Statistical Multiplexing Gain Analysis of Heterogeneous Virtual Base Station Pools in Cloud Radio Access Networks

Jingchu Liu, *Student Member, IEEE*, Sheng Zhou, *Member, IEEE*, Jie Gong, *Member, IEEE*, Zhisheng Niu, *Fellow, IEEE*, and Shugong Xu, *Fellow, IEEE*

***Abstract*— Cloud radio access network (C-RAN) was proposed recently to reduce network cost, enable cooperative communications, and increase system flexibility through centralized baseband processing. By pooling multiple virtual base stations (VBSs) and consolidating their stochastic computational tasks, the overall computational resource can be reduced, achieving the so-called statistical multiplexing gain. In this paper, we evaluate the statistical multiplexing gain of VBS pools using a multi-dimensional Markov model, which captures the session-level dynamics and the constraints imposed by both radio and computational resources. Based on this model, we derive a recursive formula for the blocking probability and also a closed-form approximation for it in large pools. These formulas are then used to derive the session-level statistical multiplexing gain of both real-time and delay-tolerant traffic. Numerical results show that VBS pools can achieve more than 75% of the maximum pooling gain with 50 VBSs, but further convergence to the upper bound (large-pool limit) is slow because of the quickly diminishing marginal pooling gain, which is inversely proportional to a factor between the one-half and three-fourth power of the pool size. We also find that the pooling gain is more evident under light traffic load and stringent quality of service requirement.**

***Index Terms*— C-RAN, VBS pooling, statistical multiplexing.**

I. Introduction

IN RECENT years, the proliferation of mobile devices such as smart phones and tablets, together with the diverse applications enabled by mobile Internet, has triggered the exponential growth of mobile data traffic [1]. To accommodate the rapid traffic growth, cellular networks have been continuously evolving toward smaller cell size, wider bandwidth, and more advanced transmission technologies. However, the problems that arise, such as the increased interference and operational cost, are difficult to be solved via the traditional radio access network (RAN) architecture, in which the processing functionalities are packed into stand-alone base stations (BSs) and the cooperation between BSs is restricted by the limited inter-BS backhaul bandwidth.

To overcome the shortcomings of the traditional RAN architecture, cloud RAN (C-RAN) [2] is proposed with centralized baseband processing. C-RAN can facilitate the adoption of cooperative signal processing and potentially reduce the operational cost. A similar idea is also proposed under with the name wireless network cloud (WNC) [3]. This kind of novel architecture has attracted substantial attentions recently: the key building blocks of C-RAN are investigated and its major use cases are identified [4]–[7]. Centralized processing is combined with dynamical fronthaul switching to address the mobility and energy efficiency issues of small cells in [8] and [9]. Concerning realization-related issues, it is demonstrated in [10]–[14] that BBU functionalities can be implemented as software, i.e. virtual base station (VBS), which runs on general purpose platforms (GPP). Compared with specialized-platform-based implementations, GPP-based implementation is more flexible in terms of the implementation of new functionalities and the management of computational resource. Further more, a VBS pool can be constructed by consolidating multiple VBSs to share the same set of computational resource. In this way, the computational resource can be utilized more efficiently and related cost can be reduced.

Despite the evident advantages of C-RAN, the massive bandwidth requirement of its fronthaul network poses a serious challenge: transmitting the baseband sample of a single 20MHz LTE antenna-carrier (AxC) requires around 1Gbps link bandwidth [15], [16]. Large-scale centralization will thus incur enormous fronthaul expenditure and potentially cancel out the gains. Fortunately, it is observed in [11] and [17] that substantial statistical multiplexing gain can be obtained even with small-scale centralization, justifying the deployment of small clusters of C-RAN. Yet, these results are obtained from simulations that are based on short-term small-scale traffic logs, and a generalized analytical model is in need to derive the optimal VBS cluster size. To this end, a session-level VBS pool model is proposed in [18] under the assumption of unconstrained

Manuscript received August 19, 2015; revised February 29, 2016; accepted May 2, 2016. Date of publication May 12, 2016; date of current version August 10, 2016. This work was supported in part by the National Basic Research Program of China (973 Program) under Grant 2012CB316001, in part by the National Science Foundation of China under Grant 61201191, Grant 61321061, Grant 61401250, and Grant 61461136004, and in part by the Intel Collaborative Research Institute for Mobile Networking and Computing. This paper was presented at the IEEE Global Communications Conference 2014. The associate editor coordinating the review of this paper and approving it for publication was N. B. Mehta.

J. Liu, S. Zhou, and Z. Niu are with the Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: liu-jc12@mails.tsinghua.edu.cn; sheng.zhou@tsinghua.edu.cn; niuzhs@tsinghua.edu.cn).

J. Gong was with the Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China. He is now with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China (e-mail: gongj26@mail.sysu.edu.cn).

S. Xu is with Intel Corporation, Beijing 100090, China (e-mail: shugong.xu@intel.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2016.2567383

radio resource and dynamic resource management. Here user sessions represent the time period in which users occupy computational resource in the pool. Nevertheless, this model does not reflect two realistic factors. Firstly, radio resource are often the main performance bottleneck for real networks, and thus the influence of radio resource should be reflected in the VBS pool model. Secondly, dynamic resource management, which re-assigns resources at the arrival and departure of each user session, may incur too much control overhead and overload the system [11]. Hence, semi-dynamic resource management, which assigns resources on much larger time scales (e.g. hours to days) than the arrival and departure of user sessions (e.g. seconds to minutes), is more realistic. This assumption is also reasonable because the traffic statistics also vary in similarly large time scales, therefore the management plan designed for some traffic statistics can be useful for a fairly long period, and only need to be occasionally adjusted in the long run.

In our previous work [19], we analyze the statistical multiplexing gain of homogeneous VBS pools, in which each VBS has the same traffic arrival, resource configuration, and service strategy. We derive a product-form expression for the stationary distribution of user sessions in each VBS and give a recursive method to compute the session blocking probability of the VBS pool. In this paper, we extend these results to heterogeneous VBS pools, in which there are multiple classes of VBSs with different session arrival, resource configurations, and service strategies. The computational complexity of the recursive method is also analyzed. Under the assumption of large VBS pools, we also derive a closed-form approximation for the blocking probability. We show through simulation that the approximation is precise even for medium-sized pools with around 50 VBSs. We then use this approximation to quantitatively investigate the statistical multiplexing gain of VBS pools under the influence of different different factors, including pool sizes, VBS heterogeneity, traffic load, and the desired levels of QoS.

The main contributions of this paper are as follows:

- We propose a realistic session-level model for heterogeneous VBS pools with both radio and computational resource constraints and semi-dynamic resource management. We show that this model constitutes a continuous-time multi-dimensional Markov chain and derive its product-form stationary distribution. We also illustrate how this model can be used to analyze real-time and delay-tolerant traffics.
- We give a recursive method to compute the blocking probability for the proposed model. This method has quadratic computational complexity, much lower than brute-force evaluation which has exponential complexity. For large VBS pools, we also derive a closed-form formula to approximate the blocking probability.
- We provide an in-depth analysis on the statistical multiplexing gain of VBS pools. We show numerically the influence of various factors including the pool size, traffic load, and QoS requirements. We also prove that the statistical multiplexing gain increases slowly as the pool size grows large, with the residual gain diminishing at a speed between $|\boldsymbol{M}|^{-3/4}$ and $|\boldsymbol{M}|^{-1/2}$. Here $|\boldsymbol{M}|$ denotes the pool size.

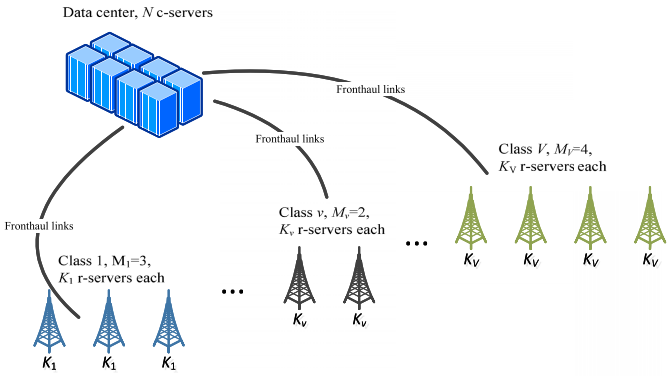

Fig. 1. An example of heterogeneous VBS pool. There are V classes of VBSs, each class may have different number of VBSs and amount of radio resource. These VBSs are consolidated in a data center and shares N units of computational resource.

The rest of the paper is organized as follows. Section II introduces the proposed model and presents the product-form stationary distribution of user sessions. Section III derives the recursive formula for the blocking probability and gives a closed-form approximation for large VBS pools. In Section IV, we derive the expression of statistical multiplexing gains and apply it to both real-time and delay-tolerant traffics in Section V. Section VI presents the numerical results and discusses the implications on realistic system design. Finally the paper is concluded in section VII.

II. Model Formulation

In this section, we introduce the Markov model for VBS pools and derive its stationary distribution. The model captures the session-level dynamics in a VBS pool. To endow our model with enough generality, we assume V different classes of VBSs. The total number of VBSs in class v ($v = 1, 2, \cdots, V$) is M_v. Each VBS in class v is assigned with K_v units of radio resource. To perform baseband signal processing on user sessions, all VBSs share a total of N units of computational resource. The overall setting is illustrated in Fig. 1. We assume homogeneous resource demands: every active session is assumed to occupy one unit of radio resource and one unit of computational resource. Note that computational workloads that are independent of user dynamics do exist in cellular systems. However, the dominant consumers for computational resources are mostly per-user functions and thus the overall workload roughly follows a linear relationship with the number of users [11]. For simplicity, hereafter we denote radio and computational resource by r-servers and c-servers, respectively.

A. Session Arrival, Service Discipline, and Admission Control

User sessions are initiated following independent Poisson processes in the coverage area of their serving VBSs. Obviously, the overall session arrival rate in a VBS is proportional to its coverage area. We allow VBSs in different classes

to have different sizes of coverage areas, and consequently, different aggregated session arrival rates. We denote the arrival rate for class-v VBSs by λ_v.

Each user session demands an exponentially distributed amount of service capacity before it leaves. Note the notion of service capacity can be flexibly interpreted according to the specific scenario this model is applied to. For example, service capacity can be considered as time duration for voice call sessions or the amount of information bits for cellular data sessions. For statistical QoS scenarios such as soft-real-time video, service capacity can still be interpreted as duration or information bits. The physical resources that enables such capacity is defined as the minimum amount of resource that can constantly satisfy a session. This means if the instantaneous requirement of a session is lower than provided, the remaining resources will become under-utilized. We assume that a VBS pool scheduler manages the service capacity so that the service capacity assigned to a class-v VBS is a function of the total number of sessions in this VBS, and the assigned capacity is equally divided among these sessions for fairness Denote the number of sessions in the m-th VBS of class-v at time t as $U_{v,m}(t)$. Then the above service strategy can be translated into a session departure rate function $f_v(U_{v,m}(t))$ for sessions in the m-th VBS of class-v at time t. The above Poisson assumptions have been widely used in existing literature to evaluate the impact of randomness on system performance [20], [21].

To guarantee that active sessions always have enough r-servers and c-servers, the VBS pool has to enforce admission control on the arriving sessions: whenever a new session arrives, an admission control agent in the VBS pool will decide whether or not to accept this session according to the current resource usage. For class-v sessions, the new session is accepted only if the number of r-servers in its serving VBS is less than K_v and the number of c-servers in the pool is less than N; otherwise the session is blocked.

B. State Transitions

Recall that we denote the number of sessions in the m-th VBS of class-v by $U_{v,m}(t)$, we can further describe the session dynamics in the VBS pool with a continuous-time stochastic process

$$\boldsymbol{U}(t) = (U_{1,1}(t), \cdots, U_{1,M_1}, \cdots, U_{V,1}, \cdots, U_{V,M_V}(t))^T.$$

Given the Markovian property of the arrival and service of processes, it is obvious that $\boldsymbol{U}(t)$ is a Markov chain. Taking the admission control policy into consideration, we can get the set of possible system states

$$\begin{aligned}\boldsymbol{U}(t) \in \mathbb{U} = \{\boldsymbol{u} \mid & 0 \le u_{v,m} \le K_v, \\ & 0 \le \sum_{v=1}^{V}\sum_{m=1}^{M_v} u_{v,m} \le N, \\ & u_{v,m} \in \mathbb{N}\},\end{aligned} \tag{1}$$

where $\boldsymbol{u} = (u_{1,1}, \cdots, u_{1,M_1}, \cdots, u_{V,1}, \cdots, u_{V,M_V})^T$ is the state vector. Because the session arrivals and departures are Markovian, $\boldsymbol{U}(t)$ is a multi-dimensional birth-and-death process. The transition rate of $\boldsymbol{U}(t)$ from an arbitrary state $\boldsymbol{u^{(i)}}$ to another state $\boldsymbol{u^{(j)}}$ is:

$$q_{\boldsymbol{u}^{(i)}\boldsymbol{u}^{(j)}} = \begin{cases} \lambda_v, & \text{if } \boldsymbol{u^{(j)}} - \boldsymbol{u^{(i)}} = \boldsymbol{e_{v,m}} \\ f_v(u_{v,m}^{(i)}), & \text{if } \boldsymbol{u^{(j)}} - \boldsymbol{u^{(i)}} = -\boldsymbol{e_{v,m}} \\ 0, & \text{otherwise} \end{cases}, \tag{2}$$

where $u_{v,m}^{(i)}$ is the $(\sum_{w=1}^{v-1} M_w + m)$-th entry of $\boldsymbol{u^{(i)}}$, and

$$\boldsymbol{e_{v,m}} = (0, \cdots, 0, \underbrace{1}_{(\sum_{w=1}^{v-1} M_w + m)\text{-th}}, 0, \cdots, 0)^T$$

is a column vector of length $\sum_{v=1}^{V} M_v$. For the ease of understanding, we illustrate the state transition graph of a simple example with $V = 1$, $M_1 = 2$, $K_1 = 3$, $N = 4$ and $f_1(n) = n\mu_0$ in Fig. 2.

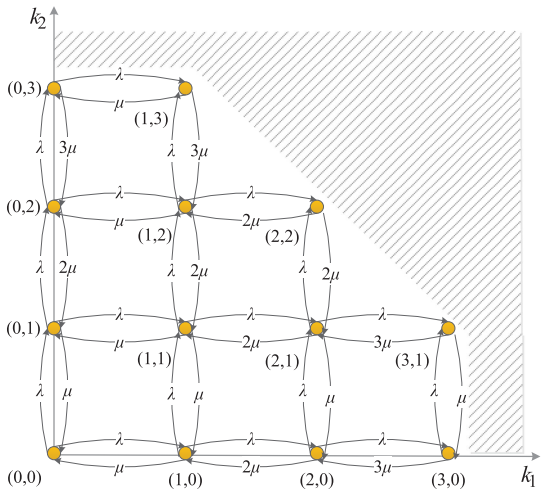

Fig. 2. Transition graph of a VBS pool with two VBSs. The k_1 and k_2 axes indicate the number of active sessions in these two VBSs, respectively. Each yellow point represents a possible pool state, and the states in the gray region are prohibited because of the computational and radio resource constraints. ($K = 3$, $N = 4$, and $f_1(n) = n\mu_0$).

A similar problem has been formulated in the context of *Stochastic Knapsack Problem* [22], which is a stochastic extension of the traditional knapsack problem. Specifically, the items which occupy a certain amount of space come into and leave a knapsack randomly. The model we formulate is mathematically equivalent to stochastic knapsacks under *coordinate convex* [23] admission control policies. However, the focus of these previous work was to find the policy that maximizes the reward of storing items, and the analysis was limited to problems with small dimensionality because the complexity increases dramatically as the number of item classes grows. In contrast, we aim at evaluating the blocking probability instead of reward, and we have to address the large-dimensionality problems due to the large sizes of VBS pools.

C. Stationary Distribution

To perform further analysis, we need to derive the stationary distribution of $\boldsymbol{U}(t)$. Fortunately, the model we formulated

guarantees the reversibility of $\boldsymbol{U}(t)$ as in the following theorem, which in turn results in a product-form expression for the stationary distribution.

Theorem 1 (Reversibility): A continuous-time Markov chain with a state set as in (1) and transition rates as in (2) is reversible.

The reversibility of $\boldsymbol{U}(t)$ has been proved in [24] for more general cases. We provide an alternative proof in the Appendix using Kolmogorov's Criterion of Reversibility. Since $\boldsymbol{U}(t)$ is reversible, the local balance equation holds in the statistical equilibrium

$$\begin{aligned}&\Pr\left\{\boldsymbol{U}(\infty)=\boldsymbol{u}^{(i)}\right\} q_{\boldsymbol{u}^{(i)}\boldsymbol{u}^{(j)}}\\&\quad=\Pr\left\{\boldsymbol{U}(\infty)=\boldsymbol{u}^{(j)}\right\} q_{\boldsymbol{u}^{(j)}\boldsymbol{u}^{(i)}}.\end{aligned}\tag{3}$$

For simplicity, $\Pr\{\boldsymbol{U}(\infty)=\boldsymbol{u}\}$ is hereafter denoted by $\Pr\{\boldsymbol{u}\}$ or $\Pr\left\{u_{1,1},\cdots,u_{1,M_1},\cdots,u_{V,1},\cdots,u_{V,M_V}\right\}$. Without loss of generality, let $\boldsymbol{u}^{(i)}$ and $\boldsymbol{u}^{(j)}$ be two arbitrary neighboring states:

$$\begin{aligned}\boldsymbol{u}^{(i)}&=(u_1,\cdots,u_{v,m},\cdots,u_{V,M_V})^T\\\boldsymbol{u}^{(j)}&=(u_1,\cdots,u_{v,m}+1,\cdots,u_{V,M_V})^T,\end{aligned}$$

and substituting (2) into (3) yields:

$$\begin{aligned}&\Pr\left\{u_1,\cdots,u_{v,m},\cdots,u_{V,M_V}\right\}\lambda_v\\&\quad=\Pr\left\{u_1,\cdots,u_{v,m}+1,\cdots,u_{V,M_V}\right\} f_v(u_{v,m}+1).\end{aligned}\tag{4}$$

Clearly, this implies a recursive equation for computing the stationary distribution. Continuing the recursion down to 0 for the $(\sum_{w=1}^{v-1} M_w+m)$-th entry:

$$\begin{aligned}&\Pr\left\{u_1,\cdots,u_{v,m}+1,\cdots,u_{V,M_V}\right\}\\&\quad=\Pr\left\{u_1,\cdots,0,\cdots,u_{V,M_V}\right\}\cdot\frac{\lambda_v^{u_m+1}}{\prod_{i=1}^{u_{v,m}+1} f_v(i)}.\end{aligned}\tag{5}$$

Repeating the same process for other entries yields:

$$\Pr\{\boldsymbol{u}\}=P_0\prod_{v=1}^{V}\prod_{m=1}^{M_v}\frac{\lambda_v^{u_{v,m}}}{\prod_{i=1}^{u_{v,m}} f_v(i)},\tag{6}$$

in which

$$\begin{aligned}P_0&=\Pr\{0,\cdots,0,\cdots,0\}\\&=\left(\sum_{\boldsymbol{u}\in\mathbb{U}}\prod_{v=1}^{V}\prod_{m=1}^{M_v}\frac{\lambda_v^{u_{v,m}}}{\prod_{i=1}^{u_{v,m}} f_v(i)}\right)^{-1}\end{aligned}\tag{7}$$

is the probability of zero state and can be derived directly from the unity of probability distribution. As can be seen in (6) and (7), the stationary distribution of any state $\boldsymbol{u}$ is proportional to the product of terms which can be solely determined by the entry values of $\boldsymbol{u}$.

Remark 1: Although we formulate our problem with the assumption of exponentially distributed service demand, it is worthy of noting that the above product-form stationary distribution also applies to other non-exponential service demand distributions. It is proved in [24] that the product form distribution is valid for any service time distributions with rational Laplace transforms.

III. Blocking Probability

A. Brute-Force Evaluation

The admission control policy we have enforced on the VBS pool will cause session blockings. These blocking events can be classified into two classes: radio blockings (denoted by B_r) and computational blockings (denoted by B_c). Radio blocking is defined as the blocking events *solely* due to insufficient r-servers, i.e. $U_{v,m}(t^-)=K$, and $\sum_{v=1}^{V}\sum_{m=1}^{M_v}U_{v,m}(t^-)<N$, while computational blocking is defined as the blocking events due to insufficient c-servers *regardless of r-servers*, i.e. $\sum_{v=1}^{V}\sum_{m=1}^{M_v}U_{v,m}(t^-)=N$. Here t^- means the epoch just prior to a session arrival. Because we define radio blocking as the events that are solely due to insufficient r-servers, the blocking events that are due to simultaneously insufficient r-servers and c-servers are explicitly classified as computational blocking. It is worth noting that these doubly blocking events can be instead classified as radio blocking without affecting the overall blocking probability. But doing so will nevertheless result in less concise mathematical definition for both classes of events. Therefore, radio and computational blockings events are mutually exclusive, i.e. $B_r\cap B_c=\varnothing$. The union set of radio and computational blocking is further defined as the overall blocking $B=B_r\cup B_c$.

Next we derive the expression for the probability of radio and computational blockings. Since Poisson arrivals see time-averages (PASTA) [25], the blocking probability can be evaluated from the stationary distribution we have just derived. Concretely, the radio blocking probability for sessions in class-v VBSs is:

$$P_v^{\mathrm{br}}=\sum_{m=1}^{M_v}\frac{1}{M_v}\sum_{\boldsymbol{u}\in\mathbb{U}_{\mathrm{br}}^{v,m}}\Pr\{\boldsymbol{u}\}\tag{8}$$

$$=P_0\sum_{\boldsymbol{u}\in\mathbb{U}_{\mathrm{br}}^{v,1}}\prod_{w=1}^{V}\prod_{m=1}^{M_w}\frac{\lambda_w^{u_{w,m}}}{\prod_{i=1}^{u_{w,m}} f_w(i)}\tag{9}$$

$$=P_0\frac{\lambda_v^{K_v}}{\prod_{i=1}^{K_v} f_v(i)}\tag{10}$$

$$\cdot\sum_{\boldsymbol{u}\in\mathbb{U}_{\mathrm{br}}^{v,1}}\left[\left(\prod_{w\neq v}\prod_{m=1}^{M_w}\frac{\lambda_w^{u_{w,m}}}{\prod_{i=1}^{u_{w,m}} f_w(i)}\right)\cdot\left(\prod_{m=2}^{M_v}\frac{\lambda_v^{u_{v,m}}}{\prod_{i=1}^{u_{v,m}} f_v(i)}\right)\right],\tag{11}$$

where $\mathbb{U}_{\mathrm{br}}^{v,m}=\{\boldsymbol{u}\mid u_{v,m}=K_v,u_{1,1}+\cdots+u_{1,M_1}+\cdots+u_{V,1}+\cdots+u_{V,M_V}<N\}$ and (9) holds because (6) is symmetric for entries with same values for index v, i.e.

$$\begin{aligned}&\Pr\left\{\cdots,u_{v,i},\cdots,u_{v,j},\cdots\right\}\\&\quad=\Pr\left\{\cdots,u_{v,j},\cdots,u_{v,i},\cdots\right\}.\end{aligned}\tag{12}$$

Similarly, the computational blocking probability is:

$$\begin{aligned}P^{\mathrm{bc}}&=\sum_{\boldsymbol{u}\in\mathbb{U}_{\mathrm{bc}}^{N}}\Pr\{\boldsymbol{u}\}\\&=P_0\sum_{\boldsymbol{u}\in\mathbb{U}_{\mathrm{bc}}^{N}}\prod_{v=1}^{V}\prod_{m=1}^{M_v}\frac{\lambda_v^{u_{v,m}}}{\prod_{i=1}^{u_{v,m}} f_v(i)},\end{aligned}\tag{13}$$

where $\mathbb{U}_{\text{bc}}^N = \{\boldsymbol{u} \mid u_{1,1}+\cdots+u_{1,M_1}+\cdots+u_{V,1}+\cdots+u_{V,M_V} = N, u_{v,m} \le K_v\}$. The overall blocking probability for class-v VBSs can then be brute-force evaluated by summing up radio and computational blocking probability

$$P_v^{\text{b}} = \Pr\{B\} = P_v^{\text{br}} + P^{\text{bc}}. \tag{14}$$

B. Recursive Evaluation

Theoretically, the blocking probability under arbitrary system parameter can be calculated with brute-force evaluation. However the calculation process is exponentially hard and can become intractable when the pool size is extremely large. To reduce the computational complexity, we next give a recursive method for calculating the blocking probability. We will first introduce two auxiliary functions and re-express the blocking probability with respect to these functions. Then we will establish a recursive relationship to evaluate those two auxiliary functions and provide an analysis on the computational complexity of the proposed recursive evaluation method. These two auxiliary functions are defined as follows:

$$C(N, \boldsymbol{M}) = \sum_{\boldsymbol{u}\in\mathbb{U}_{\text{bc}}^N} \prod_{w=1}^{V}\prod_{m=1}^{M_w} \frac{\lambda_w^{u_{w,m}}}{\prod_{i=1}^{u_{w,m}} f_w(i)}, \tag{15}$$

$$R(N, \boldsymbol{M}) = \sum_{\boldsymbol{u}\in(\mathbb{U}_{\text{bc}}^N)^C} \prod_{w=1}^{V}\prod_{m=1}^{M_w} \frac{\lambda_w^{u_{w,m}}}{\prod_{i=1}^{u_{w,m}} f_w(i)}, \tag{16}$$

where $\boldsymbol{M} = (M_1, \cdots, M_v, \cdots, M_V)^T$ and $\left(\mathbb{U}_{\text{bc}}^N\right)^C = \{\boldsymbol{u} \mid u_{1,1}+\cdots+u_{1,M_1}+\cdots+u_{V,1}+\cdots+u_{V,M_V} < N, u_{v,m} \le K_v\}$ is the complement set of set $\mathbb{U}_{\text{bc}}^N$ in set $\mathbb{U}$. Clearly, $C(N, \boldsymbol{M})$ and $R(N, \boldsymbol{M})$ are proportional to the sum of probability terms over $\mathbb{U}_{\text{bc}}^N$ and $\left(\mathbb{U}_{\text{bc}}^N\right)^C$, respectively. Therefore, the blocking probability (11) and (13) can be re-expressed as,

$$\begin{aligned} P_v^{\text{br}} &= P_0 \cdot \frac{\lambda_v^{K_v}}{\prod_{i=1}^{K_v} f_v(i)} R(N - K_v, \boldsymbol{M} - \hat{\boldsymbol{e}}_v), \\ P^{\text{bc}} &= P_0 \cdot C(N, \boldsymbol{M}), \\ P_0 &= R^{-1}(N+1, \boldsymbol{M}), \end{aligned} \tag{17}$$

where $\hat{\boldsymbol{e}}_v = (0, \cdots, 0, \overbrace{1}^{v\text{-th}}, 0, \cdots, 0)^T$ is a column vector of length V. From the definition of $C(N, \boldsymbol{M})$ and $R(N, \boldsymbol{M})$, the following recursive relationships exist:

$$C(N, \boldsymbol{M}) = \begin{cases} \dfrac{\lambda_v^{N_2(v)}}{\prod_{i=1}^{N_2(v)} f_v(i)}, & \boldsymbol{M} = \hat{\boldsymbol{e}}_v \\ \displaystyle\sum_{n=N_1(v)}^{N_2(v)} \frac{\lambda_v^n}{\prod_{i=1}^{n} f_v(i)} C(N-n, \boldsymbol{M} - \hat{\boldsymbol{e}}_v), & \boldsymbol{M} > \hat{\boldsymbol{e}}_v, \end{cases} \tag{18}$$

$$R(N, \boldsymbol{M}) = \begin{cases} 0, & N = 1 \\ R(N+1, \boldsymbol{M}) - C(N, \boldsymbol{M}), & 1 < N < \boldsymbol{M}^T\boldsymbol{K} + 1 \\ \prod_{w=1}^{V}\left(\sum_{n=1}^{K_w} \dfrac{\lambda_w^n}{\prod_{i=1}^{n} f_w(i)}\right)^{M_w}, & N = \boldsymbol{M}^T\boldsymbol{K} + 1, \end{cases} \tag{19}$$

where

$$\begin{aligned} N_1(v) &= \max\left[0, N - \sum_{w\neq v} M_w K_w - (M_v - 1)K_v\right], \\ N_2(v) &= \min(K_v, N), \end{aligned}$$

and $\boldsymbol{M}^T\boldsymbol{K} = \sum_{w=1}^{V} M_w K_w$. Following these recursive relationship, we can calculate the value $C(N, \boldsymbol{M})$ and $R(N, \boldsymbol{M})$ for arbitrary input through iterative calculation. Concretely, we can iterate for $C(N, \boldsymbol{M})$ from any $\hat{\boldsymbol{e}_v}$ following (18). After that, we can reuse the calculated $C(N, \boldsymbol{M})$ values to calculate for $R(N, \boldsymbol{M})$ by iterating from either $N = 0$ or $N = \boldsymbol{M}^T\boldsymbol{K} + 1$ according to (19). Note that the comparison between $\boldsymbol{M}$ and $\hat{\boldsymbol{e}}$ in (18) is element-wise, and in this sense $\boldsymbol{M}$ is always greater or equal to $\hat{\boldsymbol{e}}$ in non-empty pools. With the above recursive relationship, the computational complexity of blocking probability can be reduced to at most the second power of the pool size, as stated in the following theorem.

Theorem 2: The upper bound for the overall computational complexity of the proposed recursive method is:

$$C \le \left[(\max_v K_v)^2 + \max_v K_v\right] \cdot |\boldsymbol{M}|^2. \tag{20}$$

Proof: See Appendix B for proof. ■

C. Large Pool Approximation

The above quadratic computational complexity can also become intractable for very large pools. To overcome this inconvenience, next we present a closed-form approximation for the blocking probability with large VBS pools. Although the above recursive expression cannot lead us to a direct approximation, the product-form stationary distribution of $\boldsymbol{U}$ does facilitate an indirect one.

First define some auxiliary variables. Let $\tilde{U}_{w,m}$ be the number of sessions in the m-th class-w VBS when

$$\begin{aligned} N &\ge \boldsymbol{M}^T\boldsymbol{K}, \\ \mu_w &= \text{E}\left[\tilde{U}_{w,m}\right], \\ \sigma_w^2 &= \text{Var}\left[\tilde{U}_{w,m}\right].^1 \end{aligned}$$

Also, let $\tilde{S}_{\boldsymbol{M}} = \frac{1}{|\boldsymbol{M}|}\sum_{w=1}^{V}\sum_{m=1}^{M_w}\tilde{U}_{w,m}$, and $\tilde{S}_{M_w} = \frac{1}{M_w}\sum_{m=1}^{M_w}\tilde{U}_{w,m}$. Using these notations, the large-pool approximation is stated in the following Theorem.

Theorem 3 (Large Pool Blocking Probability): For $N > |\boldsymbol{M}|\mu$, the session blocking probability for class-v VBSs is:

$$\lim_{|\boldsymbol{M}|\to\infty} P_v^{\text{b}} = \frac{1}{\sqrt{2\pi|\boldsymbol{M}|\sigma^2}} \frac{1}{e^{\alpha^2/2} - 1} + \tilde{P}_v^{\text{br}}, \tag{21}$$

where $\mu = \sum_{w=1}^{V}\beta_w\mu_w$, $\sigma^2 = \sum_{w=1}^{V}\sigma_w^2$, and $\beta_w = \lim_{|\boldsymbol{M}|\to\infty}\frac{M_w}{|\boldsymbol{M}|}$; $\alpha = \frac{N-|\boldsymbol{M}|\mu}{\sqrt{|\boldsymbol{M}|}\sigma}$ is the normalized number of c-servers; $\tilde{P}_v^{\text{br}}$ is the overall blocking probability in class-v VBSs when $N > \boldsymbol{M}^T\boldsymbol{K}$.

Proof: See Appendix C for proof. ■

[1]It is obvious that $\tilde{U}_{w,m}$ are i.i.d random variables for $m = 1, 2, \cdots, M_w$.

Remark 2: With the approximation in (21), the blocking probability can be calculated in one shot as long as the first-order and second-order statistics of $\tilde{U}_{v,m}$ are known. We will see in Section V that these statistics are rather easy to obtain in some very practical scenarios.

Remark 3: Under the large-pool assumption, the blocking probability in (21) is decomposed into two terms. The first term $\frac{1}{\sqrt{2\pi|\boldsymbol{M}|\sigma^2}}\frac{1}{e^{\alpha^2/2}-1}$ reflects the portion of blockings that are solely due to insufficient computational resource, while the second term $\tilde{P}_v^{br}$ reflects the portion that are solely due to insufficient radio resource. This result reveals the decoupling feature between radio and computational blockings in large VBS pools.

Remark 4: Although we assume $K_v < \infty$ in our derivation, (21) is still true when $K \to \infty$. In this case, the approximation used in (59) may not hold anymore. But this will not cause any problem since the radio blocking probability $\tilde{P}_v^{\mathrm{b}}$ will be 0 when we have infinite radio resource. This will force the second term of (21) to become zero, canceling out the inconsistency in the above approximation.

IV. Statistical Multiplexing Gain

Since stochastic user sessions from different VBSs are consolidated, it is natural to expect a reduction in the required amount of computational resource compared with non-pooling schemes due to statistical multiplexing. Next we provide a theoretical analysis for the statistical multiplexing gain. We first derive the asymptotic utilization ratio of computational resource in large VBS pools.

Theorem 4 (Large Pool Utilization Ratio): When c-servers are sufficiently provisioned (i.e. the number of c-servers is greater or equal to the number of r-servers, or $N \geq \boldsymbol{M}^T\boldsymbol{K}$), the utilization ratio of computational resource converges almost surely to a constant number that is smaller than 1 as $|\boldsymbol{M}| \to \infty$:

$$\lim_{|\boldsymbol{M}|\to\infty} \eta \triangleq \frac{\sum_{w=1}^{V}\sum_{m=1}^{M_w}\tilde{U}_{w,m}}{N} \xrightarrow{\text{a.s.}} \frac{|\boldsymbol{M}|\mu}{N} < 1. \quad (22)$$

Proof: See Appendix D for proof. ∎

This theorem implies that there exist some $(1-\eta)$ redundant computational resource when the VBS pool is large enough. Thus this limit can be seen as an the maximum portion of c-servers that one can turn down to save computational resource. The potential to further turn down c-servers can in turn be defined as the difference between current utilization ratio of c-servers and the large-pool limit η:

Definition 1 (Residual Pooling Gain): The residual pooling gain of a VBS pool is:

$$g_r \triangleq \frac{N}{\boldsymbol{M}^T\boldsymbol{K}} - \eta. \quad (23)$$

Although some c-servers can be turned down due to the statistical multiplexing effect, the negative effect is that the overall blocking probability P^{b} will increase following (21). Hence we have to trade QoS for the statistical multiplexing gain. This tradeoff will be favorable as long as the degradation in the QoS is not very significant. Using the results in Theorem 3, we can directly derive the following corollary to quantify such "significance" and approximate the gain of VBS pools of different sizes.

Corollary 1 (Critical Tradeoff Point): When $|\boldsymbol{M}| \to \infty$, the minimum computational resource α^* required to keep the overall blocking probability $P_v^{\mathrm{b}} \leq \tilde{P}_v^{\mathrm{br}} + \delta$ ($\delta \approx 0$) for all v is

$$\alpha^* = \sqrt{2\ln\left(\frac{1}{\sqrt{2\pi|\boldsymbol{M}|\sigma^2\delta^2}} + 1\right)}. \quad (24)$$

We will show later in Fig. 3, that this critical tradeoff point is essentially the point where the blocking probability start to increase at a significantly higher speed. This type of points are often referred to "knee points".

The residual pooling gain at this critical tradeoff point is bounded as follows:

$$g_r^* = \frac{N - |\boldsymbol{M}|\mu}{\boldsymbol{M}^T\boldsymbol{K}} = \sigma\frac{\alpha^*\sqrt{|\boldsymbol{M}|}}{\boldsymbol{M}^T\boldsymbol{K}} \in \frac{\alpha^*}{\sqrt{|\boldsymbol{M}|}}\cdot\left[\frac{\sigma}{\max_v K_v}, \frac{\sigma}{\min_v K_v}\right], \quad (25)$$

by which g_r^* is roughly proportional to $\alpha^*/\sqrt{|\boldsymbol{M}|}$. Note α^* is also a function of the pool size $|\boldsymbol{M}|$, so g_r^* is not necessarily proportional to $1/\sqrt{|\boldsymbol{M}|}$. Investigating two extreme cases will help to reveal the true relationship between g_r^* and the pool size $|\boldsymbol{M}|$.

Extreme Case 1: If $|\boldsymbol{M}|$ is not very large such that $\sqrt{2\pi|\boldsymbol{M}|\sigma^2\delta^2} \ll 1$ and $\sqrt{|\boldsymbol{M}|} \ll 1/\delta^2$, then α^* is approximately constant because:

$$\begin{aligned}\alpha^* &\approx \sqrt{2\ln\left(\frac{1}{\sqrt{2\pi|\boldsymbol{M}|\sigma^2\delta^2}}\right)} \\ &= \sqrt{\ln\left(\frac{1}{2\pi\sigma^2}\right) + \ln\left(\frac{1}{\delta^2}\right) + \ln\left(\frac{1}{|\boldsymbol{M}|}\right)} \\ &\approx \sqrt{\ln\left(\frac{1}{2\pi\sigma^2\delta^2}\right)}. \end{aligned} \quad (26)$$

In this case $g_r^* \propto |\boldsymbol{M}|^{-1/2}$, which decreases slowly with $|\boldsymbol{M}|$. Even so, considering the fact that the residual pooling gain is at most 1, we can still get considerable pooling gain with a small value of $|\boldsymbol{M}|$.

Extreme Case 2:, if $|\boldsymbol{M}|$ is very large such that $\sqrt{2\pi|\boldsymbol{M}|\sigma^2\delta^2} \gg 1$, notice $\lim_{x\to 0}\ln(1+x) \approx x$:

$$\alpha^* \approx \sqrt{2\frac{1}{\sqrt{2\pi|\boldsymbol{M}|\sigma^2\delta^2}}} \propto |\boldsymbol{M}|^{-1/4}. \quad (27)$$

In this case $g_r^* \propto |\boldsymbol{M}|^{-3/4}$, which means that the decrease in the residual pooling gain will speed-up as $|\boldsymbol{M}|$ grows large.

Remark 5: As can be seen in (24), the critical point is invariant of the VBS class index v. This implies that the VBS heterogeneity is decomposed in large VBS pools. The reason for this phenomenon may be that in large VBS pools, the absolute number of c-servers is large. Therefore, different class of VBSs may tend to interfere less with each other.

V. Example Scenarios

In this section, we will apply the results derived so far to two specific scenarios: real-time and delay-tolerant traffic. For each scenario, we will first explain how they can be mapped to our model and then we will perform necessary derivations. Note in real-life systems, both type of traffic may exist at the same time. In such a case, the following analysis can still be applied if the available resources are divided to serve these two type of traffic separately.

A. Real-Time Traffic

For real-time traffic such as voice calls, active sessions will constantly bring in signal processing workload. Therefore, dedicated r-servers and c-servers need to be provisioned upon admission to guarantee the QoS of active sessions. The service capacity in this scenario equals the temporal duration of sessions, which is not affected by the scheduling policy of VBS pools once the sessions are accepted. As a result, the departure rate function can be simplified as $f_\upsilon(i) = i\mu_0$. The QoS target in this case is to keep the overall session blocking probability for class-υ VBSs under a certain small threshold $P_\upsilon^{\text{bth}} \approx 0$. Obviously, the session dynamics in different VBSs are mutually independent when computational resource are sufficiently provisioned ($N > \boldsymbol{M}^T\boldsymbol{K}$). Therefore the radio blocking probability $\tilde{P}_\upsilon^{\text{br}}$ can be calculated as

$$\tilde{P}_\upsilon^{\text{br}} = \frac{a_\upsilon^{K_\upsilon}}{K_\upsilon!}\left(\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}\right)^{-1} \le P_\upsilon^{\text{bth}} \approx 0, \tag{28}$$

where $a_\upsilon = \lambda_\upsilon/\mu_\upsilon$. Then the first-order and second-order statistics of $\tilde{U}_{\upsilon,m}$ can be approximated as follows:

$$\begin{aligned}\mathrm{E}\left[\tilde{U}_{\upsilon,m}\right] &= \frac{\sum_{i=0}^{K_\upsilon} i\frac{a_\upsilon^i}{i!}}{\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}} = \frac{a\sum_{i=0}^{K_\upsilon-1}\frac{a_\upsilon^i}{i!}}{\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}} \\ &= a_\upsilon\left(1 - \frac{a_\upsilon^{K_\upsilon}}{K_\upsilon!}\left(\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}\right)^{-1}\right) \approx a_\upsilon, \end{aligned}\tag{29}$$

$$\begin{aligned}\mathrm{E}\left[\tilde{U}_{\upsilon,m}^2\right] &= \frac{\sum_{i=0}^{K_\upsilon} i^2\frac{a_\upsilon^i}{i!}}{\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}} = \frac{a_\upsilon\sum_{i=0}^{K_\upsilon-1}(i+1)\frac{a_\upsilon^i}{i!}}{\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}} \\ &= \frac{a_\upsilon(\sum_{i=0}^{K_\upsilon-1}\frac{a_\upsilon^i}{i!} + a_\upsilon\sum_{i=0}^{K_\upsilon-2}\frac{a_\upsilon^i}{i!})}{\sum_{i=0}^{K_\upsilon}\frac{a_\upsilon^i}{i!}} \approx a_\upsilon + a_\upsilon^2. \end{aligned}\tag{30}$$

Then

$$\mu_\upsilon \approx a_\upsilon, \tag{31}$$

$$\sigma_\upsilon^2 \approx a_\upsilon. \tag{32}$$

B. Delay-Tolerant Traffic

For delay-tolerant traffic such as packet data, the pool scheduler can opportunistically divide the total service capacity among active sessions. Here we assume constant service capacity rate $f_\upsilon(i) = \mu_\upsilon$ for class-υ VBSs and *Proportional Fairness* scheduling algorithm which effectively divides the total service capacity equally among active sessions. Although this assumption manifests a processor sharing model, it is essentially equivalent to a Markovian queueing model with the same λ_υ and μ_υ. Note because sessions would require certain amount of radio resources for signaling, new sessions would be rejected if there're no more signaling radio channels regardless of the data channels left. Therefore even the sessions can wait they still cannot be admitted into the system. If the rejected session decides to wait and retry, it will be considered as a new session. WhatâĽ™s more, many delay-tolerant traffic or elastic traffic still have a minimum rate requirement, which also limits the number of sessions that can be simultaneously served by the system.

To derive the statistics in this scenario, first let $a_\upsilon = \lambda_\upsilon/\mu_\upsilon$ be the traffic load of the VBSs and define the following auxiliary function $A(a, K)$:

$$\begin{aligned} A(a,K) &= \sum_{i=0}^{K} a^i = \frac{1-a^{K+1}}{1-a}, \\ A'_a(a,K) &= \left(\sum_{i=0}^{K} a^i\right)'_a = \sum_{i=1}^{K} i a^{i-1} \\ &= \frac{1-(K+1)a^K + Ka^{K+1}}{(1-a)^2}, \\ A''_a(a,K) &= \left(\sum_{i=0}^{K} a^i\right)''_a = \sum_{i=2}^{K} i(i-1)a^{i-2}. \end{aligned}\tag{33}$$

With these definitions, the average and covariance of $\tilde{U}_{\upsilon,m}$ can be expressed as:

$$\mathrm{E}\left[\tilde{U}_{\upsilon,m}\right] = \frac{\sum_{i=1}^{K_\upsilon} i a_\upsilon^i}{\sum_{i=0}^{K_\upsilon} a_\upsilon^i} = \frac{a_\upsilon A'_a(a_\upsilon, K_\upsilon)}{A(a_\upsilon, K_\upsilon)}, \tag{34}$$

$$\begin{aligned}\mathrm{E}\left[\tilde{U}_{\upsilon,m}^2\right] &= \frac{\sum_{i=1}^{K_\upsilon} i^2 a_\upsilon^i}{\sum_{i=0}^{K_\upsilon} a_\upsilon^i} = \frac{\sum_{i=1}^{K_\upsilon} i a_\upsilon^i + \sum_{i=2}^{K_\upsilon} i(i-1)a_\upsilon^i}{\sum_{i=0}^{K_\upsilon} a_\upsilon^i} \\ &= \frac{a_\upsilon A'_a(a_\upsilon, K_\upsilon) + a_\upsilon^2 A''_a(a_\upsilon, K_\upsilon)}{A(a_\upsilon, K_\upsilon)}. \end{aligned}\tag{35}$$

Again when computational resource are sufficiently provisioned ($N > \boldsymbol{M}^T\boldsymbol{K}$), we have

$$\tilde{P}_\upsilon^{\text{br}} = \frac{a_\upsilon^k}{\sum_{i=0}^{K_\upsilon} a_\upsilon^i} = \frac{a_\upsilon^k}{A(a_\upsilon, K_\upsilon)}. \tag{36}$$

Although these formulas are already enough for us to evaluate the performance of a VBS pool, the evaluation is nevertheless quite cumbersome. To simplify these formulas, we further assume that K_υ for all υ are large enough such that

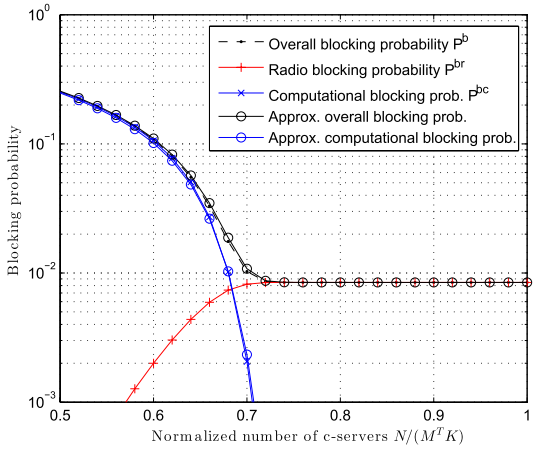

Fig. 3. Blocking probability of a homogeneous VBS pool as a function of normalized number of c-servers ($N/\boldsymbol{M}^T\boldsymbol{K}$) under real-time traffic. Black box indicates the knee point. Simulation parameters: $M_1 = 40$, $a_1 = 20$, $P_1^{\text{bth}} = 10^{-2}$, $K_1 = 30$.

$K_v^2 a_v^{K_v} \to 0$.[2] In this regime,

$$A(a,K) \approx \frac{1}{1-a},$$
$$A'_a(a,K) \approx \frac{1}{(1-a)^2},$$
$$A''_a(a,K) \approx \frac{2}{(1-a)^3}. \tag{37}$$

Using (37), (34) and (35) can be simplified as

$$\mathrm{E}\left[\tilde{U}_{v,m}\right] \approx \frac{a_v}{1-a_v}, \tag{38}$$

$$\mathrm{E}\left[\tilde{U}^2_{v,m}\right] \approx \frac{a_v}{1-a_v} + \frac{2a_v^2}{(1-a_v)^2}. \tag{39}$$

Thus

$$\mu_v \approx \frac{a_v}{1-a_v}, \tag{40}$$

$$\sigma_v^2 \approx \frac{a_v}{1-a_v} + \frac{a_v^2}{(1-a_v)^2}. \tag{41}$$

VI. Numerical Results

In this section, we will use the recursive method to numerically evaluate the blocking probability and compare them with the large-pool approximations.

A. Basic Characteristics

Fig. 3 shows the exact and large-pool-approximated blocking probability of a VBS pool under real-time traffic and different number of c-servers, and Fig. 4 shows the same metrics for a VBS pool under delay-tolerant traffic. Note x-axis is normalized by the number of c-servers required without pooling to show the relative pooling gain. As can be seen, the trend in both figures are similar. This coincide with our large-pool approximation results that the blocking probability are affected only by the first- and second-order statistics of the number of sessions in the VBS pool. For this reason, we will only present

[2]This assumption is realistic because $a_v^{K_v}$ will diminish exponentially when $a_v < 1$. Thus $K_v^2 a_v^{K_v}$ will be driven to 0 for large enough K_v.

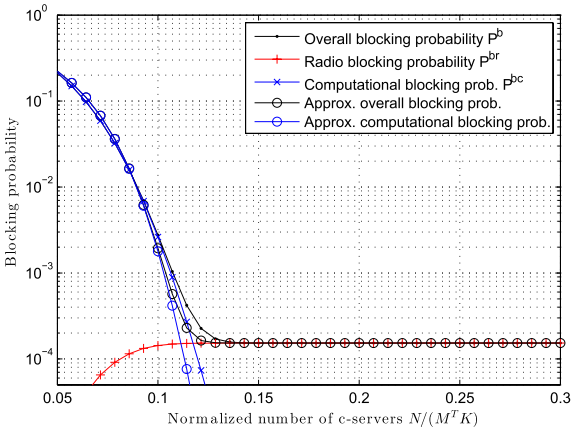

Fig. 4. Blocking probability of a homogeneous VBS pool as a function of normalized number of c-servers ($N/\boldsymbol{M}^T\boldsymbol{K}$) under delay-tolerant traffic. Simulation parameters: $M_1 = 100$, $a_1 = 0.5$, $P_1^{\text{bth}} = 5 \times 10^{-4}$, $K_1 = 10$.

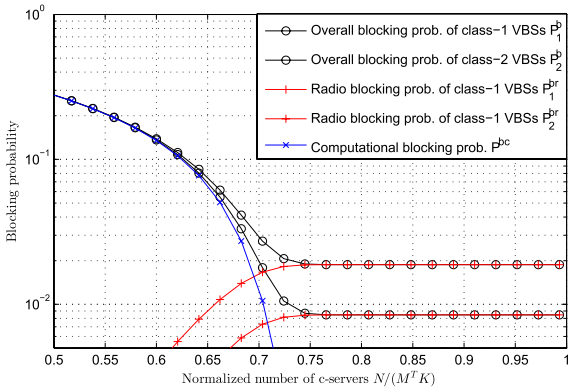

Fig. 5. Blocking probability of a heterogeneous VBS pool as a function of normalized number of c-servers ($N/\boldsymbol{M}^T\boldsymbol{K}$) under real-time traffic. Simulation parameters: $\boldsymbol{M} = [20,20]^T$, $\boldsymbol{a} = [20,20]^T$, $\boldsymbol{P}^{\text{bth}} = [1,2]^T \times 10^{-2}$, $\boldsymbol{K} = [30,28]^T$.

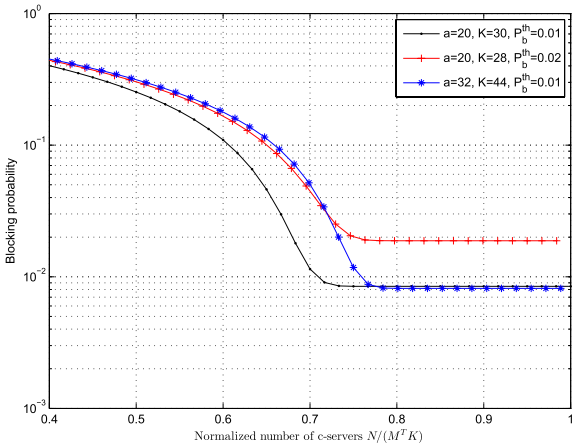

Fig. 6. Blocking probability of a homogeneous VBS pool as a function of normalized number of c-servers ($N/\boldsymbol{M}^T\boldsymbol{K}$) under real-time traffic with different traffic load and QoS guarantees. Pool size $M = 40$.

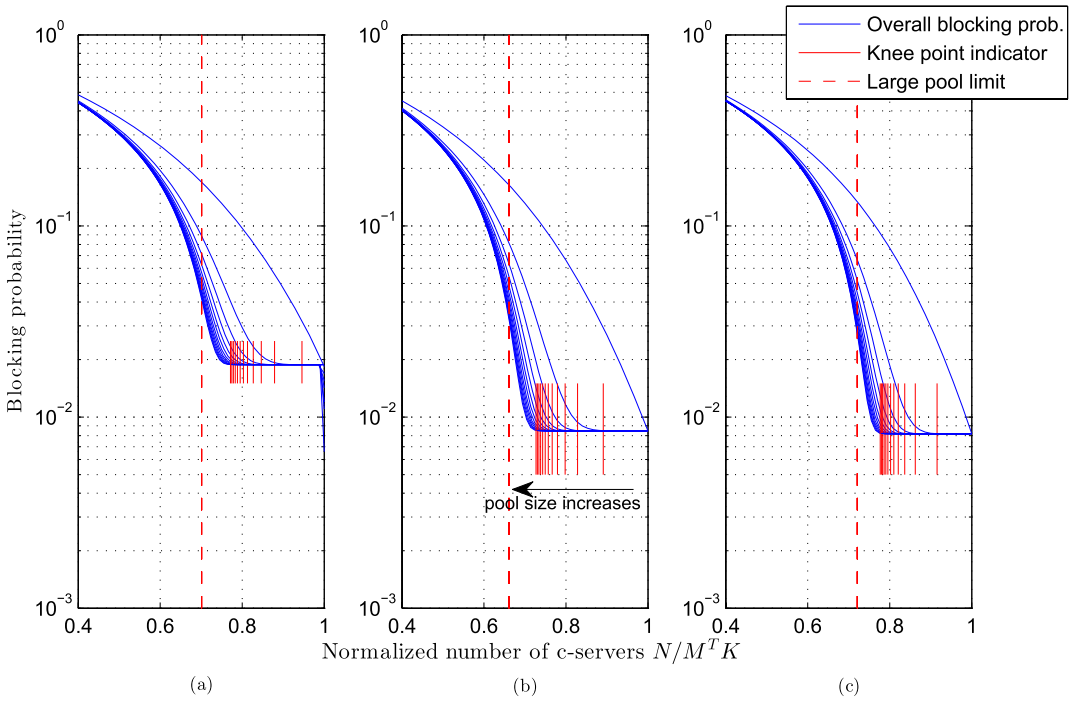

Fig. 7. Blocking probability in homogeneous VBS pools as a function of normalized number of c-servers under real-time traffic and different pool size. Red vertical line indicates the knee point position. Red dashed line indicates the large-pool limit. Curves to the left correspond to larger pool size. (a) $a = 20$, $K = 28$, $P_{\text{bth}} = 0.02$. (b) $a = 20$, $K = 30$, $P_{\text{bth}} = 0.01$. (c) $a = 32$, $K = 44$, $P_{\text{bth}} = 0.01$.

results for real-time traffic from now on, and the conclusions we draw should apply to the delay-tolerant case as well.

We can observe some basic blocking characteristic from these two figures: 1) when the number of c-servers is sufficient, the computational blocking probability P^{bc} is very small and below the scope of this figure; the overall blocking probability is dominated by radio blocking probability P^{br}, which is around the desired threshold P^{bth}. 2) As the number of c-servers decreases from its largest value $\boldsymbol{M}^T\boldsymbol{K}$, the computational blocking probability increases rapidly while the radio blocking probability start to decrease slightly; the net result of these two trends is a plateau before the critical tradeoff point ("knee point") and a significant increase after it. 3) If the number of c-servers is to further decrease, the overall blocking probability will be dominated by the computational blocking probability and saturates at probability 1. The radio blocking probability will decrease rapidly and its influence on overall blocking probability will diminish. Fig. 3 and Fig. 4 also show the approximated blocking probability. The fact that the "knee point" configuration saved more than 20% computational resources with a penalty of only 10^{-4} increase in the blocking probability demonstrates the benefit of statistical multiplexing. the As expected, these approximations are coherent with the exact values.

B. Heterogeneous VBSs

In Fig. 5, we show the blocking probability of a VBS pool with two class of VBSs. The two classes have the same number of VBSs and the same traffic load, but the QoS, and thus the number of provisioned r-servers, is different. From the curves we can observe similar basic blocking characteristics as in the single class case. Also, we can see that the overall blocking probability for the two classes are different: they are respectively close to their threshold blocking probability when c-servers are sufficient since the overall blocking probability are dominated by the radio blocking, and converges to the same curve when c-servers become insufficient because the computational blocking probability begins to overwhelm.

C. Influence of Traffic Load and QoS Target

In Fig. 6, we illustrate the influence of different traffic load and QoS target. As can be seen, the desired level of QoS (P^{bth}) determines the minimum blocking probability(i.e. height of the "plateau" to the right of the figure); while the traffic load a determines the position of the "knee point" and how fast the blocking probability saturates to 1.

D. Statistical Multiplexing Gain

Most importantly, we can quantify the statistical multiplexing gain of the simulated VBS pool with the equations derived previously. In Fig. 7, we compare the overall blocking probability of three VBS pools under varying pool size. The "knee point" position and large-pool limit are also marked out with vertical lines. As the pool size increases, the blocking probability curve (and so does the "knee point") is pushed to the left. But the closer the "knee point" is to the large-pool limit, the slower the remaining distance decreases with the pool size $|\boldsymbol{M}|$. This indicates a decreasing marginal statistical multiplexing gain. Comparing the curves, we can find that the

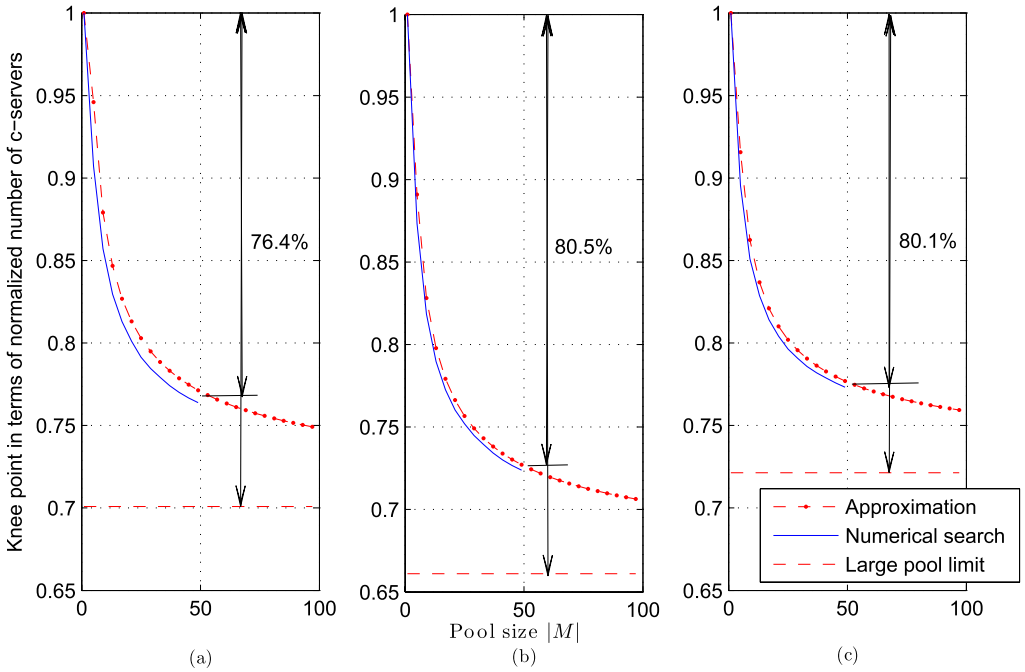

Fig. 8. Knee point position as a function of pool size. Knee point position is measured in terms of the number of c-servers required normalized by the number of c-servers needed without pooling. Red dotted line shows the knee point values derived from large-pool approximation, whereas blue line shows the values directly searched from numerical results. Red dashed lines represent the large-pool limit η as pool size approaches infinity. The percentage of the pooling gain at 50 VBSs with respect to the maximum possible gain $(1-\eta)$ is also noted in the figure. (a) $a = 20$, $K = 28$, $P_{\text{bth}} = 0.02$. (b) $a = 20$, $K = 30$, $P_{\text{bth}} = 0.01$. (c) $a = 32$, $K = 44$, $P_{\text{bth}} = 0.01$.

traffic load and the desired level of QoS have influence on the blocking probability and the statistical multiplexing gain.

To better investigate this influence, we show the knee point position versus varying pool size in Fig. 8. Firstly we can find that a medium sized VBS pool can readily obtain considerable statistical multiplexing gain and the marginal gain diminishes fast. Thus a huge number of VBSs is needed so that the pooling gain can approach the large-pool limit. These observations imply that a C-RAN formed with multiple medium sized VBS pools can obtain almost the same pooling gain as the one formed with a single huge pool. If we further take the expenditure of fronthaul network into consideration, the former choice may be far more economical than the latter one.

By contrasting the left and middle curves, we can see that stricter QoS requirements can increase the pooling gain. This is because on one hand, we need to increase the number of r-servers $\boldsymbol{K}$ in order to reduce the blocking probability, which will in turn increase $\boldsymbol{M}^T\boldsymbol{K}$; on the other hand, the average number of c-servers occupied is always around $|\boldsymbol{M}|\mu$. Therefore the stricter the QoS, the more idle c-servers there will be in the VBS pool and consequently the more the pooling gain. Also, we can see that the increase in traffic load will reduce the pooling gain by pushing the "knee point" to larger values. This observation indicates that, we may need to dynamically adjust the size[3] of VBS pools in order to get a satisfactory pooling gain under fluctuating traffic load.

[3]This can be achieved by dynamically changing the switching configuration of fronthaul so that the traffic of VBSs can be sent to a data center of the desired size.

VII. Conclusion

In this article, we proposed a multi-dimensional Markov model for VBS pools to analyze their statistical multiplexing gain. We showed that the proposed model have a product-form expression for the stationary distribution. We derived a recursive method for calculating the blocking probability of a VBS pool, and gave closed-form approximation when the pool is large enough. Based on these results, we derived the expressions for the statistical multiplexing gains and applied them to both real-time and delay-tolerant traffic. Numerical results reveal that 1) the pooling gain reaches a significant level even with medium pool size (more than 75% of the pooling gain can be achieved with around 50 VBSs); 2) the marginal gain of larger pool size tend to be negligible; 3) lighter traffic load and tighter QoS level can increase the pooling gain.

Our model can be extended in several aspects to accommodate for more general scenarios. Firstly, we assume that user sessions occupy equal and fixed amounts of radio and computational resources. Yet realistic resource scheduling algorithm may allocate different amount of resources for each individual session. To accommodate such cases, our model need to be further relaxed to allow state transitions among non-neighboring states. Secondly, we assume sessions are only attached to one cell of the system. Nevertheless, coordinated-multipoint (CoMP) transmission/reception may introduce sessions that simultaneously consume radio resources from multiple cells. This means the admission control of CoMP session is hinged upon the available radio resources in all its serving cells, which can be accounted for by introducing more comprehensive admission controls in the model.

Thirdly, we assume session arrival and service to be Poission. However, many emerging types of multi-media traffic is found to exhibit certain burstiness. The influence of burstiness can be investigated by assuming more general stochastic traffic and service models, e.g Interupted Poisson Process [26]. Fourthly, we investigated real-time and delay-tolerant traffic separately whereas they are likely to coexist in real system. To evaluate such heterogeneous traffic, our model need to be extended to account for the different resource usage patterns of the two session types. Last but not the least, our resource reservation model may not be the most efficient possible. For example, unused service capacity can be further shared to increase statistical multiplexing gain. Regarding this, our model need to be further refined to support more general admission control and service strategies.

APPENDIX A
REVERSIBILITY OF PROPOSED MODEL

To proof that $\boldsymbol{U}(t)$ is reversible, we first give the Kolmogorov's Criterion of Reversibility.

Theorem 5 (Kolmogorov's Criterion): A continuous-time Markov chain is reversible if and only if its transition rates satisfy

$$q_{\boldsymbol{u^{(1)}u^{(2)}}} q_{\boldsymbol{u^{(2)}u^{(3)}}} \cdots q_{\boldsymbol{u^{(n-1)}u^{(n)}}} q_{\boldsymbol{u^{(n)}u^{(1)}}} \tag{42}$$

$$= q_{\boldsymbol{u^{(1)}u^{(n)}}} q_{\boldsymbol{u^{(n)}u^{(n-1)}}} \cdots q_{\boldsymbol{u^{(3)}u^{(2)}}} q_{\boldsymbol{u^{(2)}u^{(1)}}} \tag{43}$$

for all finite sequences of states $\boldsymbol{u^{(1)}}, \boldsymbol{u^{(2)}}, \cdots, \boldsymbol{u^{(n)}} \in \mathbb{U}$.

We next verify that the Kolmogorov's Criterion is satisfied by $\boldsymbol{U}(t)$ for any finite sequences of states $\boldsymbol{u^{(1)}}, \boldsymbol{u^{(2)}}, \cdots, \boldsymbol{u^{(n)}} \in \mathbb{U}$.

Case 1: If (42) (or (43)) equals to 0, then there must be at least one product term being 0 in (42) (or (43)). Assuming this term to be $q_{\boldsymbol{u^{(i)}u^{(i+1)}}}$, then according to (2), the term $q_{\boldsymbol{u^{(i+1)}u^{(i)}}}$ in (43) (or (42)) must also be 0. Thus (43) (or (42)) must also equal to 0.

Case 2: If neither (42) nor (43) is 0, then none of the terms in (42) and (43) equals 0. According to (2), the transition resulting in the term $q_{\boldsymbol{u^{(i)}u^{(i+1)}}}$ must be between neighboring states, i.e. $\boldsymbol{u^{(i+1)}} - \boldsymbol{u^{(i)}} = (0, \cdots, 0, \pm 1, 0, \cdots, 0)^T$.

Notice that a transition such that the m-th entry of a state is changed from u_m to $u_m + 1$ will produce a product term λ in (42) and a product term $f(u_m + 1)$ in (43). And reversely, a transition such that the m-th entry is changed from $u_m + 1$ to u_m will produce a product term $f(u_m + 1)$ in (42) and a product term λ in (43).

If we denote the number of state transitions in the transition loop

$$(\boldsymbol{u^{(1)}}, \boldsymbol{u^{(2)}}, \cdots, \boldsymbol{u^{(n)}}, \boldsymbol{u^{(1)}}) \tag{44}$$

such that the $(\sum_{w=1}^{v-1} M_w + m)$-th state entry is changed from $u_{v,m}$ to $u_{v,m} + 1$ by $n^+_{v,m,u_{v,m}}$ times, and the number of transitions such that the $(\sum_{w=1}^{v-1} M_w + m)$-th entry is changed from $u_{v,m} + 1$ to $u_{v,m}$ by $n^-_{v,m,u_{v,m}}$ times, then

$$\text{Eq.(42)} = \prod_{w=1}^{V} \prod_{m=1}^{M_w} \prod_{u_{w,m}=1}^{K} \lambda^{n^+_{w,m,u_{w,m}}} f_w^{n^-_{w,m,u_{w,m}}}(u_{w,m} + 1), \tag{45}$$

$$\text{Eq.(43)} = \prod_{w=1}^{V} \prod_{m=1}^{M_w} \prod_{u_{w,m}=1}^{K} \lambda^{n^-_{w,m,u_{w,m}}} f_w^{n^+_{w,m,u_{w,m}}}(u_{w,m} + 1). \tag{46}$$

To make (44) a closed loop, we must have

$$n^+_{v,m,u_{v,m}} = n^-_{v,m,u_{v,m}} \tag{47}$$

for all $v = 1, 2, \cdots, V$, $m = 1, 2, \cdots, M_v$ and $u_{v,m} = 1, 2, \cdots, K$. Substituting (47) into (45) and (46) and we get that (42) equals to (43). Thus $\boldsymbol{U}(t)$ is reversible.

APPENDIX B
PROOF OF THEOREM 2

The calculation of a single $C(N, \boldsymbol{M})$ term involves at most K_v summations. Also, there are at most $N \leq \boldsymbol{M}^T \boldsymbol{K}$ such terms for some specific $\boldsymbol{M}$. Hence the computational complexity for all $C(N, \boldsymbol{M})$ terms for a VBS pool sized $\boldsymbol{M}$ is bounded as

$$C_1 \leq (\max_v K_v) \boldsymbol{M}^T \boldsymbol{K} |\boldsymbol{M}| \leq (\max_v K_v)^2 |\boldsymbol{M}|^2, \tag{48}$$

where $|\boldsymbol{M}| = \sum_{w=1}^{V} M_w$. At the same time, the calculation of $R(N, \boldsymbol{M})$ involves only a single subtraction (or summation). Again, there are at most $\boldsymbol{M}^T \boldsymbol{K}$ such terms for some specific $\boldsymbol{M}$. Therefore the computational complexity of $R(N, \boldsymbol{M})$ for a VBS pool with size vector $\boldsymbol{M}$ is bounded as

$$\begin{aligned} C_2 &\leq \boldsymbol{M}^T \boldsymbol{K} |\boldsymbol{M}| \\ &\leq (\max_v K_v) |\boldsymbol{M}|^2. \end{aligned} \tag{49}$$

All together, the overall computational complexity is bounded as

$$\begin{aligned} C &= C_1 + C_2 \\ &\leq \left[(\max_v K_v)^2 + \max_v K_v \right] |\boldsymbol{M}|^2. \end{aligned} \tag{50}$$

This bound is essentially quadratic in the pool size $|\boldsymbol{M}|$. Therefore, the computational complexity of blocking probability should also be quadratic in the pool size.

APPENDIX C
PROOF OF THEOREM 3

From the definition of $\tilde{S}_{\boldsymbol{M}}$ we know

$$\begin{aligned} \lim_{|\boldsymbol{M}| \to \infty} \tilde{S}_{\boldsymbol{M}} &= \lim_{|\boldsymbol{M}| \to \infty} \frac{1}{|\boldsymbol{M}|} \sum_{w=1}^{V} \sum_{m=1}^{M_w} \tilde{U}_{w,m} \\ &= \lim_{|\boldsymbol{M}| \to \infty} \sum_{w=1}^{V} \frac{M_w}{|\boldsymbol{M}|} \frac{\sum_{m=1}^{M_w} \tilde{U}_{w,m}}{M_w} \\ &= \lim_{|\boldsymbol{M}| \to \infty} \sum_{w=1}^{V} \beta_w \tilde{S}_{M_w}. \end{aligned} \tag{51}$$

According to the Central Limit Theorem, $\tilde{S}_{M_w}$ converges in distribution to a normal random variables as $|\boldsymbol{M}| \to \infty$:

$$\lim_{M_v \to \infty} \tilde{S}_{M_w} \sim N(\mu_w, \frac{\sigma_w^2}{M_w}). \tag{52}$$

Since $\tilde{U}_{v,m}$ are independent random variables for all v and m, $\tilde{S}_{M_v}$ are also independent. Therefore $\tilde{S}_{\boldsymbol{M}}$ will also converge to a normal distributed random variable:

$$\begin{aligned}\lim_{|\boldsymbol{M}|\to\infty} \tilde{S}_{\boldsymbol{M}} &= \lim_{|\boldsymbol{M}|\to\infty} \sum_{w=1}^{V} \beta_w \tilde{S}_w \\ &\sim N(\sum_{w=1}^{V} \beta_w \mu_w, \sum_{w=1}^{V} \beta_w \frac{\sigma_w^2}{M_w}) \\ &\sim N(\mu, \frac{\sigma^2}{|\boldsymbol{M}|}).\end{aligned} \tag{53}$$

To express the blocking probability in terms of this normal distribution, we next establish a relationship between the stationary distributions of $\boldsymbol{U}$ and $\tilde{\boldsymbol{U}}$. Let $\tilde{P}_0$ be the probability of zero state for $\tilde{\boldsymbol{U}}$, then from the product-form stationary distribution of $\boldsymbol{U}$ and $\tilde{\boldsymbol{U}}$ we can get the following scaling relationship between the stationary distribution of $\boldsymbol{U}$ and $\tilde{\boldsymbol{U}}$:

$$\frac{\Pr\{\boldsymbol{U} = \boldsymbol{u}\}}{P_0} = \frac{\Pr\left\{\tilde{\boldsymbol{U}} = \boldsymbol{u}\right\}}{\tilde{P}_0}. \tag{54}$$

From the definition of P_0 and $\tilde{P}_0$, the following relationship exists:

$$\begin{aligned}\frac{P_0}{\tilde{P}_0} &= \Pr\left\{\sum_{w=1}^{V}\sum_{m=1}^{M_w} \tilde{U}_{w,m} \le N\right\}^{-1} \\ &= \Pr\left\{\frac{1}{|\boldsymbol{M}|}\sum_{w=1}^{V}\sum_{m=1}^{M_w} \tilde{U}_{w,m} \le \frac{N}{|\boldsymbol{M}|}\right\}^{-1} \\ &= \Pr\left\{\tilde{S}_{\boldsymbol{M}} \le \frac{N}{|\boldsymbol{M}|}\right\}^{-1}.\end{aligned} \tag{55}$$

Notice in the above relationship, $P_0/\tilde{P}_0$ is determined by the probability distribution of $\tilde{S}_M$. Therefore we can use the large-pool limit of $\tilde{S}_{\boldsymbol{M}}$ to get the following approximation

$$\lim_{|\boldsymbol{M}|\to\infty} \frac{P_0}{\tilde{P}_0} = \left[1 - Q(\frac{N}{|\boldsymbol{M}|})\right]^{-1}, \tag{56}$$

where $q(x)$ and $Q(x)$ are respectively the probability density function (PDF) and cumulative tail distribution of $N(\mu, \frac{\sigma^2}{|\boldsymbol{M}|})$. With these relationships, we can now approximate the blocking probability:

$$\begin{aligned}\lim_{|\boldsymbol{M}|\to\infty} P^{bc} &= \lim_{|\boldsymbol{M}|\to\infty} \Pr\left\{\sum_{w=1}^{V}\sum_{m=1}^{M_w} U_{w,m} = N\right\} \\ &= \lim_{|\boldsymbol{M}|\to\infty} \frac{P_0}{\tilde{P}_0} \Pr\left\{\sum_{w=1}^{V}\sum_{m=1}^{M_w} \tilde{U}_{w,m} = N\right\} \\ &= \lim_{|\boldsymbol{M}|\to\infty} \frac{P_0}{\tilde{P}_0} \Pr\left\{\tilde{S}_{\boldsymbol{M}} = \frac{N}{|\boldsymbol{M}|}\right\} \\ &= \frac{P_0}{\tilde{P}_0}\frac{1}{|\boldsymbol{M}|} q\left(\frac{N}{|\boldsymbol{M}|}\right),\end{aligned} \tag{57}$$

$$\begin{aligned}\lim_{|\boldsymbol{M}|\to\infty} P_v^{br} &= \lim_{|\boldsymbol{M}|\to\infty} \Pr\left\{U_{v,1} = K_v, \sum_{w=1}^{V}\sum_{m=1}^{M_w} U_{w,m} < N\right\} \\ &= \lim_{|\boldsymbol{M}|\to\infty} \frac{P_0}{\tilde{P}_0} \Pr\left\{\tilde{U}_{v,1} = K_v, \sum_{w=1}^{V}\sum_{m=1}^{M_w} \tilde{U}_{w,m} < N\right\} \\ &= \lim_{|\boldsymbol{M}|\to\infty} \frac{P_0}{\tilde{P}_0} \Pr\left\{\tilde{U}_{v,1} = K_v\right\} \\ &\quad \cdot \Pr\left\{\sum_{m=2}^{M_v} \tilde{U}_{v,m} + \sum_{w\ne v}\sum_{m=1}^{M_w} \tilde{U}_{w,m} < N - K_v\right\} \\ &= \lim_{|\boldsymbol{M}|\to\infty} \frac{P_0}{\tilde{P}_0} \tilde{P}_v^{br} \Pr\left\{\tilde{S}_{\boldsymbol{M}} - \frac{\tilde{U}_{v,1}}{|\boldsymbol{M}|-1} < \frac{N-K_v}{|\boldsymbol{M}|-1}\right\} \\ &= \frac{P_0}{\tilde{P}_0}\tilde{P}_v^{br}\left[1 - Q(\frac{N}{|\boldsymbol{M}|})\right].\end{aligned} \tag{58}$$

Notice the fifth equality of (58) holds because, as $|\boldsymbol{M}| \to \infty$, N should also approach infinity as $N > |\boldsymbol{M}|\mu$. Hence

$$\lim_{|\boldsymbol{M}|\to\infty} Q(\frac{N-K_v}{|\boldsymbol{M}|}) = Q(\frac{N}{|\boldsymbol{M}|}). \tag{59}$$

Also, $\lim_{|\boldsymbol{M}|\to\infty} Q(\frac{N}{|\boldsymbol{M}|}) = e^{-\alpha^2/2}$. Therefore, the approximation for the overall session blocking probability of class-v VBSs is

$$\begin{aligned}\lim_{|\boldsymbol{M}|\to\infty} P_v^b &= \lim_{|\boldsymbol{M}|\to\infty} (P^{bc} + P_v^{br}) \\ &= \left[1 - Q(\frac{N}{|\boldsymbol{M}|})\right]^{-1} \\ &\quad \cdot \left\{\frac{1}{|\boldsymbol{M}|} q\left(\frac{N}{|\boldsymbol{M}|}\right) + \tilde{P}_v^{br}\left[1 - Q(\frac{N}{|\boldsymbol{M}|})\right]\right\} \\ &= \frac{\sqrt{|\boldsymbol{M}|}}{|\boldsymbol{M}|\sqrt{2\pi\sigma^2}} \frac{e^{-\alpha^2/2}}{1 - e^{-\alpha^2/2}} + \tilde{P}_v^{br} \\ &= \frac{1}{\sqrt{2\pi|\boldsymbol{M}|\sigma^2}} \frac{1}{e^{\alpha^2/2} - 1} + \tilde{P}_v^{br}.\end{aligned} \tag{60}$$

Appendix D
Proof of Theorem 4

The first part of our proof is straightforward using (53). Since $\tilde{S}_{\boldsymbol{M}}$ will also converge to a normal distributed random variable $N(\mu, \frac{\sigma^2}{|\boldsymbol{M}|})$ as $|\boldsymbol{M}| \to \infty$, according to the strong law of large numbers:

$$\begin{aligned}&\Pr\left\{\lim_{|\boldsymbol{M}|\to\infty} \eta = \frac{|\boldsymbol{M}|\mu}{N}\right\} \\ &= \Pr\left\{\lim_{|\boldsymbol{M}|\to\infty} \frac{\sum_{w=1}^{V}\sum_{m=1}^{M_w} \tilde{U}_{w,m}}{N} = \frac{|\boldsymbol{M}|\mu}{N}\right\} \\ &= \Pr\left\{\lim_{|\boldsymbol{M}|\to\infty} \tilde{S}_{\boldsymbol{M}} = \mu\right\} = 1.\end{aligned} \tag{61}$$

Hence $\eta \xrightarrow{\text{a.s.}} \frac{|\boldsymbol{M}|\mu}{N}$. Also, it is easy to see that $U_{v,m} \leq K_v$ and $\Pr\left\{U_{v,m} < K_v\right\} > 0$. Therefore $\mu_v < K_v$ and

$$\begin{aligned}\mu &= \sum_{w=1}^{V} \mu_w \beta_w \\ &< \sum_{w=1}^{V} K_w \beta_w = \frac{\sum_{w=1}^{V} K_w M_w}{|\boldsymbol{M}|} = \frac{\boldsymbol{M}^T \boldsymbol{K}}{|\boldsymbol{M}|} \\ &< \frac{N}{|\boldsymbol{M}|}. \end{aligned} \tag{62}$$

Thus $\frac{|\boldsymbol{M}|\mu}{N} < 1$.

REFERENCES

[1] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013–2018*, Cisco Systems Inc., San Jose, CA, USA, 2013.

[2] China Mobile Research Institute. (Jun. 2014). *C-RAN: The Road Towards Green RAN (Version 3.0)*. [Online]. Available: http://labs.chinamobile.com/cran/wpcontent/uploads/2014/06/20140613-C-RAN-WP-3.0.pdf

[3] Y. Lin, L. Shao, Z. Zhu, Q. Wang, and R. K. Sabhikhi, "Wireless network cloud: Architecture and system requirements," *IBM J. Res. Develop.*, vol. 54, no. 1, pp. 4:1–4:12, 2010.

[4] *Suggestions on Potential Solutions to C-RAN*, NGMN Alliance, Frankfurt, Germany, 2013.

[5] J. Liu, T. Zhao, S. Zhou, Y. Cheng, and Z. Niu, "CONCERT: A cloud-based architecture for next-generation cellular systems," *IEEE Wireless Commun.*, vol. 21, no. 6, pp. 14–22, Dec. 2014.

[6] O. Simeone, A. Maeder, M. Peng, O. Sahin, and W. Yu. (Dec. 2015). "Cloud radio access network: Virtualizing wireless access for dense heterogeneous systems." [Online]. Available: http://arxiv.org/abs/1512.07743

[7] V.-G. Nguyen, T.-X. Do, and Y. Kim, "SDN and virtualization-based LTE mobile network architectures: A comprehensive survey," *Wireless Pers. Commun.*, vol. 86, no. 3, pp. 1401–1438, 2016.

[8] S. Namba, T. Matsunaka, T. Warabino, S. Kaneko, and Y. Kishi, "Colony-RAN architecture for future cellular network," in *Proc. IEEE Future Netw. Mobile Summit (FutureNetw)*, Jul. 2012, pp. 1–8.

[9] K. Sundaresan, M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy, "FluidNet: A flexible cloud-based radio access network for small cells," in *Proc. ACM 19th Annu. Int. Conf. Mobile Comput. Netw.*, 2013, pp. 99–110.

[10] Z. Zhu *et al.*, "Virtual base station pool: Towards a wireless network cloud for radio access networks," in *Proc. 8th ACM Int. Conf. Comput. Frontiers*, Ischia, Italy, 2011, p. 34.

[11] S. Bhaumik *et al.*, "CloudIQ: A framework for processing base stations in a data center," in *Proc. ACM 18th Annu. Int. Conf. Mobile Comput. Netw.*, Istanbul, Turkey, 2012, pp. 125–136.

[12] Q. Yang *et al.*, "BigStation: Enabling scalable real-time signal processingin large MU-MIMO systems," in *Proc. ACM SIGCOMM Conf.*, Hong Kong, 2013, pp. 399–410.

[13] W. Wu, L. E. Li, A. Panda, and S. Shenker, "PRAN: Programmable radio access networks," in *Proc. 13th ACM Workshop Hot Topics Netw. (HotNets-XIII)*, New York, NY, USA, 2014, pp. 6:1–6:7. [Online]. Available: http://doi.acm.org/10.1145/2670518.2673865

[14] S. Zhou, T. Zhao, Z. Niu, and S. Zhou, "Software-defined hyper-cellular architecture for green and elastic wireless access," *IEEE Commun. Mag.*, vol. 54, no. 1, pp. 12–19, Jan. 2016.

[15] *CPRI Specification v6.0: Interface Specification*, CPRI Coop., 2013.

[16] J. Liu, S. Xu, S. Zhou, and Z. Niu, "Redesigning fronthaul for next-generation networks: Beyond baseband samples and point-to-point links," *IEEE Wireless Commun.*, vol. 22, no. 5, pp. 90–97, Oct. 2015.

[17] T. Werthmann, H. Grob-Lipski, and M. Proebster, "Multiplexing gains achieved in pools of baseband computation units in 4G cellular networks," in *Proc. IEEE 24th Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 3328–3333.

[18] I. Gomez-Miguelez, V. Marojevic, and A. Gelonch, "Deployment and management of SDR cloud computing resources: Problem definition and fundamental limits," *EURASIP J. Wireless Commun. Netw.*, vol. 2013, no. 1, pp. 1–11, 2013.

[19] J. Liu, S. Zhou, J. Gong, Z. Niu, and S. Xu, "On the statistical multiplexing gain of virtual base station pools," in *Proc. IEEE Global Commun. Conf.*, Dec. 2014, pp. 2283–2288.

[20] S. Borst, "User-level performance of channel-aware scheduling algorithms in wireless data networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 3, pp. 636–647, Jun. 2005.

[21] K. Son, H. Kim, Y. Yi, and B. Krishnamachari, "Base station operation and user association mechanisms for energy-delay tradeoffs in green cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 8, pp. 1525–1536, Sep. 2011.

[22] K. W. Ross and D. H. K. Tsang, "The stochastic knapsack problem," *IEEE Trans. Commun.*, vol. 37, no. 7, pp. 740–747, Jul. 1989.

[23] J. M. Aein and O. S. Kosovych, "Satellite capacity allocation," *Proc. IEEE*, vol. 65, no. 3, pp. 332–342, Mar. 1977.

[24] J. Kaufman, "Blocking in a shared resource environment," *IEEE Trans. Commun.*, vol. COM-29, no. 10, pp. 1474–1481, Oct. 1981.

[25] R. W. Wolff, "Poisson arrivals see time averages," *Oper. Res.*, vol. 30, no. 2, pp. 223–231, 1982.

[26] J. Wu, S. Zhou, and Z. Niu, "Traffic-aware base station sleeping control and power matching for energy-delay tradeoffs in green cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 8, pp. 4196–4209, Aug. 2013.

Jingchu Liu (S'14) received the B.S. degree from the Department of Electronic Engineering, Tsinghua University, China, in 2012, where he is currently pursuing the Ph.D. degree with the Department of Electronic Engineering. From 2015 to 2016, he visited the Autonomous Networks Research Group, Ming Hsieh Department of Electrical Engineering, University of Southern California, CA, USA. His research interests include cloud-based wireless networking, data-driven network management, network data analytics, and green wireless communications.

Sheng Zhou (S'06–M'12) received the B.E. and Ph.D. degrees in electronics engineering from Tsinghua University, Beijing, China, in 2005 and 2011, respectively. In 2010, he was a Visiting Student with the Wireless System Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, USA. He is currently an Assistant Professor with the Department of Electronic Engineering, Tsinghua University. His research interests include cross-layer design for multiple antenna systems, cooperative transmission in cellular systems, and green wireless communications. He co-received the Best Paper Award at the Asia-Pacific Conference on Communication in 2009 and 2013, the 23th IEEE International Conference on Communication Technology in 2011, and the 25th International Teletraffic Congress in 2013.

Jie Gong (S'09–M'13) received the B.S. and Ph.D. degrees from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2008 and 2013, respectively. From 2012 to 2013, he visited the Institute of Digital Communications, University of Edinburgh, Edinburgh, U.K. From 2013 to 2015, he was a Post-Doctoral Scholar with the Department of Electronic Engineering, Tsinghua University. He is currently an Associate Research Fellow with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His research interests include cloud RAN, energy harvesting, and green wireless communications. He was a co-recipient of the Best Paper Award from the IEEE Communications Society Asia-Pacific Board in 2013.

Zhisheng Niu (M'98–SM'99–F'12) received the degree from Beijing Jiaotong University, China, in 1985, and the M.E. and D.E. degrees from the Toyohashi University of Technology, Japan, in 1989 and 1992, respectively. From 1992 to 1994, he was with Fujitsu Laboratories Ltd., Japan. In 1994, he joined Tsinghua University, Beijing, China, where he is currently a Professor with the Department of Electronic Engineering. He is also a Guest Chair Professor with Shandong University, China. His major research interests include queuing theory, traffic engineering, mobile Internet, radio resource management of wireless networks, and green communication and networks.

He was an active volunteer for various academic societies, including the Director of Conference Publications (2010-2011) and the Director of the Asia-Pacific Board (2008-2009) of the IEEE Communication Society, a Membership Development Coordinator (2009-2010) of the IEEE Region 10, a Councilor of IEICE-Japan (2009-2011), and a Council Member of the Chinese Institute of Electronics (2006-2011). He was a Distinguished Lecturer (2012-2015) and the Chair of the Emerging Technology Committee (2014-2015) of the IEEE Communication Society, a Distinguished Lecturer (2014-2016) of the IEEE Vehicular Technologies Society, a member of the Fellow Nomination Committee of the IEICE Communication Society (2013-2014), a Standing Committee Member of the Chinese Institute of Communications (CIC) (2012-2016), and the Associate Editor-in-Chief of the *IEEE/CIC China Communications* joint publication.

Dr. Niu is a fellow of IEICE. He received the Outstanding Young Researcher Award from the Natural Science Foundation of China in 2009 and the Best Paper Award from the IEEE Communication Society Asia-Pacific Board in 2013. He co-received the Best Paper Awards from the 13th, 15th, and 19th Asia-Pacific Conference on Communication in 2007, 2009, and 2013, respectively, and the International Conference on Wireless Communications and Signal Processing (2013), and the Best Student Paper Award from the 25th International Teletraffic Congress. He was the Chief Scientist of the National Basic Research Program (so called 973 Project) of China on Fundamental Research on the Energy and Resource Optimized Hyper-Cellular Mobile Communication System (2012-2016), which is the first national project on green communications in China.

Shugong Xu (SM'06–F'16) received the Ph.D. degree from the Huazhong University of Science and Technology, in 1996. He was a Research Director and Principal Scientist with the Communication Technologies Laboratory, Huawei Technologies. He was a Principal Investigator of the Intel Collaborative Research Institute for Mobile Networking and Computing (ICRI-MNC) and co-directed the research programs for this new institute after he joined Intel Corporation in 2013. Among his responsibilities with Huawei Technologies, he founded and directed Huawei's green radio research program GREAT. He was also the Chief Scientist and Lead for the China National 863 project on End-to-End Energy Efficient Networks. Prior to joining Huawei Technologies in 2008, he was with Sharp Laboratories of America as a Senior Research Scientist. He is currently the Director of ICRI-MNC. One of his most referenced papers has over 1200 Google Scholar citations, in which the findings were among the major triggers for the research and standardization of the IEEE 802.11S. He holds over 20 U.S. patents granted. Some of these technologies have been adopted in international standards, including the IEEE 802.11, 3GPP LTE and DLNA. He has authored over 60 peer-reviewed research papers in top international conferences and journals. His recent research interests include mobile networking and computing, next generation wireless communication platform, network intelligence, and SDN/NFV.