v, and z, respectively representing the numbers of packets at queues 1 and 2 and the time:

$$\Phi(u,v,z) = \sum_{n\geq 0}\sum_{m\geq 0}\sum_{t\geq 0}\mathbb{P}(\mathbf{X}_t = n, \mathbf{Y}_t = m)u^n v^m z^t.$$

This function Φ satisfies the following equation:

$$\begin{aligned}\Phi(u,v,z)[1-zA(u)B(v)u^{-1}] &= 1 - zA(u)B(v) \\ &[\Phi(u,0,z)(u^{-1}-vu^{-1}) + \Phi(0,v,z)(u^{-1}-v^{-1}) \\ &+ \Phi(0,0,z)(vu^{-1}+v^{-1}-u^{-1}-1)]. \quad (15)\end{aligned}$$

The kernel method cannot be applied directly. Indeed, we need to compute three generating functions ($\Phi(u,0,z)$, $\Phi(0,v,z)$, and $\Phi(0,0,z)$), while we can only apply the kernel method (at most) twice. It is, however, possible to find an additional relation between these functions:

$$\Phi(u,0,z) = 1 + zA(u)B(0)\big[\Phi(0,0,z) + [v^1]\Phi(0,v,z)\big].$$

This relation can be obtained in two different ways. We can derive Eq. (15) according to v at $v = 0$. Alternatively, we can go back to the system dynamics: queue 2 is empty at the end of some time slot $t \geq 1$ if it does not receive any external arrival during this time slot and, at the end of time slot $t-1$, queue 1 was empty and queue 2 contained at most one packet.

This equation gives the relation

$$\Phi(u,0,z) = 1 + \frac{A(u)}{A(0)}[\Phi(0,0,z) - 1].$$

We are now in a position to apply the kernel method, by defining first $U(v,z) = zA(U(v,z))B(v)$ and then $V(z) = zA(V(z))B(V(z))$. The generating function Π of the stationary distribution of the number of packets in queue 2 can be computed similarly to the previous cases. For simplicity, we only give its asymptotic behavior:

Theorem 5: With δ previously defined, for $\mathbf{Y} \sim \Pi$,

$$\mathbb{P}(\mathbf{Y} \geq R) \underset{R\to\infty}{\sim} \frac{(1-\lambda_A-\lambda_B)\delta(\delta-1)}{(1-B(\delta))(1-(T_A\circ B)'(\delta))}\delta^{-R}. \quad (16)$$

D. Non-independent arrivals

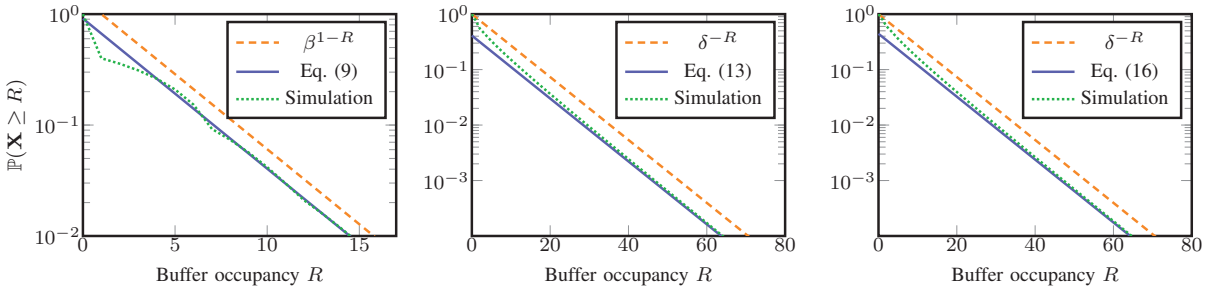
The analysis can be extended to networks with more generic arrival processes. We take the network of Fig. 3 as an example.

- The arrivals of flows 1 and 2 may be dependent. The global arrival process is then described by a generating function $A(u,v)$ that cannot be written as a product $A(u)B(v)$.
- Within each flow, the numbers of arrivals at different time slots may not be i.i.d. anymore. Instead, they may be described by a modulated process (which includes modulated Markov On-Off processes). The modulation is described by a finite Markov chain. The system dynamics are then described by a system of equations on generating functions (one per state of the Markov chain).

V. Numerical evaluation

We tested our formulas against simulations in three different scenarios: the single-server case of §III, the multiflow single-server case of §IV-B and the tandem network of two single-server queues of §IV-C. The service is deterministic.

Performing simulations consists in computing the stationary distribution of the truncated processes (the number of packets in each queue never exceeds 200) whose dynamics are described by Eqs. (3), (11) or (14). The approximation of the stationary distribution is obtained by iteratively computing the distribution after t steps for a large enough t (the stopping criterion is when the distance in total variation between the t-th and the $t+1$-th distribution is less than 10^{-12}).



(a) Single queue (b) Two flows with priorities (c) Two servers

Fig. 5. Numerical evaluation of the kernel method. Parameters $A = D_{2/30,6}$ and $B = D_{2/5,1}$.

Each arrival process has a bimodal distribution with generating function $D_{p,M}(u) = (1-p) + pu^M$, for some p and M: at each time slot, either M packets arrive, which occurs with probability p, or no packet arrives. With this distribution, the arrival rate is $D'_{p,M}(1) = Mp$. In the numerical results of Fig. 5, we take $A = D_{2/30,6}$ and $B = D_{2/5,1}$. This choice of the functions A and B is arbitrary (other distributions lead to similar observations)

Fig. 5a illustrates the case of a single queue. The curve β^{1-R} is the one that would be obtained using Doob's inequality from [6] (we do not use β^{-R} because we consider the size of the queue before service and not after as in [6]). The simulation confirms that we obtain the exact asymptotic behavior, and shows that we improve Doob's inequality by a factor 1.5. We remark that the simulation curve has some irregularities for small values of R. This is explained by the arrival of packets in batches of 6. It is possible to compute the exact error bound from Eq. (8) by deriving the first terms explicitly: $[u^R]E(u) = \frac{1}{R!}\frac{d^R}{(du)^R}E(u)\big|_{u=0}$.

Fig. 5b illustrates the case of a single-server queue with two flows. We focus on the buffer occupancy of flow 2. Since flow 1 has priority, its buffer occupancy is still given by Fig. 5a. Again, the simulation validates our theoretical results. Up to our knowledge, there is no formula similar to Doob's inequality, so the curve δ^{-R} only mimics a *Doob-like inequality*.

Fig. 5c illustrates the error bound in the second queue of the tandem network of §IV-C. The error bound differs only from the previous case by a constant factor.

Although we obtain the exact asymptotic in those two cases, it seems that these are lower bounds of the error. Indeed, we only computed the first term. But, once again, as we were able to compute an exact formula for the error bounds, more terms are derivable using Taylor expansions.

VI. Conclusion

In this paper, we have demonstrated on simple examples that methods from analytic combinatorics can be successfully applied to the analysis of queueing systems. We have focused on computing backlog bounds, but we believe delay bounds can be derived by using the same techniques as in [6], for the FIFO, EDF (earliest-deadline-first) and priorities policies. Moreover, combining the computations described in §IV would allow other service policies to enter our framework, in particular some discrete version of GPS (generalized processor sharing). Following the approach of [12], we could also consider a continuous-time extension of our work based on Laplace transforms. The greatest challenge is to cope with networks of queues. A simple example with two queues has been analyzed. The same analysis can be extended for more than two queues, but this analysis is still partial since the packets are aggregated at each queue. Further investigation needs to be done, in particular in view of the techniques presented in [13].

References

[1] P. Sehier, A. Bouillard, F. Mathieu, and T. Deiß, "Transport Network Design for FrontHaul," in *3rd IEEE Workshop on Next Generation Backhaul/Fronthaul Networks* , Toronto, Canada, Sep. 2017.

[2] M. Fidler and A. Rizk, "A guide to the stochastic network calculus," *IEEE Commun. Surveys and Tutorials*, vol. 17, no. 1, pp. 92–105, 2015.

[3] Y. Jiang and L. Yong, *Stochastic Network Calculus*. Springer, 2008.

[4] F. Ciucu, A. Burchard, and J. Liebeherr, "Scaling properties of statistical end-to-end bounds in the network calculus," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2300–2312, 2006.

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

[6] F. Poloczek and F. Ciucu, "Scheduling analysis with martingales," *Perform. Eval.*, vol. 79, pp. 56–72, 2014.

[7] F. Ciucu and F. Poloczek, "On multiplexing flows: Does it hurt or not?" in *IEEE Conf. on Comput. Commun., INFOCOM*, 2015, pp. 1122–1130.

[8] M. A. Beck, "Advances in theory and applicability of stochastic network calculus,," PhD thesis, University of Kaiserslautern, 2016.

[9] P. Nikolaus and J. B. Schmitt, "On per-flow delay bounds in tandem queues under (in)dependent arrivals," in *IFIP Networking Conference*, 2017, pp. 1–9.

[10] P. Flajolet and R. Sedgewick, *Analytic Combinatorics*, 1st ed. Cambridge University Press, 2009.

[11] P. Flajolet and F. Guillemin, "The formal theory of birth-and-death processes, lattice path combinatorics and continued fractions," *Advances in Applied Probability*, vol. 32, no. 03, pp. 750–778, 2000.

[12] C. Banderier and P. Flajolet, "Basic analytic combinatorics of directed lattice paths," *Theor. Comput. Sci.*, vol. 281, no. 1–2, pp. 37–80, 2002.

[13] M. Bousquet-Mélou and M. Mishna, "Walks with small steps in the quarter plane," *Contemp. Math.*, vol. 520, pp. 1–40, 2010.

[14] G. Fayolle, R. Iasnogorodski, and V. Malyshev, *Random Walks in the Quarter Plane: Algebraic Methods, Boundary Value Problems, Applications to Queueing Systems and Analytic Combinatorics*, 2nd ed. Springer Publishing Company, Incorporated, 2017.

[15] D. G. Kendall, "Some Problems in the Theory of Queues," *J. Royal Stat. Soc. Series B (Methodological)*, vol. 13, no. 2, pp. 151–185, 1951.