Queuing Network Models of Multiservice RANs

Queuing Network Models of Multiservice RANs

ANDREA MARIN, Università Cà Foscari Venezia, Venezia, Italy
MICHELA MEO, Politecnico di Torino, Torino, Italy
MATTEO SERENO, Università di Torino, Torino, Italy
MARCO AJMONE MARSAN, Politecnico di Torino, Torino, Italy and IMDEA Networks Institute, Madrid, Spain

In this article, we present a new queuing network model for the analysis of a portion of a radio access network (RAN) comprising macro cell base stations (BSs) and small cell BSs offering "streaming" and "elastic" services. Streaming services require a certain data rate for a random time. The required data rates depend on the type of service (e.g., audio and video). Elastic services require the transfer of random data volumes, and their data rate adjusts dynamically based on the capacity not utilized by the streaming services. To derive performance measures for the proposed model, we develop a computationally efficient framework that exploits a new product form result for streaming services, relying on a well-known blocking policy, and an approximate product form for elastic services. Insensitivity to the distribution of service requirements holds in the case of negligible end user mobility.

We show the high accuracy of our model in predicting the performance of practical system configurations by conducting a thorough comparison between the model's results and those obtained from a detailed discrete-event simulator. Through this analysis, we uncover significant counter-intuitive behaviors that arise from the competition between streaming services with diverse demands, and that are effectively captured and predicted by our modeling approach.

Our computationally efficient queuing model is a useful new tool to support design and planning of multi-service RANs whose complex structures result from the coexistence of BSs of different generations in dense areas.

CCS Concepts: • **Networks** → **Network performance modeling**; **Network performance analysis**; • **Mathematics of computing** → *Markov processes;*

Additional Key Words and Phrases: Radio access network, streaming and elastic services, performance evaluation, queuing network model, product form, insensitivity

ACM Reference Format:
Andrea Marin, Michela Meo, Matteo Sereno, and Marco Ajmone Marsan. 2024. Queuing Network Models of Multiservice RANs. *ACM Trans. Model. Perform. Eval. Comput. Syst.* 9, 2, Article 7 (April 2024), 26 pages. https://doi.org/10.1145/3649307

Authors' addresses: A. Marin, Università Ca' Foscari Venezia, Venezia, Italy, via Torino 155, 30172; e-mail: marin@unive.it; M. Meo, Politecnico di Torino, Torino, Italy, Corso duca degli Abruzzi 24, 10129; e-mail: michela.meo@polito.it; M. Sereno, Università di Torino, Torino, Italy, Corso Svizzera, 185-10149; e-mail: matteo.sereno@unito.it; M. Ajmone Marsan, Politecnico di Torino, Torino, Italy, Corso duca degli Abruzzi 24, 10129; e-mail: ajmone@polito.it.

ACM 2376-3639/2024/04-ART7
https://doi.org/10.1145/3649307

1 INTRODUCTION

The **Radio Access Networks (RANs)** that today cover large metropolitan areas are quite complex. For example, the metropolitan area of Milan in Italy is covered by about 5,000 **Base Stations (BSs)** belonging to four infrastructured **Mobile Network Operators (MNOs)**. Not all BSs are equal: some transmit at high power and have a reach of 1 km or more; some use much lower power and have much shorter reach, down to tens of meters. Each BS provides services in the cell it defines, and the mix of services requested by end users is extremely diverse, including video, audio, images, text, and large and small data chunks, among others. According to the 5G jargon, services are grouped in service categories, such as eMBB (enhanced mobile broadband) mMTC (massive machine-type communications), and URLLC (ultra-reliable low-latency communications), implemented over network slices—that is, virtual networks implemented using (possibly non-disjoint) subsets of resources.

Using low power values, hence reducing the reach of BSs, has the double advantage of a lower amount of electromagnetic radiation and of a higher spatial reuse of the spectrum portions allocated to mobile communication services, which leads to an increase in capacity per unit area. The latter aspect, known under the name *spectrum reuse* or *network densification* has been the key factor that allowed RANs to dramatically increase their capacity over the years, thus serving more and more users at higher and higher data rates. It is expected that further densification will take place with the diffusion of 5G networks, and even more when 6G arrives.

The effective deployment of more than 1,000 BSs by an MNO in a metropolitan area calls for sophisticated design and planning tools to forecast both coverage and capacity, so as to provide customers with their chosen services in the most efficient and cost-effective way.

The planning problem is further complicated by the fact that several RAN generations typically coexist, at least for some periods of time, so that users can access services through BSs of different generations, with different capacity, different reach, and different technology. The resulting RAN architectures include overlapping cells of different sizes. Today, the most visible aspect of the presence of cells of different types is due to the coexistence of 4G and 5G, with the residual presence of older generations for some types of narrow band Internet of Things services.

The complexity of the design and planning problem is such that only rather small portions of a RAN can be considered at a time (accounting for hundreds or thousands of BSs in one study is not feasible), but even considering few cells at a time raises significant issues.

Simulation is typically very cumbersome and can be effective to verify the indications of analytical models. The solution of analytical models is also far from trivial, because the mix of different service types and the limited capacity of BSs make the analysis difficult. On the one hand, using a direct analysis approach based on continuous-time Markov chains (assuming exponential assumptions are acceptable) is not feasible due to the humongous size of the resulting state space; on the other hand, the standard queuing network modeling approach is also, in general, intractable. Indeed, product forms allowing an efficient numerical analysis of the systems are not known, unless additional assumptions are incorporated into the model.

In a recent work [18], we derived the conditions under which the queuing model of one BS offering multiple services admits a product form solution for the limiting joint probability distribution of the numbers of active services of different types, even allowing, under some conditions, non-exponential distributions for service durations or volumes. The considered service mix consists of streaming and elastic services. Streaming services require a constant data rate for their entire duration, whereas elastic services can adapt their data rate to the capacity available in the RAN. Streaming and elastic services can be viewed as the basic building blocks of the 5G service categories that we mentioned earlier.

The results of our previous work [18] showed that product form is achieved through a state-dependent admission control algorithm for elastic services, and in addition proved that the conditions for the existence of a product form solution, in the case of negligible mobility of end users, also lead to the insensitivity of the result to the distribution of service requirements, thus allowing the model to depart from exponential assumptions.

In this article, we use an approach inspired by our prior work [18] to study the queuing network model of a portion of a multiservice RAN comprising several BSs.

We prove that the queuing network model admits a product form solution for the limiting joint probability distribution of the numbers of active streaming services of different types at different BSs under a state-dependent routing of service requests for both types of services. In addition, we develop an approximate but accurate product form solution for elastic customers, and we prove that in case of negligible user mobility, the results of the product form are insensitive to service time distributions.

While a few works in the literature have discussed analytical models for the case of one BS loaded with traffic resulting from a mix of streaming and elastic services (see, e.g., [3, 5, 6, 8–10, 15, 17, 22–24])—but the possibility of a product form solution and of an insensitivity to service duration was never proved before [18]—to the best of our knowledge, no work has yet tackled the modeling of groups of several multiservice BSs.

The solution technique that we propose in this article can scale to portions of RANs comprising some tens of cells and a few thousand users.

2 THE RADIO ACCESS NETWORK

We consider a portion of a RAN comprising a number $N^{(BS)}$ of BSs and a number $N^{(UE)}$ of end user terminals (called **User Equipments (UEs)**). Each BS k, with $k = 1, 2, \ldots, N^{(BS)}$, has a user plane data rate C_k bit/s.

UEs roam over the area served by the RAN, and in different time periods they may be associated with different BSs (while at any time being associated with only one BS, except for short handover transients). The association of UEs with BS k (i.e., the dwell times of UEs at the BS) have durations described by the i.i.d. random variables δ_k.

UEs may request service from the BS with which they are currently associated. Services requested by UEs are of two different types: streaming and elastic.

Streaming services are characterized by a fixed data rate and a random duration. Streaming services can belong to different classes, depending on their data rate requirement and/or duration. Examples of streaming services are the access to real-time videos (e.g., to watch a live sport event or a newscast) or conference calls including audio and video streams from remote partners. We denote the number of streaming service classes with S. Class i ($i = 1, 2, \ldots, S$) is characterized by a fixed data rate $R_i^{(s)}$ and by service durations described by i.i.d. random variables. The duration of a generic class i service is denoted by $\tau_i^{(s)}$.

Elastic services are characterized by a random volume of data to be transferred and by the possibility to use a variable data rate (that may drop temporarily to zero), according to the available data rate at the BS. Elastic services can belong to different classes, depending on the volume of data to be transferred. Examples of elastic services are the download of web pages or recorded videos, and the transmission of messages possibly including audio, video, and images. We denote with E the number of elastic service classes. Class j ($j = 1, 2, \ldots, E$) is characterized by volumes of data to be transferred that are described by i.i.d. random variables. The volume to be transferred by a generic class j service is denoted by $\varphi_j^{(e)}$.

A BS (say BS k) allocates resources for a service either because one of the UEs associated with BS k issues a new service request or because a UE with an active service that was previously

associated with a different BS becomes associated with BS k due to roaming. A streaming service request of class i is accepted by BS k only if the BS can allocate the required data rate $R_i^{(s)}$ (possibly reducing the data rate used by elastic services); otherwise, the request is blocked. Elastic service requests use the BS data rate not used by streaming customers (we will refer to such data rate with the term *residual data rate*). Elastic service requests of class j are always accepted by BS k, due to the fact that they have no minimum data rate requirement, until the maximum number of admissible simultaneous class j services, $N_{j,k}^{(e)}$, is reached. All active elastic services evenly share the residual data rate, regardless of their class. BS k can reserve a portion of its user plane data rate C_k to be used by elastic services, so as to avoid that the data rate available to elastic services drops to zero. We denote the data rate reserved to elastic services in BS k as $C_k^{(e)}$.

Summing up, streaming services are blocked if at the epoch of their arrival there is not enough free bandwidth to begin their service, whereas elastic services are dropped if at their arrival epoch they find the buffer for elastic services saturated. Furthermore, when a request is blocked, it may still be served by an adjacent BS if this has free resources and the connection is physically possible, and otherwise it is dropped. Of course, the possibility of serving the request from a different BS greatly depends on the RAN topology, and in particular on the location of the UE and of the BSs in the area. If the cells defined by the BSs overlap in the UE location (i.e., the signals emitted by the BSs reach the UE with sufficient strength and low enough interference), then it is possible to choose alternate BSs to serve the UE request when the preferred BS does not have enough bandwidth. The selection of the BS to which the UE addresses its service request is usually done by the BS based on the measurements performed by the UE on the quality of the signal it receives from BSs in the area.

Active services leave a BS (say BS k) for one of three possible reasons: (i) the service reaches completion, where in case of streaming services, this means that the total amount of time of service (accounting for service times in all cells visited during service) reaches the value sampled for $\tau_i^{(s)}$, and in case of elastic services, completion means that the total amount of transferred data (accounting for all cells visited during service and the available data rate) reaches the value sampled for $\varphi_j^{(e)}$; (ii) the UE modifies its association to a BS different from k (e.g., because of roaming); and (iii) the active service request that reaches BS k is blocked upon arrival.

The changes of association of a UE (say UE u) from one to another of the BSs present in the service area because of roaming are described by the routing probabilities of its services. In this case, when UE u requests to change its association from BS k to a different BS, the service request reaches BS ℓ, with probability $p_{k,\ell}^{(M)}$. Note that this probability is conditional on the UE leaving the BS due to mobility, not to service completion. Obviously, we have

$$\sum_{\substack{\ell=1 \\ \ell \neq k}}^{N^{(BS)}} p_{k,\ell}^{(M)} = 1 \qquad \forall k \in 1, 2, \ldots N^{(BS)},\ p_{kk}^{(M)} = 0\,. \tag{1}$$

3 THE QUEUING NETWORK MODEL

The RAN portion under investigation is modeled with a network of queues in which each BS corresponds to a queue. Customers moving over the queuing network correspond to service instances. Hence, customers are said to be either streaming or elastic of the appropriate class, depending on the type of service they correspond to.

In the case of streaming customers of class i, we assume that service times at queue k are modeled by i.i.d. exponentially distributed random variables with rate $\mu_{k,i}^{(s)}$. In the case of elastic customers

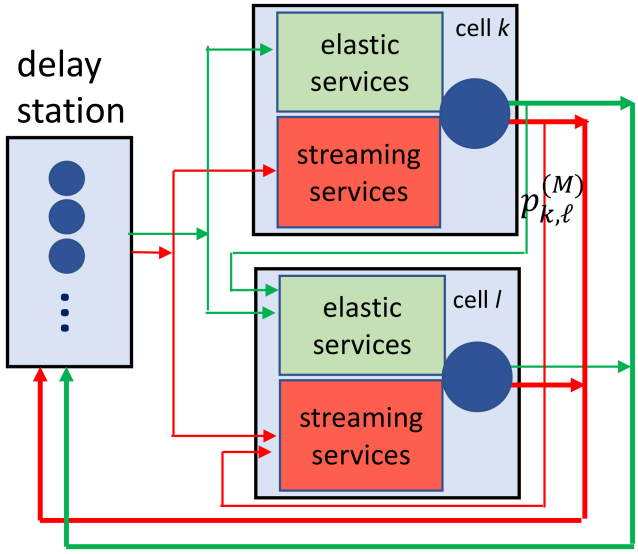


Fig. 1. Sketch of the queuing network model.

of class j, we assume that the amounts of data to be transferred are modeled by i.i.d. exponentially distributed random variables with rate $\nu_{k,j}^{(e)}$. Insensitivity to service data/volume distributions will be discussed later on. For all customers, we assume that dwell times are modeled by i.i.d. exponentially distributed random variables with rate $\mu_{k,H}$.

The service of (non-blocked) streaming customers of class i at queue k follows a multiserver paradigm, since services proceed in parallel, each one for a time duration equal to the minimum between the time to completion and the dwell time, hence for exponentially distributed times with rate $\mu_{k,i}^{(s)} + \mu_{k,H}$.

The service of (non-blocked) elastic customers follows a state dependent processor sharing paradigm, since elastic services proceed in parallel, fairly sharing the BS residual data rate. The service time duration for elastic customers of class j results from the combination of the volume of data to be transmitted, the dwell time, and the residual data rate.

One additional infinite-server queue (called *delay station*) models the time between the end of a (streaming or elastic) service instance and the generation of a new service request (of the same type) from the same UE. At this queue, that will be numbered 0 (all other queues inherit the numbering of the corresponding BS), and service times are modeled by arbitrary i.i.d. random variables with mean $(\mu_{0,i}^{(s)})^{-1}$ and $(\mu_{0,j}^{(e)})^{-1}$ in the case of streaming (or elastic) customers of class i (j). A sketch of the queuing network is provided in Figure 1; for the sake of simplicity, the sketch refers to the delay station and two queues (corresponding to two BSs) only, namely k and l. Thin lines represent roaming, whereas thick lines represent service completion.

In case a customer reaches a queue where it cannot be accepted (because the available data rate is not sufficient in the case of a streaming customer, or, in the case of an elastic customer, because the maximum admissible number has already been reached), we assume it is transferred to queue ℓ with probability $p_{k,\ell}^{(B)} = p_{k,\ell}^{(M)}$ for all k and ℓ, which implies that a blocked service request behaves exactly as a non-blocked request in terms of routing. This assumption is known with the name of *skipping* in the literature and is instrumental to our modeling approach. The assumption is reasonable in the case of further handovers, since a request that cannot be accepted

at a BS tries to move to another one of the neighboring BSs, but these are the same toward which handovers are possible. Similarly, customers that cannot be accepted at a queue are moved to the delay station with the same probability with which service would complete at the queue. This is interpreted as a forced completion, hence a true blocking of the request. It is worth noting that in real networks, it is reasonable to assume that both $p^{(B)}_{k,\ell}$ and $p^{(M)}_{k,\ell}$ are non-zero for the same values of ℓ (i.e., for those ℓ that correspond to BSs in physical proximity of the UE). While imposing that $p^{(B)}_{k,\ell} = p^{(M)}_{k,\ell}$ is required for the network of queues to admit product form, it is also a realistic assumption whenever (as is usually the case) the decision to move a request from a cell to another depends on the quality of the received signal and, hence, on the distance of the UE from the BS.

4 STATIONARY ANALYSIS OF THE QUEUING NETWORK

We are interested in computing the limiting joint distribution of the numbers of customers of the different streaming and elastic service classes at all queues. Although the CTMC underlying the model is finite, the very high cardinality of the state space for realistic scenarios makes the solution of the system of global balance equations intractable. For this reason, we investigate the existence of a product form solution allowing the definition of a numerically stable and efficient algorithm for the computation of the performance indices.

More formally, we are interested in computing the joint probabilities

$$P\{\mathcal{N}_0 = \mathrm{N}_0, \mathcal{N}_1 = \mathrm{N}_1, \ldots, \mathcal{N}_{N^{(BS)}} = \mathrm{N}_{N^{(BS)}}\} \tag{2}$$

with

$$\mathcal{N}_k = \left[\mathcal{N}^{(s)}_k, \mathcal{N}^{(e)}_k\right], \qquad \mathcal{N}^{(s)}_k = \left[\mathcal{N}^{(s)}_{k,1}, \ldots, \mathcal{N}^{(s)}_{k,S}\right], \qquad \mathcal{N}^{(e)}_k = \left[\mathcal{N}^{(e)}_{k,1}, \ldots, \mathcal{N}^{(e)}_{k,E}\right],$$

where the random variables $\mathcal{N}^{(s)}_{k,i}$ and $\mathcal{N}^{(e)}_{k,j}$ indicate the numbers of streaming services of class i and elastic services of class j in progress at BS k, respectively.

In addition,

$$\mathrm{N}_k = \left[\mathrm{N}^{(s)}_k, \mathrm{N}^{(e)}_k\right], \qquad \mathrm{N}^{(s)}_k = \left[n^{(s)}_{k,1}, \ldots, n^{(s)}_{k,S}\right], \qquad \mathrm{N}^{(e)}_k = \left[n^{(e)}_{k,1}, \ldots, n^{(e)}_{k,E}\right],$$

where $n^{(s)}_{k,i}$ and $n^{(e)}_{k,j}$ are the values assumed by the random variables $\mathcal{N}^{(s)}_{k,i}$ and $\mathcal{N}^{(e)}_{k,j}$, respectively.

The existence and uniqueness of the limiting distribution is ensured by the ergodicity of the CTMC underlying the queuing network that, in turn, is implied by the observation that the state space is finite and irreducible.

Observe that the marginal distribution for the network of streaming services can be obtained without considering the network of elastic services. More precisely, the stochastic process underlying the network of streaming services is a modulating process (or environment) for the Markov modulated process underlying the network of elastic services.

As a consequence, we propose a solution method that consists of two steps. First, we describe only the network of streaming services, we derive the conditions for its product form solution, and we give a convolution-based algorithm for the efficient computation of the stationary performance indices. In this step, we also study the conditions for the insensitivity of the model to the distribution of the service times. In the second step, we resort to a standard approximation for the analysis of Markov modulated processes. In fact, it is well known that, in general, exact solutions for such models are intractable [19], thus we develop a computationally efficient method to obtain accurate estimates of the stationary performance indices including dropping probabilities. The efficiency of this method relies on the product form solution of the network of elastic services conditioned to the state of the streaming network.

4.1 Product Form Solution for the Network of Streaming Services

Let us consider the network of streaming services with the service and blocking policy defined in Section 3. The following theorem shows that there exists a product form solution both among the stations that form the streaming network and within each station among the classes of service.

Theorem 1 (Product Form for the Streaming Network). *The streaming network with the blocking policy defined in Section 3 has product form stationary distribution—that is,*

$$P\left\{\mathcal{N}_0^{(s)} = \mathbf{N}_0^{(s)}, \ldots, \mathcal{N}_{N^{(BS)}}^{(s)} = \mathbf{N}_{N^{(BS)}}^{(s)}\right\} = \frac{1}{G^{(s)}} \prod_{k=0}^{N^{(BS)}} g_k\left(\mathbf{N}_k^{(s)}\right), \tag{3}$$

and moreover, we have that

$$g_k\left(\mathbf{N}_k^{(s)}\right) = \prod_{t=1}^{S} g_{k,t}\left(n_{k,t}^{(s)}\right) \tag{4}$$

at every queue k, where $k = 0, \ldots, N^{(BS)}$ and $G^{(s)}$ is the normalizing constant. Functions $g_{k,t}(n_{k,t}^{(s)})$ are defined as

$$\forall k > 0, t \qquad g_{k,t}\left(n_{k,t}^{(s)}\right) = \frac{1}{n_{k,t}^{(s)}!}\left(\frac{\sigma_{k,t}}{\mu_{k,H} + \mu_{k,t}^{(s)}}\right)^{n_{k,t}^{(s)}}$$

and $g_{0,t}(n_{0,t}^{(s)})$ is defined, as for the infinite server station of BCMP theorem, $\sigma_{k,t}$ is any non-trivial solution of the BCMP-like system of traffic equations [4] on the routing probabilities defined for streaming service t as

$$p_{k,\ell}^{t*} = \frac{\mu_{k,H}}{\mu_{k,H} + \mu_{k,t}^{(s)}} p_{k,\ell}^{(M)} + \frac{\mu_{k,t}^{(s)}}{\mu_{k,H} + \mu_{k,t}^{(s)}} \mathbb{1}_{\ell=0} \qquad k > 0,$$

where $\mathbb{1}$ is the indicator function and $p_{0,\ell}^{t*}$ is the probability that a service is initiated at cell ℓ.

The routing matrices $[p_{k,\ell}^{t*}]$ used to solve the model are obtained as a mixture of routing probabilities due to mobility and to service completion. In other words, routing probabilities in the queuing network do not coincide with roaming probabilities in the system. Indeed, since the residence times of all streaming jobs are independent of the states of the stations, and given the exponential assumption, the probability of terminating the service because of mobility is $\mu_{k,H}/(\mu_{k,H} + \mu_{k,t}^{(s)})$. Moreover, the residence time is still exponentially distributed because it turns out to be the minimum of two independent exponentially distributed random variables (mobility and service times).

Proof. In contrast with the monolithic proofs based on the verification of the system of global balance equations of the entire streaming network as in the work of Pittel [20], our proof follows the ideas of Balsamo et al. [2], or alternatively we may resort to the extended quasi-reversibility result of Chao et al. [13]. Notice that the application of the **Reversed Compound Agent Theorem (RCAT)** (see, e.g., [2]) implies that the resulting product form solution satisfies the system of global balance equations of the queuing network.

According to the RCAT methodology, we have to consider each queue in isolation and identify those transitions corresponding to job departure events that cause a routing to another queue. Then, we reverse this process and verify the conditions of the theorem that, in our case, can be summarized as follows: (i) each state of the queues can be reached thanks to a job departure, (ii) a departure event from a non-empty queue is always possible, and (iii) the rate of the departure transitions in the reversed process is the same for each flow in the network. Condition (i) is trivially satisfied. Condition (ii) is satisfied thanks to the skipping policy (see also [2]). As for condition (iii),

we consider a single isolated queue k serving streaming services. If the queue can accommodate $n_{k,1}^{(s)}, \ldots, n_{k,S}^{(s)}$ customers, then class t customers are served with rate $n_{k,t}^{(s)}\mu_{k,t}^{(s)}$ and leave the station for mobility with rate $n_{k,t}^{(s)}\mu_{k,H}$. This corresponds to a situation in which customers of class t are served with rate $\mu_{k,H} + \mu_{k,t}^{(s)}$, and at the service completion, we use the mobility routing with probability $\mu_{k,H}/(\mu_{k,H}+\mu_{k,t}^{(s)})$ and the service completion routing with its complement. This implies that the process underlying a single isolated queue is reversible (but not the process underlying the joint process!), thus the reversed rate of the service completion events (due either to mobility or actual service completion) are equal to the intensity of the arrival process at station k for each class.

In conclusion, according to RCAT, this ensures the product form of the model. For what concerns the multiclass delay station, its product form properties are well known from the literature (see, e.g., [4]). □

Notice that although we inherit the skipping policy from other works [2, 20], the proof of the product form for networks of Erlang-B stations is, to the best of our knowledge, new.

Corollary 1 (Insensitivity). *If the effect of mobility is negligible (i.e., mobility is very slow with respect to service completion), the model is insensitive to moments of the service time distribution higher than the first one.*

Proof Sketch. Notice that the topology of the network imposes that, after completion, service requests return to the delay station. If we neglect mobility, we can represent the service time distribution with a Coxian distribution and observe that the proof of Theorem 1 still holds true. Coxian distributions are dense in the domain of non-negative continuous distributions, and this suffices to conclude the proof. □

For what concerns the algorithm for the computation of the normalizing constant $G^{(s)}$, the standard convolution algorithm for multiclass queuing networks (see [11, 12] for details) cannot be applied in our model because of the finite capacity of the queues. The relations between the convolution constants and the average stationary performance indices are different from those well known from the literature because of the finite capacity of some stations, but their derivation is totally analogous [11, 12]. Notice that the choice of resorting to a convolution algorithm rather than a mean value analysis will be clear in Section 4.3 since we will use the convolution table to sample a subset of states of the network of streaming services.

Theorem 2 explains how we can efficiently compute the normalizing constant required by Theorem 1. If we denote by $\mathbf{m}^{(s)}$ the population of streaming services (e.g., the maximum number of customers in each streaming class), and focus the attention on a given station (i.e., station k), we can denote by

$$\mathcal{S}_k(\mathbf{m}^{(s)}) = \left\{ (n_1, \ldots, n_S) : n_i \leq m_i^{(s)} \wedge \sum_{i=1}^{S} n_i R_i^{(s)} \leq C_k \right\} \tag{5}$$

the set of all vectors $\mathbf{n}$ that can populate station k when the network has $\mathbf{m}^{(s)}$ customers, and we assume $C_0 = \infty$.

Theorem 2 (Convolution for Streaming Services). *For the product form network of streaming services, the following relation between normalizing constants holds:*

$$G_{\Omega,\mathbf{m}^{(s)}}^{(s)} = \sum_{\mathbf{n}\in\mathcal{S}_k(\mathbf{m}^{(s)})} g_k(\mathbf{n}) G_{\Omega\setminus\{k\},\mathbf{m}^{(s)}-\mathbf{n}}^{(s)},$$

where Ω is a non-empty (sub)set of stations and $k \in \Omega$ is an arbitrary element, $\mathbf{m}^{(s)} = (m_1^{(s)}, \ldots, m_S^{(s)})$ is the population of streaming services, and $\mathcal{S}_k(\mathbf{m}^{(s)})$ is defined by Equation (5).

Proof. To simplify the notation and thus make the proof clearer and more understandable, let us assume that there is only one class of streaming services; therefore, instead of $\mathbf{m}^{(s)}$, we may simply use $N^{(UE)}$. The extension to a greater number of classes is straightforward. For the proof of this theorem, we denote the state space of the queuing network as follows:

$$(\Omega, N^{(UE)}) = \left\{ \mathbf{n} = (n_0, n_1, \ldots, n_{BS}) : n_k \text{ is such that } n_k \ldots R_k \leq C_k, \text{for } k = 1, \ldots, BS, \text{and } \sum_{k=0}^{BS} n_k = N^{(UE)} \right\}. \quad (6)$$

From Equation (3), we can derive that

$$G_{(\Omega, N^{(UE)})} = \sum_{\mathbf{n} \in (\Omega, N^{(UE)})} \prod_{k=0}^{N^{(BS)}} g_k(n_k). \quad (7)$$

The normalizing constant depends on the number of stations, on the number of end user terminals (i.e., customers in the queuing network), and on the data rate available at each station. Note that the maximum number of streaming services that can access the station is derived from the data rates of the stations. In other words, the capacities, in terms of the maximum number of customers that can access the station, are obtained from the data rates.

Similarly to the classical Buzen derivations [12], we must define the boundary conditions. In particular,

$$G_{(\Omega', 0)} = 1\,, \quad (8)$$

where $\Omega' \subseteq \Omega$, and

$$G_{(\{h\}, N)} = \sum_{j \in (\{h\}, N)} g_h(j)\,. \quad (9)$$

The previous equation defines the computation of the normalizing constant when the set of stations includes only one station, the one with index h, and each state of $(\{h\}, N)$ is an integer (or to be more precise, a vector with only one component).

Moreover, it may happen that for a given combination of the number of customers, and a subset of stations, the corresponding set of states is empty. In this case, we can write

$$G_{(N', \Omega')} = 0, \quad \forall N' : (N', \Omega') = \emptyset, \quad \text{with } \Omega' \subseteq \Omega\,. \quad (10)$$

We can rewrite Equation (7) by focusing on the number of customers in a given station (i.e., station h), and then we have

$$G_{(N, \Omega)} = \sum_{j=0}^{N} \left\{ \sum_{\substack{\mathbf{n} \in (\Omega, N)) \\ \text{with } n_h = j}} \prod_{k=0}^{N^{(BS)}} g_k(n_k) \right\}. \quad (11)$$

Note that there can be values for j that are not 'reachable' as states of station h. Classic examples concern values of j that exceed the capacity of the station (i.e., $j : j \cdot R_h > C_h$). For such 'illegal' values of j, the corresponding partition of the state space (Ω', N') with $n_h = j$ is empty. This is the way in which finite capacity stations are managed. The previous equation can be rewritten as

$$G_{(N, \Omega)} = \sum_{j=0}^{N} g_h(j) \left\{ \sum_{\mathbf{n} \in (\Omega - \{h\}, N - j)} \prod_{k=0}^{N^{(BS)}} g_k(n_k) \right\}. \quad (12)$$

This concludes the proof. □

It must be clear that this theorem can be considered a direct derivation of Buzen's result for stations with state-dependent service stations. However, it is worth emphasizing that in our case, the state dependency is used to manage the finite capacity of the stations. It follows that the algorithm we use is an adaptation of Buzen's algorithm for queuing networks with state-dependent stations, and we employ the classical implementation described in the work of Bruell and Balbo [11].

Similarly to what has been done for the convolution method proposed by Buzen [12], the operation of the algorithm can be described with the help of a two-dimensional "tableau" with as many columns as there are stations in the queuing network and as many rows as the number of customers. The first row and the first column of the tableau are initialized according to Equations (7) and (9), whereas all other elements of the tableau are computed by using (12) (with the condition defined by (10)).

The time complexity is the same of the one developed for classical product form queuing networks with state dependent service stations—that is, $O((N^{(BS)}+1)\cdot(N^{UE})^2)$.

Remark. We must point out that for what concerns the normalizing constant algorithm, the original (small) contribution of this article lies in the application of Buzen's algorithm to classical product form queuing networks with *state-dependent service stations* to product form queuing networks with blocking due to stations with finite capacity, skipping policy, and *stations with state-independent service rates*. For all other aspects, the algorithm we use is similar to Buzen's algorithm. We must add that all of the derivations we have presented refer to the case of only one class of streaming services, but our algorithm can be easily extended to the case of multiple classes of streaming services. In the case of such extension, the complexity, obviously, changes with the number of classes, as the exponent of the term that takes into account the number of customers. In the case of systems with a growing number of classes (this could be the case of next-generation RANs with many different types of services), one could resort to adapting algorithms developed for product form queuing networks that explore tradeoffs between the number of classes and the number of stations. A remarkable example of this type of algorithms is Recal [14]. Of course, the new algorithms for product form queuing networks with blocking due to stations with finite capacity and skipping policy must be adapted in the same way as we adapted the algorithm for the case of a single class.

4.2 Product Form for the Network of Elastic Services Conditioned on the State of Streaming Networks

Assume that the network of streaming services is in a certain state $\mathbf{N}^{(s)}$ and persists in this state for a sufficient long time to let the stochastic process underlying the network of elastic services reach its limiting distribution. Clearly, this distribution depends on $\mathbf{N}^{(s)}$ since this state determines the residual data rate for elastic services. Henceforth, in this subsection, we will reason under this assumption.

Similarly to the case of streaming services, the network of elastic services presents a product form solution. However, in this case the stations have constant service capacity (with the exception of the delay station) and processor sharing discipline. Given the small amount of data corresponding to elastic services, we ignore the effects of mobility. Therefore, Theorem 1 reduces to the result of Pittel [20]—that is, functions $g_{k,t}$ must be replaced by

$$h_{k,t}(n_{k,t}^{(e)}) = \begin{cases} \left(\frac{p_{0,k}}{\mu_{k,t}^{(e)}}\right)^{n_{k,t}^{(e)}} & \text{if } 0 < k \leq N^{(BS)} \\ \frac{1}{n_{0,t}^{(e)}!}\left(\frac{1}{\mu_{0,t}^{(e)}}\right)^{n_{0,t}^{(e)}} & \text{if } k = 0, \end{cases}$$

where, for $k > 0$, we define the service rate for class t elastic service as the proportion of the residual data rate used for serving the requests:

$$\mu_{k,t}^{(e)} = \frac{C_k - \sum_{i=1}^{S} R_i^{(s)} N_{k,i}^{(s)} + C_k^{(e)}}{\varphi_t^{(e)}} \frac{n_{k,t}^{(e)}}{\sum_{i=1}^{E} n_{k,i}^{(e)}} .$$

Also the network of elastic services is insensitive to moments higher than the first of the data to be transferred.

Finally, although convolution Theorem 2 holds also for the elastic network, we can resort to the mean value analysis proposed in the work of van der Gaast et al. [25] since the network satisfies the assumptions of the model with finite capacity stations analyzed in this work. In this case, the computational complexity is bounded by $O((N^{(BS)} + 1)^2 H)$, $H = \prod_{i=1}^{E}(m_i^{(e)} + 1)$, where $m_i^{(e)}$ denotes the total number of elastic services of class i since all stations with the exception of the delay station have a fixed service rate.

4.3 Approximate Solution of the Joint Model between Streaming and Elastic Services

The standard approach to the approximate solution of Markov modulated processes is that of computing the stationary probabilities of the modulated process (elastic service network) conditioned to each state of the environment (streaming service network), assuming that this does not change (see, e.g., [16]). The underlying assumption is that the modulated process reaches the stationary behavior conditioned to the environment state after a short transient, and hence the marginal stationary probabilities can be obtained by averaging these values. More formally, let $\pi(e, m)$ be the joint stationary probability and $\pi_e(e)$, $\pi_m(e)$ the marginal stationary distributions of the environment and the modulated process, respectively. Then, it holds that

$$\pi_m(m) = \sum_e \pi(m|e)\pi_e(e) \simeq \sum_e \pi^*(m|e)\pi_e(e)\,, \tag{13}$$

where $\pi^*(m|e)$ is the stationary distribution of the modulated process conditioned to a non-changing environment in state e.

In our case, $\pi_e(e)$ can be computed efficiently thanks to Theorems 1 and 2, and the analogues for the elastic service network allow us to compute efficiently $\pi^*(m|e)$. However, we have to tackle the problem of the cardinality of the state space of the streaming services that makes a direct use of (13) unfeasible.

We propose a novel method to solve Markov modulated processes when the cardinality of the state space of the environment is very high. The idea is to sample L states from the space of the streaming network according to the technique of *perfect sampling*—that is, the samples are chosen with a probability that exactly corresponds to their stationary probability (see, e.g., [7]).

Perfect sampling is usually achieved thanks to an algorithm called *coupling from the past* [21]. Briefly, when the stationary distribution of the CTMC is not known, or when it is known but difficult to manage because of the state space cardinality, one may run the simulation of parallel discrete time chains starting in each state. The goal is to find a t such that the simulations of the parallel chains meet at the same state before $t = 0$.

However, in our case, we know the probability distribution that we want to use for our samples. Therefore, the problem reduces to a problem of sampling from a known distribution. In general, this may be not easy, but we can exploit two important characteristics of our model—that is, its product form distribution and the entries of the convolution tables that we have generated to obtain the normalizing constant. In fact, the convolution approach generates a table of normalizing constants from which we can draw the samples according to their stationary distribution. Suppose

we have applied the algorithm to the stations in ascending order $0, \ldots, N^{(BS)}$. In accordance with convolution theory on queuing networks [11, 12], the marginal distribution of station $N^{(BS)}$ is given by

$$P\{\mathcal{N}_{N^{(BS)}} = \mathbf{N}_{N^{(BS)}}\} = g_{N^{(BS)}}(\mathbf{N}_{N^{(BS)}}) \frac{G^{(s)}_{\Omega\setminus\{N^{(BS)}\}, \mathbf{m}^{(s)}-\mathbf{N}_{N^{(BS)}}}}{G^{(s)}_{\Omega, \mathbf{m}^{(s)}}}.$$

At this point, we can sample component $N^{(BS)}$ of state vector. The step can be recursively repeated for station $N^{(BS)} - 1$ conditioned on the choice made for station $N^{(BS)}$ and so on until we reach station 0. Therefore, with $N^{(BS)}$ random variates per sample and the convolution table, we can construct our perfect sample set $\mathcal{P}$ very efficiently. Thus, we can approximate Equation (13) as follows:

$$\pi_m(m) \simeq \frac{1}{L} \sum_{\mathbf{m}^{(e)} \in \mathcal{P}} \pi^*(m|e),$$

whose evaluation is fast, thanks to the low complexity of the convolution algorithm for elastic services.

The number of samples contained in $\mathcal{P}$ is clearly related to the accuracy that one desires to achieve. However, (13) is already an approximation and represents the main source of error in the results. According to our experiments, a few hundred samples are sufficient to have very accurate results as discussed in Section 5.

4.4 Computation of the Blocking Probabilities

A streaming service is blocked in two cases: (i) it exits from the delay station and returns to the same station without being served by any BS (because either no BS has enough bandwidth to serve the request or the skipping policy brings it back to the delay station) or (ii) its service at a BS is interrupted because of mobility, but the job returns to the delay station (for the same reasons as mentioned previously). These events occur with a state-dependent probability following the skipping routing described before. To compute the blocking probability, we use a relation among the network station throughputs that we obtain from the convolution table (see, e.g., [11, 12]). More specifically, let $X^{(s)}_{0,t}$ be the throughput of the delay station for class t, and let $X^{(s)}_{k,t}$ be the throughput of station $k > 0$ for class t. Among the jobs that leave station k, only $\mu^{(s)}_{k,t}/(\mu_{k,H} + \mu^{(s)}_{k,t})$ complete their service, whereas the others try to access another BS. Thus, the total throughput of the network for class t can be expressed by

$$X^{(s)}_t = \sum_{k=1}^{N^{(BS)}} \frac{\mu^{(s)}_{k,t}}{\mu_{k,H} + \mu^{(s)}_{k,t}} X^{(s)}_{k,t}.$$

Since the total service demand is $X^{(s)}_{0,t}$, the blocking probability for class t is $1 - X^{(s)}_t / X^{(s)}_{0,t}$. A similar reasoning (actually simpler, thanks to the assumption of no mobility) holds also for elastic services.

4.5 Methodological Contributions

Product forms have been widely applied for the performance evaluation of telecommunication systems. This article relies on this theory to develop a novel efficient algorithm for the solution of a queuing network with multiple classes of customers that interact in a peculiar way. To the best of our knowledge, this is the first time that the *skipping* policy, first introduced in the work of Pittel [20] and still widely studied (as, e.g., in [25]) has been applied to networks of multiclass queues of Erlang type with alternative routing for impatient customers. This new product form model has required a specific convolution algorithm for the computation of the normalizing constant that is

a non-trivial generalization of the one presented in other works [11, 12]. The performance indices obtained from the convolution output are original since they must account for the finite capacity and the competition of the customer classes. Indeed, we do not have a traditional 'queue capacity' that can accommodate a certain number of customers for each class, but the availability of the data rate gives the condition of blocking. Finally, we are not aware of any other work using a perfect sampling algorithm based on the convolution table to study Markov modulated processes. This approach is flexible in the sense that it allows one to balance the desired accuracy with the computational efforts by choosing the amount of samples to consider.

5 EXPERIMENTS

In this section, we present two different sets of numerical results. The first one refers to a hypothetical RAN layout, whereas the second considers a real placement of macro and small cell BSs.

5.1 A Road Segment

We consider a portion of a RAN offering services over a road segment with one macro cell BS (named BS_M, and indexed 1) and three small cell BSs (named BS_{S1}, BS_{S2}, BS_{S3}, and indexed 2, 3, 4). The reach of BS_M includes the whole road segment, whereas the coverage of each of the three small cells includes only a portion of the road segment, and small cells are adjacent to one another. We assume that each small cell has an area equal to 1/9 of the area reached by the macro cell so that 1/3 of the considered area is served by the three small cells and the remaining 2/3 are served by the macro cell.

Since the density of UEs is assumed to be constant over time, whenever a UE requests a new service after an idle period, the request is directed to a BS with a probability proportional to the cell area, hence 2/3 for the macro cell BS and 1/9 for each small cell.

The requests for handover from cell i to cell j have the following values:

$$\mathbf{P^M} = [P^M_{i,j}] = \begin{bmatrix} 0 & 3/8 & 2/8 & 3/8 \\ 3/4 & 0 & 1/4 & 0 \\ 1/2 & 1/4 & 0 & 1/4 \\ 3/4 & 0 & 1/4 & 0 \end{bmatrix}. \tag{14}$$

UEs can request access to the streaming and elastic services offered by the RAN. A streaming service instance can belong to one of two classes: either audio or video. A video service requires a data rate equal to 10 Mb/s, whereas an audio service requires 100 kb/s. Video streaming service durations are i.i.d. exponentially distributed random variables with an average of 1,800 seconds. Audio durations are i.i.d. exponentially distributed random variables with an average of 600 seconds. Elastic services require the transfer of i.i.d. exponentially distributed random amounts of data with an average of 1 Mb. The capacity of each BS (for macro as well as small cells) is 300 Mb/s, of which 5 are reserved for elastic services. Streaming services are accepted at a BS as long as the data rate is available (without consuming the 5 Mb/s reserved for elastic services). Elastic services are accepted up to a maximum number set equal to 50. The dwell times of UEs in BS_{S1}, BS_{S2}, BS_{S3} are i.i.d. exponentially distributed random variables with an average of 400 seconds, whereas the dwell times in BS_M are i.i.d. exponentially distributed random variables with an average of 600 seconds. Note that these numbers are derived from the specified user density in the cell and the cell area. The number of small cells we consider within the macro cell area is typical of today's RANs (as we see in the real deployment example that we discuss next), and the fact that all BSs use the same data rate stems from the fact that operators often deploy the same equipment to implement macro and small cells, changing just the emitted power. Of course, our model copes equally well with different BS data rates.

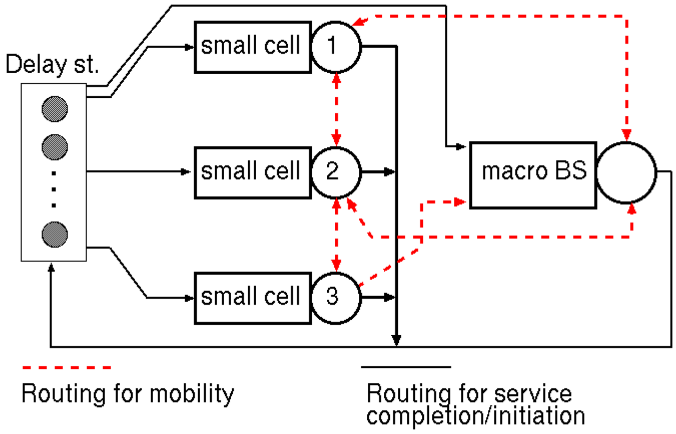


Fig. 2. Sketch of the queuing network modeling the RAN portion comprising one macro cell BS and three small cell BSs.

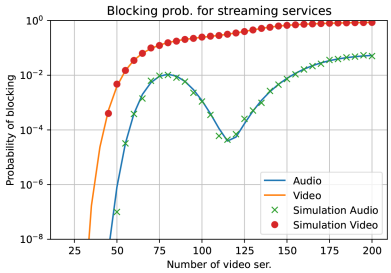
After a streaming service completion, UEs remain idle for i.i.d. exponentially distributed random times with an average of 200 seconds before issuing a new service request of the same type. Instead, the dynamic of elastic services is much faster. Following completion, a new request is issued after i.i.d. exponentially distributed random times with an average of 5 seconds. We consider the case of a fixed number $N^{(UE)}$ of UEs in the road segment, and we study the network performance for variable values of $N^{(UE)}$. For every video user in the area, we have two audio and two elastic users. We study the system for up to 200 video users, hence 1,000 users in total.

Figure 2 shows the structure of the queuing network model used to study the described portion of a RAN.

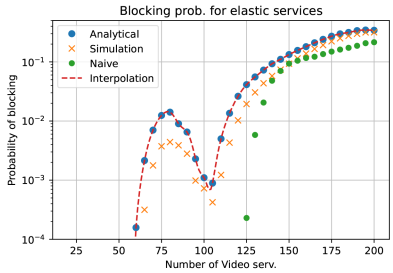
The first two plots in Figure 3 report the blocking probabilities for video and audio streaming services and for elastic services, as a function of the number of video streaming customers in the network of queues.

In Figure 3(a), we report analytical predictions and simulation results for the blocking probability of audio and video services. Observe the extremely good match between analytical and simulation results (the analytical model is indeed exact for these services so that the good match is a validation of the simulator). The other evident and surprising element of the plots is the oscillation in the blocking probability of audio services. This phenomenon is in line with what was already observed in our previous work [18] in the case of a single cell loaded by streaming and elastic services, and results from the interplay of video services with a high data rate and audio services whose data rate is two orders of magnitude smaller. Blocking one video service makes room for 100 voice services, and when the probability of blocking one more video requests becomes significant, the probability of blocking for audio services decreases.

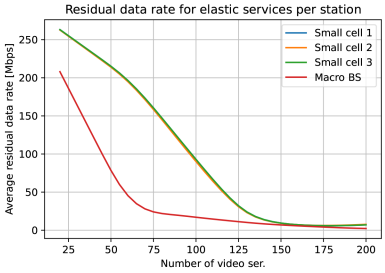
It should be observed that a service level agreement with a blocking probability limit for audio services equal to 10^{-3} is respected up to about 60 video customers, violated for a number of video customers between about 60 and 100, respected again for a number of video customers between about 100 and 130, and finally violated when the number of video customers is more than about 130. This behavior is quite unexpected and should be known to an MNO, who can activate appropriate measures to avoid user dissatisfaction (e.g., setting a limit to the maximum number of simultaneously active video services which is compatible with the first crossing of a desired threshold for the blocking probability of audio services, as predicted by our model—if such threshold is set to 10^{-3}, our model says that the number of active



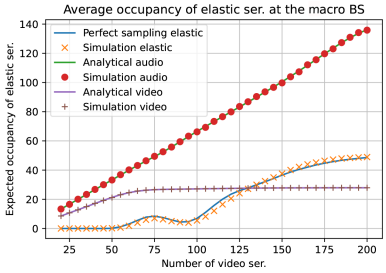
(a) Blocking probabilities for streaming services



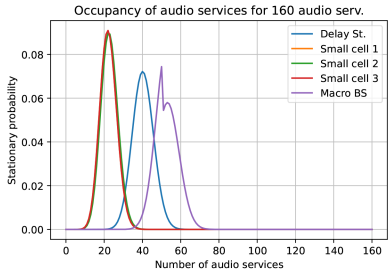
(b) Blocking probabilities for elastic services



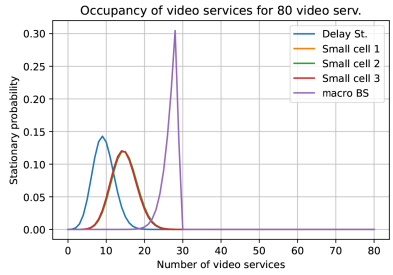
(c) Available data rate for elastic services



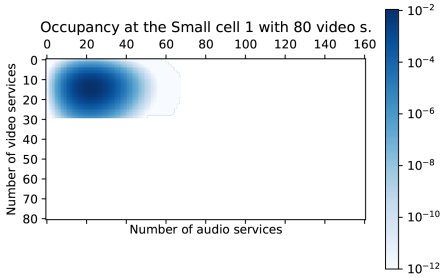
(d) Average number of services at the macro BS



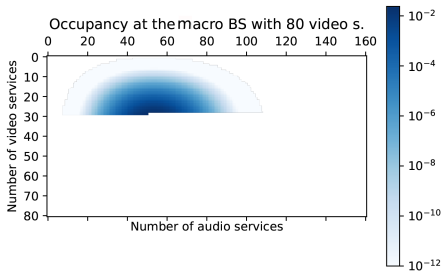
(e) Distribution of audio services at the stations



(f) Distribution of video services at the stations



(g) Distribution of streaming services at the small cells 1-3



(h) Distribution of streaming services at the macro BS

Fig. 3. Numerical results for the case of one macro cell BS with user density 1 and three small cell BSs with user density 2.

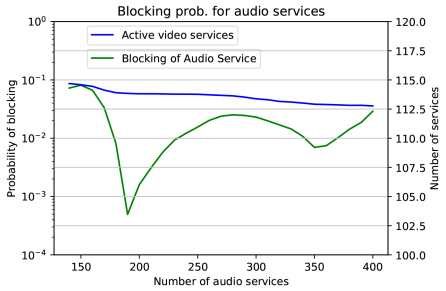


Fig. 4. Audio blocking probability and average number of active video services in the system versus the number of audio and elastic users, for a fixed number of video users.

video services should not exceed 60; this is easily implemented through an admission control algorithm).

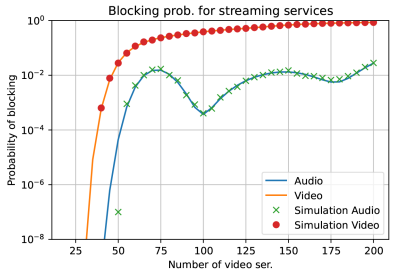
In Figure 3(b), we report analytical predictions and simulation results for the blocking probability of elastic services. We can see that the oscillations in blocking probability are present also in this case, and that the analytical model provides predictions which are pessimistic with respect to simulation. It is quite interesting to observe that the careful model we present in this article allows much better results to be obtained with respect to a "naive" approach that only considers the average data rate that remains for elastic services after the allocation to streaming services. In this case, analytical predictions (green dots in Figure 3(b)) come close to simulation only when blocking probabilities approach 10% and are unreasonably optimistic otherwise.

Figure 3(c) reports the residual data rate as a function of the number of video streaming customers in the network of queues at the macro BS (red curve) and at the three small cell BSs (the curves overlap because the cells are assumed to have the same size and the same user density). We see that in this case the macro BS carries more load deriving from streaming services, which is due to the fact that its area is six times larger.

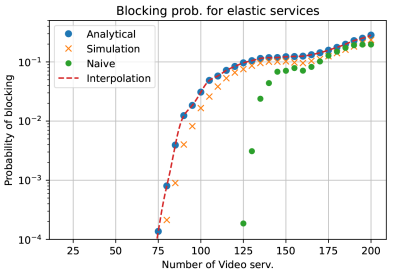
Figure 3(d) shows the average numbers of active audio, video, and elastic services at the macro BS as well as in each small cell. It is important to note the oscillation in the number of active elastic services, as well as the very good match between simulation results and results obtained with perfect sampling.

Figure 3(e) and (f) report the probability density functions of the numbers of audio and video services in progress at the four BSs, as well as the number of customers that are experiencing a pause between a service and the next one, in the case in which 80 video customers, 160 audio, and 160 elastic customers are present in the queuing network. Again, because of symmetry, the curves of the three small cell BSs overlap. We can see that the number of active video services at the macro cell BS is with high probability close to 29, which is the maximum possible number (remember that each BS can allocate up to 295 Mb/s to streaming services, and each video service requires 10 Mb/s). The spike in the distribution of the number of audio services in progress at the macro BS for 50 audio services is quite interesting. This value corresponds to the allocation of the remaining 5 Mb/s (of the 300 available, after 290 have been allocated to video and 5 are reserved to elastic) to audio services. The same phenomenon is also visible in Figure 3(g) and (h), which represent through heat maps the joint probability distributions of the numbers of active video and audio services at small cells (equal for the three small cells because of symmetry) and at the macro. We can see that the probability distribution at the macro peaks around the discontinuity at 29 videos and 50 audio.

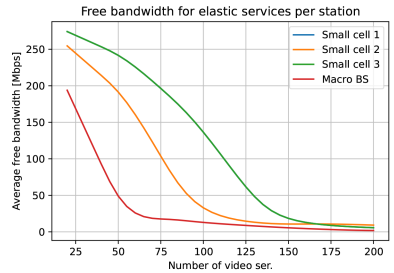
To further visualize the presence of oscillations in blocking probabilities, we plot in Figure 4 the total blocking probability in the four cells for audio services in the case of a constant number of



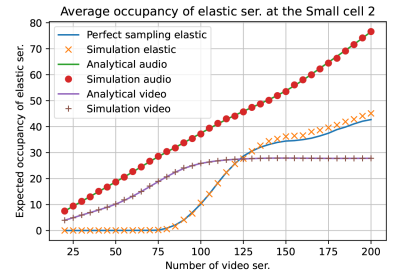
(a) Blocking probabilities for streaming services



(b) Blocking probabilities for elastic services



(c) Available data rate for elastic services



(d) Average number of jobs at the small cell 2

Fig. 5. Numerical results for the case of one macro cell BS with user density 1 and three small cell BSs with user density 1, 2, and 1.

video users, for a growing number of audio and elastic service users. The number of video users is fixed at 200, whereas the number of audio and elastic users grows from 140 to 400. In the figure, we also plot the average number of active video users, which starts at about 114 (the maximum possible number is 116, i.e., 29 in each one of the four cells) and decreases for increasing audio and elastic services load, due to increased blocking. Oscillations are due to the fact that the reduction of the number of active video services of one unit in a cell frees 10 Mb/s, which are sufficient to accept 100 audio services. Indeed, we observe slightly steeper decreases in the number of active videos in proximity of the minima of the audio blocking probability.

To see what happens when the three small cells are not symmetric, we also present results for the case in which the macro cell, as well as small cells 1 and 3, have user density equal to 1, and only small cell 2 has user density equal to 2.

This leads to handover probabilities from cell i to cell j with the following values:

$$\mathbf{P^M} = [P^M_{i,j}] = \begin{bmatrix} 0 & 1/5 & 3/5 & 1/5 \\ 3/4 & 0 & 1/4 & 0 \\ 1/2 & 1/4 & 0 & 1/4 \\ 3/4 & 0 & 1/4 & 0 \end{bmatrix} \tag{15}$$

and to dwell times with an average equal to 600 seconds for all cells.

Results for this case are shown in Figure 5. In Figure 5(a) and (b), we report blocking probabilities for streaming and elastic services, and we again see the oscillations already observed in Figure 3(a), as well as the superior accuracy of our modeling approach with respect to the naive analysis. In Figure 5(c), we see that in this case the residual data rate is lower for small cell 2 with respect to small cells 1 and 3, due to the higher user density. Finally, in Figure 5(d), we report the average numbers of active audio, video, and elastic services at small cell 2. Also in this

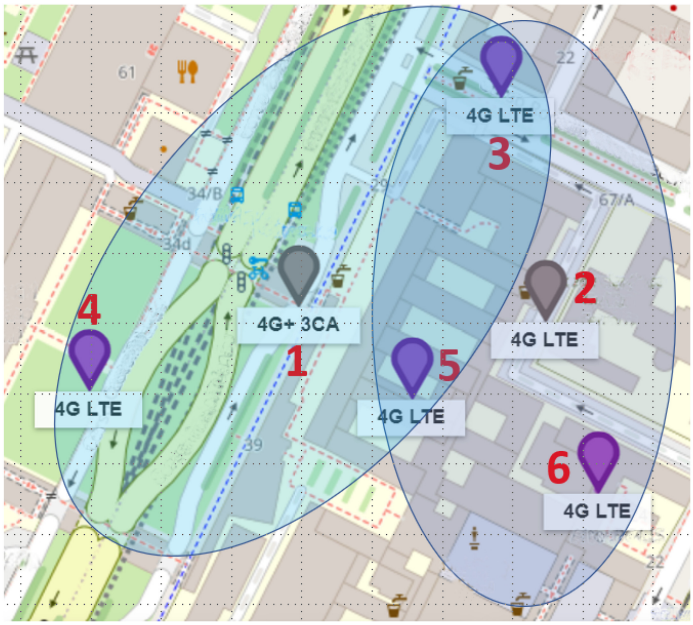


Fig. 6. Layout of the BSs covering a university campus. Macro cell BSs are in positions 1 and 2; small cell BSs are in positions 3, 4, 5, and 6.

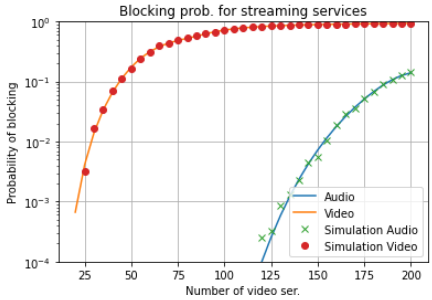
case, the match between simulation results and results obtained with perfect sampling is very good.

5.2 A University Campus

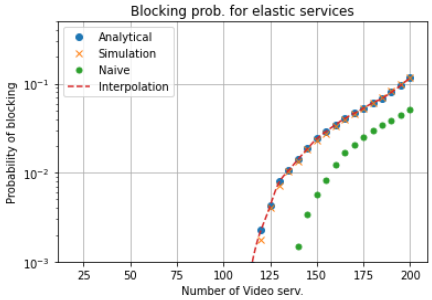
Our second numerical example considers a real deployment of BSs over a university campus. Two macro cell BSs and four small cell BSs provide services over the campus area and are positioned as shown in the map of Figure 6 (taken from the website lteitaly.it that reports BS positions in Italy), in which gray and purple markers represent macro cells and small cells, respectively. The macro cell BS in position 1 (called *macro BS 1*) has within its coverage area the small cell BSs in positions 3, 4, and 5 (referred to as small BS 3, 4 and 5). The small BS 5 is also within the coverage area of macro BS 2, and so are small BSs 3 and 6.

The parameters that characterize the system are the same as for the previous experiments, except for the fact that macro BS 1 has a capacity of 300 Mb/s and all other BSs have a capacity of 100 Mb/s (with 5 Mb/s reserved to elastic services at all BSs), the average size of the files to be transmitted by elastic services is 100 kb, the average think time for elastic services is 500 ms, and dwell times in small BSs 3, 4, 5, and 6 are, respectively, 702, 583, 400, and 654 seconds. The handover probabilities (for streaming services) from cell i to cell j have the following values:

$$\mathbf{P^M} = [P^M_{i,j}] = \begin{bmatrix} 0.0 & 0.1 & 0.187 & 0.613 & 0.1 & 0.0 \\ 0.1 & 0.0 & 0.319 & 0.0 & 0.1 & 0.481 \\ 0.465 & 0.516 & 0.0 & 0.0 & 0.019 & 0.0 \\ 0.284 & 0.0 & 0.0 & 0.0 & 0.716 & 0.0 \\ 0.1 & 0.1 & 0.29 & 0.324 & 0.0 & 0.186 \\ 0.0 & 0.349 & 0.0 & 0.0 & 0.651 & 0.0 \end{bmatrix}. \tag{16}$$

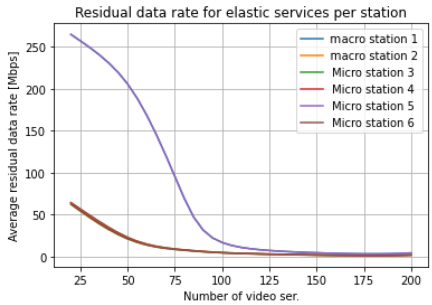


(a) Blocking probabilities for streaming services

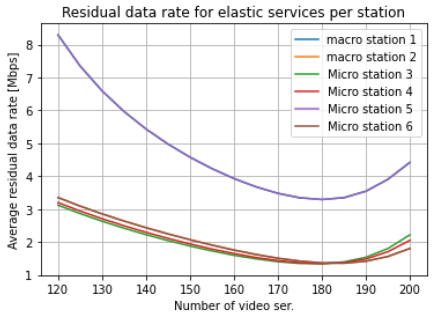


(b) Blocking probabilities for elastic services

Fig. 7. Blocking probability for streaming and elastic services versus the number of video users in the real BS layout. In the system, for each video user, we also have two audio users and two elastic users.



(a) 15–200 video users



(b) 150–200 video users

Fig. 8. Average residual data rate at all BSs in the university campus scenario: plot (b) shows a zoom on the tail of plot (a).

The average residence times and the routing probabilities have been estimated taking into account the sizes of the areas covered by the cells and by balancing the mobility flows of users among these areas. To obtain a closed system, we have assumed that users move among the considered cells neglecting incoming (departure) flows from (to) the outside [1].

In this setting, the blocking probabilities for streaming and elastic services are reported in Figure 7 as a function of the number of video users in the system (for each video user, there are also 2 audio users and 2 elastic users in the system; hence, with 200 video users, the total number of users in the system is 1,000). In Figure 7(b), which shows the blocking probabilities for elastic services, blue and green dots represent the analytical results of our model and the naive model, respectively, whereas orange crosses correspond to simulation results. The curves once more prove the accuracy of our approach and its superiority to the naive approach based on the use of the average residual data rate for the computation of the blocking probability of elastic services.

The average residual data rate for elastic services for all BSs is presented in Figure 8. Macro BS 1 exhibits higher values of the average residual data rate with respect to all other BSs because it has three times more capacity. All other BSs have equal capacity and very similar values of average residual data rate. The plot on the right zooms in the behavior with large numbers of users (from 600 to 1,000 in total), allowing us to see that the data rate available to elastic services (in addition to the reserved 5 Mb/s) is significantly larger at macro BS 1 than at all other BSs. For example, with 200 video users (1,000 total users), the data rate available to elastic services is about 7 Mb/s

(5 reserved plus about 2, as we see on the plot) at all BSs, except macro BS 1, where it is about 9.5 Mb/s.

The average number of services in progress at each one of the small cell BSs is reported in Figure 10. The behavior of the four small cell BSs is very similar, as expected, since differences are only due to (small) asymmetries in routing. The maximum average number of simultaneously active video services is 9, thus consuming 90 of the available 100 Mb/s, so that 5 Mb/s remain for audios, in addition to the 5 Mb/s reserved for elastic services. The average number of simultaneously active audio services grows up to 50, thus bringing the data rate allocated to streaming services to 95 Mb/s, hence respecting the reservation of 5 Mb/s for elastic services. The average number of elastic services in progress is very low when the number of users is small, because the residual data rate is high, so that each elastic service is allocated a high data rate and thus completes very quickly (transferring 1 Mb at 5 Mb/s requires 200 ms only). Instead, when streaming services consume a large portion of the data rate available at the BS (we can see in Figure 8 that this begins to happen for a number of video users between 75 and 100), the average residual data rate becomes small, the duration of each elastic service grows, and so does the average number of elastic services in progress. Note that the average number of simultaneous elastic services remains far from the maximum value permitted, namely 50.

The average number of services in progress at each one of the macro cell BSs is reported in Figure 11. The results for the two macro BSs are obviously different, since macro BS 1 has a data rate three times higher than macro BS 2. For this reason, we see that in macro BS 1, the average number of simultaneously active video services grows up to 29, hence consuming 290 Mb/s out of the 295 available for streaming services. On the contrary, at macro BS 2, the number of active video services grows only up to 9, hence consuming 90 Mb/s out of the 95 available. In both BSs, the average number of simultaneously active audio services grows up to 50, consuming the 5 Mb/s that are not used by video and not reserved to elastic services.

The average number of simultaneously active elastic services, like in the previous plots, is very small before the available data rate saturates, and also in this case does not reach the maximum value of 50.

Quite interesting is the fact that if we increase the average file size to be transferred by elastic services to 1 Mb, and we simultaneously increase the average time between the end of an elastic service and the issue of a new request to 5 seconds (the same parameters we used for the case of one macro cell BS and three small cell BSs in Section 5.1), we obtain very accurate results for all cells, except macro cell BS 1, whose results are reported in Figure 12. In this figure, we can observe a significant discrepancy between the average number of simultaneously active elastic services predicted by our model and the number computed by detailed simulations. The reason of this discrepancy is in the transient behavior in the dynamics of elastic services between state changes of streaming services. Indeed, our model computes the steady state distribution of the number of elastic services for every configuration of streaming services and averages the results. This implies that our model is oblivious to the transients in the number of elastic services generated at every change of state of streaming services. Consider, for example, a change of state induced by the termination of one video service. When this happens, all of a sudden 10 Mb/s become available, and are immediately exploited by elastic services, whose number goes down very fast, toward the steady state associated with the new value of available data rate. Conversely, when a new video service starts, the available data rate is reduced by 10 Mb/s, but the number of active elastic services only grows toward the new steady state value with the dynamics of their arrival rate combined with the new (slower) service rate. This is what we see in Figure 9 that reports the average of 1,000 sample paths of the number of elastic services in progress when the number of active video services goes down from 9 to 8 and then back to 9 at BS 2 of our campus setting (note that 9 is

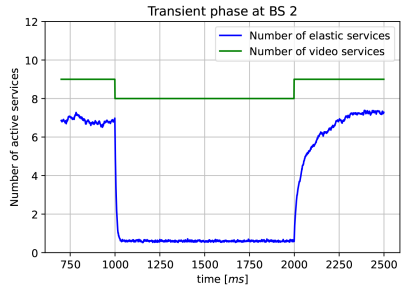
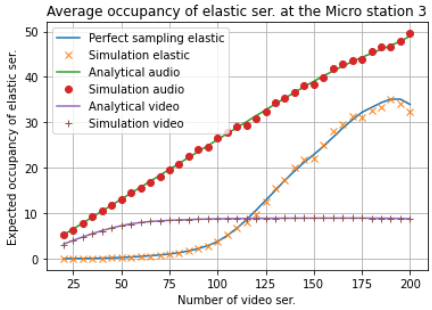
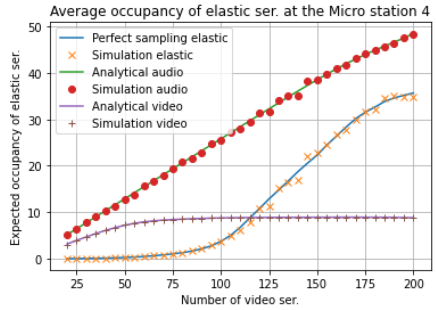


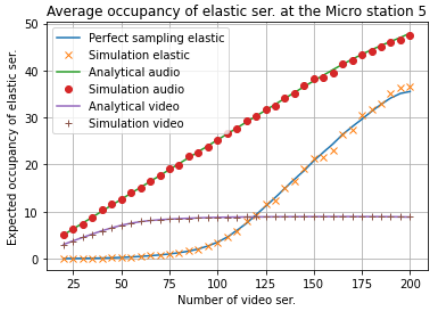
Fig. 9. Example of a transient at BS 2.



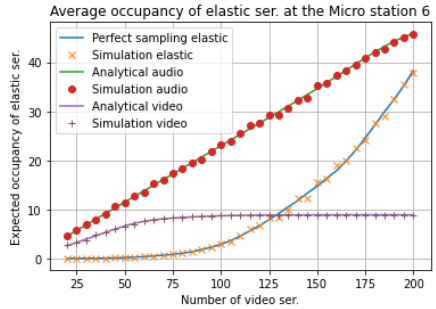
(a) Small cell base station 3



(b) Small cell base station 4



(c) Small cell base station 5



(d) Small cell base station 6

Fig. 10. Average number of streaming and elastic services active at each of the small cell BSs in the real BS layout.

the maximum number of videos that can be accepted at the BS, since the BS capacity is 100 Mb/s, 5 Mb/s are reserved to elastic services, and each video requires 10 Mb/s). When a video ends, the number of elastic services starts from its steady state value (about 7) and decreases very fast to about 1 (the transient lasts few tens of seconds). When a new video starts, the number of elastic services starts growing, but it takes about 300 seconds to return to the steady state value.

This shows that the reaction of the number of elastic services to a video completion is faster than the reaction to a video start so that the real average number of simultaneously active elastic services results lower than what our model predicts, as we see in Figure 11. A confirmation of this effect is given by the fact that considering the faster dynamics of elastic services that we used to generate the results in Figure 12, the inaccuracy becomes negligible. The reason the inaccuracy is

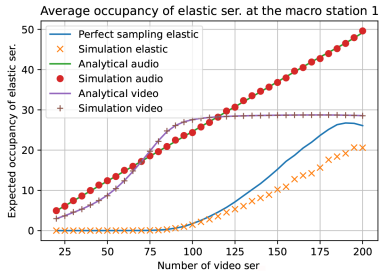
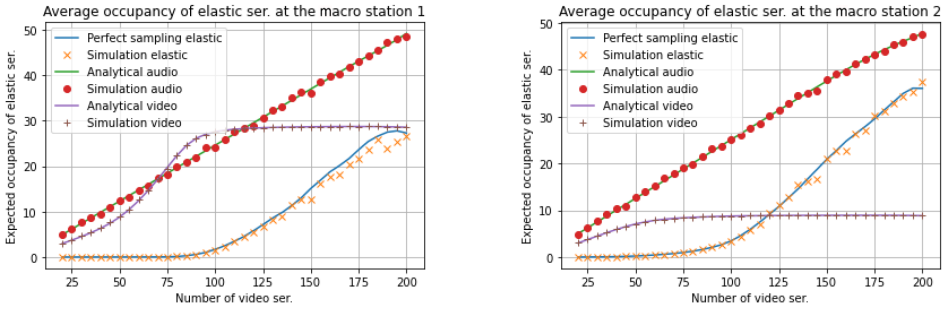


Fig. 11. Average number of streaming and elastic services active at the macro cell BS 1 in the real BS layout with 1-Mb file size for elastic services.



(a) Macro cell base station 1

(b) Macro cell base station 2

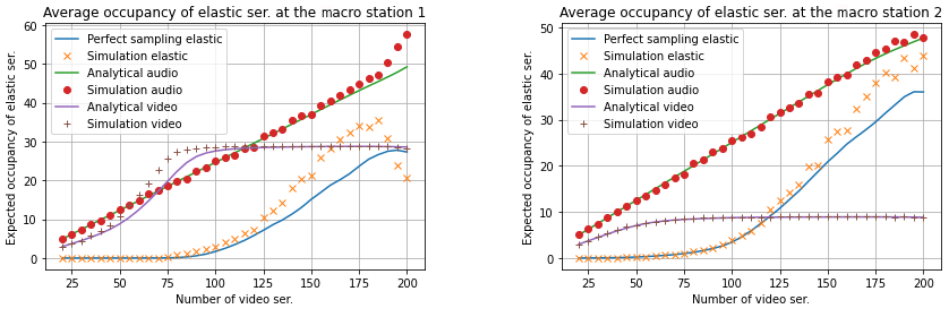
Fig. 12. Average number of streaming and elastic services active at each of the macro cell BSs in the real BS layout.

visible only for macro cell BS 1 is in its higher data rate with respect to all other BSs (300 Mb/s instead of 100). This leads to the possibility of 29 simultaneous active video services (instead of 9), which makes the dynamics of the number of active video services over three times faster for this BS, hence generating a larger number of transients for elastic services.

5.3 Comparison with a Routing Policy without Skipping

Up to now, we discussed the accuracy of the analytical model by comparing against simulations that implement the same routing that is necessary to obtain the product form solution for the network of queues that describes the dynamics of streaming services—that is, the skipping routing policy. Indeed, we observed that the extremely good match between the analytical and simulation results for streaming services is a validation of the simulator rather than the model, whereas the very good accuracy of the results for elastic services is an indication of the accuracy of the approximations introduced in their analytical model solution. However, the actual behavior of service requests in RANs does not follow the skipping policy, so it is important to assess the impact of such approximation with respect to results obtained for realistic service request routing. In other words, up to now, we discussed the validity of the approximations introduced in the model *solution*, whereas now we will discuss the impact of the approximations introduced in the model *construction*.

We have already commented on the fact that the skipping policy may or may not closely reflect the behavior of a real system, because of the need to have the same probabilities of moving to another queue in the case of mobility/completion and in the case of blocking. The main discrepancy



(a) Macro cell base station 1

(b) Macro cell base station 2

Fig. 13. Average number of streaming and elastic services active at each of the macro cell BSs in the real BS layout; model results with skipping and simulation results without skipping.

between model and real system may reside in the equality of the probabilities to return to the delay station: whereas in the model a service request may be blocked even if some of the neighboring cells have resources to accept it, in a real system blocking occurs only when all of the cells that can serve the terminal are congested. To verify the impact of this discrepancy, we present simulation results of the system behavior assuming that blocking occurs only when the resources of all cells in the modeled group are congested.

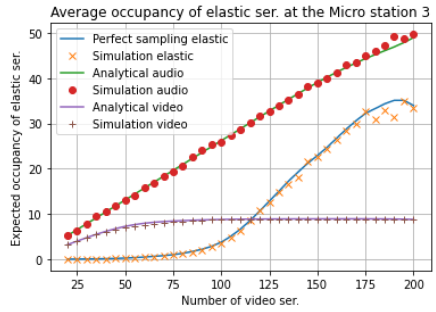
In Figure 13, we show the average number of video, audio, and elastic services simultaneously in progress at the macro cell BS 1 and 2 as functions of the number of video users in the system. In Figure 14, we report the same quantities for the four small cell BSs. Results show that the model provides acceptable estimates, especially in the regions before saturation of resources—that is, in regions where the system is expected to operate most of the time.

Blocking probabilities for streaming and elastic service requests are presented in Figure 15. We can see that the skipping routing policy leads to pessimistic estimates for the service request blocking probability, as can be expected by considering that skipping implies a possibility of blocking at each visited BS, whereas the implemented routing produces blocking only when all considered BSs have no available resources. The fact that the efficient solution technique leads to pessimistic estimates is quite positive for applications in network planning and design, since it provides conservative estimates, and leads to some overdimensioning, as is customary in the networking domain.

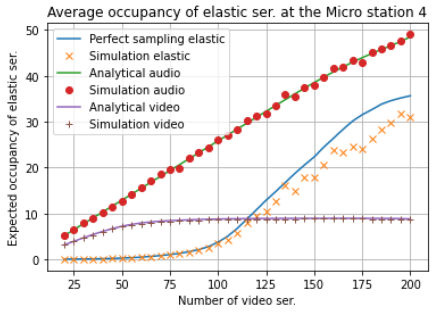
6 STRENGTHS AND LIMITATIONS OF THE APPROACH

The study by simulation and the exact analysis of multiservice RANs is extremely complicated. The cardinality of the system state space makes a direct solution of the underlying stochastic process unfeasible even for quite small networks. As discussed in Section 4, approximate methods are available to study Markov modulated processes such as those used in other works [16, 19], but they are inapplicable to our models because of the extremely large state space of the modulating process—that is, the network of streaming services.

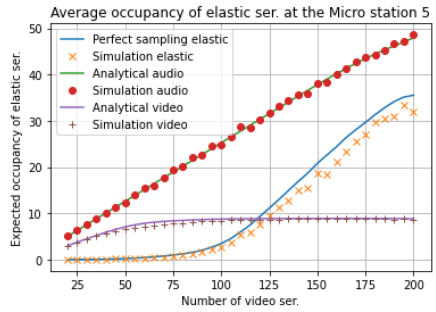
One simple way to overcome these problems could be to study the network of elastic services conditioned to the *average* residual data rate not used by streaming services. This method has been compared to ours in Section 5 under the name *naive*. Unfortunately, despite its low complexity, this approach provides extremely inaccurate and optimistic estimates of the blocking probabilities. This is due to the non-linearity of the blocking probabilities as function of the residual data rate. Indeed, blocking probabilities tend to grow only when the BS is close to saturation, thus averaging the residual data rate leads to very optimistic estimations of the performance.



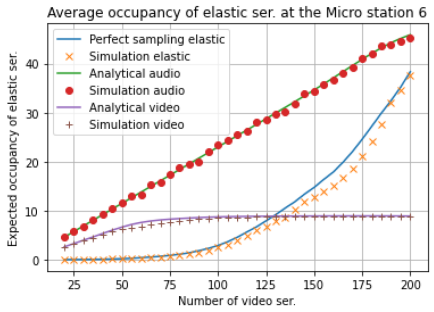
(a) Small cell base station 3



(b) Small cell base station 4

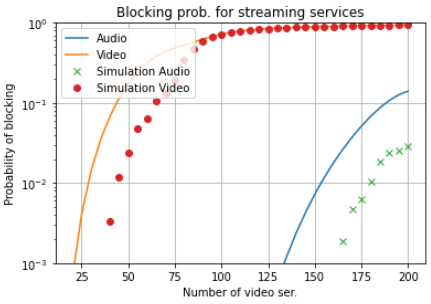


(c) Small cell base station 5

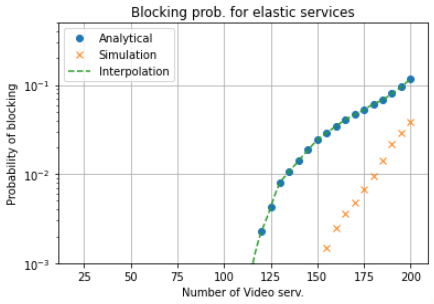


(d) Small cell base station 6

Fig. 14. Average number of streaming and elastic services active at each of the small cell BSs in the real BS layout; model results with skipping and simulation results without skipping.



(a) Blocking probabilities for streaming services



(b) Blocking probabilities for elastic services

Fig. 15. Blocking probability for streaming and elastic services versus the number of video users in the real BS layout; model results with skipping and simulation results without skipping.

Simulation is another tool that could be used to study multiservice RANs, and, indeed, we have resorted to this approach to validate our solution method. However, discrete event simulations have serious problems for realistic scenarios. The most important is the fact that the dynamics of elastic services are much faster than those of streaming services, and this implies that most of the events of the simulation are devoted to handle elastic services, making the time required to skip the transient phase quite long. Moreover, an accurate estimation of the blocking probability in a moderate load for audio services (e.g., $\sim 10^{-4}$) requires to process millions of events even with a small number of cells.

It is interesting to observe that while the problem of the different speeds of the dynamics of elastic and streaming services undermines the applicability of discrete event simulations, it is the characteristic that allows our approach to work properly. This makes the two methods somehow complementary.

Finally, observe that the procedure proposed to perfectly sample a streaming network state can also be used to randomly choose the initial state of the simulation model, thus reducing the duration of the simulation transient to that required by elastic services to reach their stationary behavior.

The proposed approach has been shown to work for networks consisting of seven stations and 1,000 users (400 audio, 200 video, and 400 elastic), which would be unfeasible with other approaches since the resulting models have state space cardinality larger than $O(10^{20})$. Notice from the asymptotic complexity of the convolution algorithm that the number of stations can grow to a dozen without creating numerical issues, whereas higher numbers of customers and classes negatively affect the applicability of the method.

7 CONCLUSION

In this article, we presented the first tractable analytical model for the performance evaluation of a group of BSs of a RAN offering a mix of elastic and streaming services. The model is based on a closed network of queues, in which customers represent service instances that roam over the area served by the BSs, until either service completion or blocking. The queuing network model is exact for what concerns streaming customers and provides an approximation for the behavior of elastic customers. The model can be solved in product form under some specific conditions concerning customer routing, and in addition exhibits insensitivity to service distributions, provided that mobility can be neglected.

Numerical results showed quite good accuracy of the elastic services model, and highlighted unexpected oscillating behaviors that were already observed in the literature in the case of just one BS but were described in this article for the first time in the case of a group of BSs.

Capturing such behaviors in an analytical model is very important, as it allows an MNO to understand the dynamics of its network, and to correct undesired behaviors with simple admission control algorithms and/or traffic management approaches.

ACKNOWLEDGMENTS

This paper was supported in part by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on “Telecommunications of the Future” (PE00000001 - program “RESTART”, projects R4R and ITA NTN). We would also like to thank GNCS-INdAM for the support to this research.

REFERENCES

[1] M. Ajmone Marsan, M. Meo, and M. Sereno. 2021. Modeling simple HetNet configurations with mixed traffic loads. In *Proceedings of the 22nd IEEE International Symposium on a World of Wireless, Mobile, and Multimedia Networks (WoWMoM'21)*. 119–128.

[2] S. Balsamo, P. G. Harrison, and A. Marin. 2010. A unifying approach to product-forms in networks with finite capacity constraints. *ACM SIGMETRICS Performance Evaluation Review* 38, 1 (2010), 25–36.

[3] G. P. Basharin and T. V. Aterekova. 2010. Analytical model of streaming and elastic traffic with dynamic channel allocation scheme. In *Proceedings of the International Congress on Ultra Modern Telecommunications and Control Systems*. 1086–1090.

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

[5] N. Benameur, S. B. Fredj, F. Delcoigne, S. Oueslati-Boulahia, and J. Roberts. 2001. Integrated admission control for streaming and elastic traffic. In *Quality of Future Internet Services*. Lecture Notes in Computer Science, Vol. 2156. Springer, 69–81.
[6] E. Bernal-Mor, V. Pla, and J. Martinez-Bauset. 2010. Robust admission control for streaming and elastic services in cellular networks. In *Proceedings of the IEEE Symposium on Computers and Communications*. 372–374.
[7] J. Blanchet and X. Chen. 2019. Perfect sampling of generalized Jackson networks. *Mathematics of Operations Research* 44, 2 (2019), 1–32.
[8] S. Borst and N. Hegde. 2007. Integration of streaming and elastic traffic in wireless networks. In *Proceedings of the 26th IEEE International Conference on Computer Communications (IEEE INFOCOM'07)*. 1884–1892.
[9] O. J. Boxma, A. Bumb, R. Nunez-Queija, and H.-P. Tan. 2005. *Integration of Streaming and Elastic Traffic in a Single UMTS Cell: Modeling and Performance Analysis*. Technical Report. EURANDOM.
[10] O. J. Boxma, A. F. Gabor, R. Nunez-Queija, and H.-P. Tan. 2006. Performance analysis of admission control for integrated services with minimum rate guarantees. In *Proceedings of the 2nd Conference on Next Generation Internet Design and Engineering (NGI'06)*. 7–47.
[11] S. C. Bruell and G. Balbo. 1980. *Computational Algorithms for Closed Queueing Networks*. North Holland.
[12] J. P. Buzen. 1973. Computational algorithms for closed queueing networks with exponential servers. *Communications of the ACM* 16, 9 (1973), 527–531.
[13] X. Chao, M. Miyazawa, and M. Pinedo. 1999. *Queueing Networks—Customers, Signals, and Product Form Solutions*. John Wiley & Sons.
[14] A. E. Conway and N. D. Georganas. 1986. RECAL—A new efficient algorithm for the exact analysis of multiple-chain closed queuing networks. *Journal of the ACM* 33, 4 (Aug. 1986), 768–791.
[15] D. Garcıa, J. Martınez, and V. Pla. 2005. Admission control policies in multiservice cellular networks: Optimum configuration and sensitivity. In *Wireless Systems and Mobility in Next Generation Internet*. Lecture Notes in Computer Science, Vol. 3427. Springer, 121–135.
[16] E. Gelenbe and C. Rosenberg. 1990. Queues with slowly varying arrival and service processes. *Management Science* 36, 8 (1990), 928–937.
[17] S. Hanczewski, M. Stasiak, and J. Weissenberg. 2021. A model of a system with stream and elastic traffic. *IEEE Access* 9 (2021), 7789–7796.
[18] Andrea Marin, Marco Ajmone Marsan, Michela Meo, and Matteo Sereno. 2024. Queuing models of links carrying streaming and elastic services. *Computer Networks* 244 (2024), 110306.
[19] I. Mitrani. 2002. Spectral expansion solutions for Markov-modulated queues. In *Network Performance Engineering*. Lecture Notes in Computer Science, Vol. 5233. Springer, 423–446.
[20] B. G. Pittel. 1979. Closed exponential networks of queues with saturation: The Jackson-type stationary distribution and its asymptotic analysis. *Mathematics of Operations Research* 4, 4 (1979), 357–378.
[21] J. G. Propp and D. B. Wilson. 1996. Exact sampling with coupled Markov chains and applications to statistical mechanics. In *Proceedings of the International Conference on Random Structures and Algorithms*. 223–252.
[22] R. Ramjee, R. Nagarajan, and D. Towsley. 1997. On optimal call admission control in cellular networks. *Wireless Networks Journal* 3, 1 (1997), 29–41.
[23] W. Song and W. Zhuang. 2007. Multi-class resource management in a cellular/WLAN integrated network. In *Proceedings of the IEEE Wireless Communications and Networking Conference*. 3070–3075.
[24] W. Song and W. Zhuang. 2007. Resource allocation for conversational, streaming, and interactive services in Cellular/WLAN interworking. In *Proceedings of the IEEE Global Telecommunications Conference (IEEE GLOBECOM'07)*. 4785–4789.
[25] J. van der Gaast, R. B. M. de Koster, I. J. B. F. Adan, and J. A. C. Resing. 2020. Capacity analysis of sequential zone picking systems. *Operations Research* 68, 1 (2020), 161–179.

Received 15 June 2023; revised 17 January 2024; accepted 12 February 2024