This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

Load-Based On/Off Scheduling for Energy-Efficient Delay-Tolerant 5G Networks

Haluk Çelebi, Yavuz Yapıcı, İsmail Güvenç, and Henning Schulzrinne

***Abstract*—Dense deployment of small cells is seen as one of the major approaches for addressing the traffic demands in next-generation 5G wireless networks. The energy efficiency, however, becomes a concern along with the deployment of massive amount of small cells. In this study, we consider the energy-efficient small cell networks (SCN) using smart on/off scheduling (OOS) strategies, where a certain fraction of small base stations (SBS) are put into less energy-consuming sleeping states to save energy. To this end, we first represent the overall SCN traffic by a new *load* variable, and analyze its statistics rigorously using Gamma approximation. We then propose two novel OOS algorithms exploiting this load variable in centralized and distributed fashions. We show that proposed load based OOS algorithms can lead to as high as 50% of energy savings without sacrificing the average SCN throughput. In addition, load based strategies are shown to work well under high SCN traffic and delay-intolerant circumstances, and can be implemented efficiently using the load statistics. We also show that the performance of load based algorithms gets maximized for certain length of sleeping periods, where assuming short sleep periods is as energy-inefficient as keeping SBSs in sleep states for very long.**

***Index Terms*—5G, delay tolerant network (DTN), energy efficiency, sleep mode, small cell network (SCN).**

I. Introduction

Massive densification of small cell networks (SCNs) is commonly seen as one of the major pillars of 5G wireless networks to cope with the ever-increasing mobile data traffic [1], [2]. For such dense deployments of SCNs, developing dynamic cell management and user-access mechanisms are crucial for saving energy at off-peak hours, and for boosting the throughput of the network [3], [4]. Active small base stations (SBSs) not only consume energy, but also increase interference in the communications environment. Therefore, green and energy-efficient strategies that opportunistically put SBSs into sleep mode becomes important for unplanned cell locations, especially with dynamically varying user distributions, spatial load, and traffic load.

The number of SBSs that are required to satisfy the quality of service requirements (QoS) of users change continuously due to the varying traffic demand. In particular, numerous types of user equipment (UEs) such as tablets, mobile phones, gaming consoles, e-readers, and machine type devices cause heterogeneous traffic patterns. This heterogeneous traffic environment necessitates energy-efficient management techniques where SBSs switch to energy-saving sleeping modes when they are not needed, and get activated back whenever the demand is high. The respective energy-saving strategies can be developed taking into account SBS utilization, which takes advantage of flexibility of SBS hardware, and UE's delay tolerance.

Along with increasing SBS density in SCN deployments, an arbitrary UE may be in the communications range of multiple SBSs, which is referred to as *coverage overlapping*. The UEs may therefore not always offload its traffic to the nearest SBS (i.e., it may be in a sleeping mode), and it is also possible to employ other nearby SBSs to this end. Moreover, along with the on/off scheduling (OOS) strategies in the context of energy-efficient SCNs, the set of available SBSs dynamically change over time due to the putting of some SBSs into the energy-saving states. Considering this highly dynamic network topology, we come up with a more suitable traffic load representation, which can easily be updated along with varying set of available SBSs, and take into account all possible set of SBSs (not only the nearest one).

A. Literature Review

There are various studies in the literature considering cell utilization in the context of traffic load description. In [5], cell utilization is modeled as a coupled non-linear function of several parameters involving UE-cell distance, fading, interference, UE's service demand, and load neighboring cells. The respective properties are analytically investigated assuming hexagonal topology. In [6], utilization of a cell is formulated considering location of users, and assuming exponential service time. A load-minimization based energy-saving schemes is proposed in [7] along with implicit formulation on the cell load.

The approach of [5]–[7] basically relies on the assumption of *static UE-cell association* while investigating the cell utilization. The load values therefore need to be recomputed each time the UE-cell association or energy-saving status of an SBS changes, which in turn increases the associated computational complexity. As a result, while these models are aiming at modeling the cell utilization accurately, their complexity makes them lose their practicality especially for the highly dynamic topology of SCNs employing OOS strategies.

The flexibility of SBSs greatly facilitates the realization of energy-saving communications schemes. The energy efficiency of SBSs have been rigorously studied in the literature [4], [8]–[10]. The energy savings, however, can be further improved

H. Çelebi is with the Department of Electrical and Computer Engineering, Columbia University, New York, NY (e-mail: haluk@ee.columbia.edu).

Y. Yapıcı and İ. Güvenç are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC (e-mail: {yyapici,iguvenc}@ncsu.edu).

H. Schulzrinne is with the Department of Computer Science, Columbia University, New York, NY (e-mail: hgs@cs.columbia.edu).

This research was supported in part by NSF under the grant CNS-1814727.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

by integrating energy-efficient sleep-mode techniques with flexible UE access strategies. As an example, the approach in [11] considers to divide the geographical area of the communications network into multiple grids. In each of these grids, the maximum number of SBSs are selected at the peak-traffic times to serve all the UEs satisfactorily. In the idle periods, a subset of these selected SBSs are kept active while the remaining SBSs are turned off. This strategy yields up to energy savings of 53% in dense areas, and of 23% in sparse areas.

The *delay tolerance* of UEs is yet another important feature needs to be considered in designing smart OOS strategies. Considering a commercial jet airplane, the flight data generated within 30 minutes is around 10 TB [12]. If the size of data to be transferred is this large, UE may need to defer its transmission depending on the availability of near-by cells. For a given delay-tolerance budget in a dense SCN, each UE may have the option of either getting service from relatively farther SBSs immediately, or waiting for much closer another SBS to become available. While the delay tolerant networks (DTN) have been studied extensively in the literature [13]–[15], it has not been well explored in the context of energy-efficient SCNs. The study in [16] considers an OOS scheme for a single SBS based on the accumulated tasks, and its delay-energy efficiency aspects are analyzed. In this work, we explore the UE's delay tolerance at a large-scale SCN (instead of a single SBS) employing energy-efficient OOS schemes.

B. Contributions

In this work, which is a rigorously extended version of [17], we study energy-efficient OOS strategies for SBSs in next-generation 5G networks. Considering a user-centric approach, where SBS selection mechanism is managed at UEs, we propose a novel *load* based OOS framework with a promise of more energy-efficient SCNs. Our specific contributions are listed as follows.

- We propose a simple yet effective traffic load definition for dense SCNs involving randomly distributed SBSs and UEs. The status (e.g., sleeping, non-sleeping) of SBSs change dynamically, and UEs may be within the coverage of multiple SBSs. The traffic load of the overall network is represented by a random variable. The distribution of this load variable is derived rigorously, which is verified through extensive simulations.
- Towards achieving energy-efficient SCNs, we propose two load based (LB) OOS algorithms, where certain fraction of SBSs with relatively low load values are put into less energy consuming (i.e., sleeping) states for a random duration of time. In particular, we introduce *centralized* LB (CLB) and *distributed* LB (DLB) as two novel OOS algorithms. Although CLB demands instantaneous load value of *all SBSs*, DLB, instead, requires load value of *much fewer* SBSs by exploiting the *analytical load distribution*. The numerical results verify that CLB and its computationally efficient alternative DLB have very close performance.
- We also consider two benchmark OOS techniques, which are *random on/off* (ROO) and *wake-up control* (WUC). While ROO is a simple baseline algorithm [18], WUC is a more complex sophisticated algorithm requiring full-control of the macro base station (MBS) dynamically. The numerical results verify that CLB and DLB are superior to ROO, and have similar performance as WUC. Furthermore, as the overall SCN traffic increases, WUC turns out to be less energy-efficient than either CLB or DLB.

The rest of this paper is organized as follows. Section II introduces the system model for SCNs with dynamic on/off operation of SBSs. Section III analytically derives the traffic load distribution for a given UE using a Gamma distribution approximation. Section IV proposes the centralized and distributed strategies to conduct on/off operation of SBSs. Section V presents numerical results, and Section VI concludes the paper.

II. System Model

In this section, we first overview the network model, then describe the novel load based model for the network traffic, and finally describe the power consumption model of the SBSs.

A. SCN Model

We consider a densely packed SCN where low-power SBSs are operated to deliver mobile data to UEs of interest. We assume that SBSs and UEs are distributed randomly over a 2-dimensional horizontal plane following the homogeneous Poisson point process (HPPP) with densities ρ_{c} and ρ_{u}, respectively. Each UE is assumed to be able to receive service from any SBS separated by at most the threshold distance R_{th}. In addition, UEs generate traffic at random time intervals, and request to offload a file where the file size and the service request intervals have exponential distribution with rates λ_{F} and λ_{U}, respectively.

Considering that each UE is not involved in transmission all the time (due to exponentially distributed service request times), the energy efficiency of the overall network is desired to be improved by putting some of the SBSs into less energy-consuming (i.e., sleeping) states. Leaving details of sleeping states and the associated OOS strategies to Section IV, each SBS in sleeping states is assigned with a random sleep time T_{s}, which follows exponential distribution with rate λ_{S}. The delayed access strategy under consideration is given in Fig. 1, which assumes that any UE has tolerable delay of at most w_{t} seconds (i.e., *waiting time*). If a UE with active service request finds at least one available (i.e., *idle*) SBS within the threshold distance R_{th} and the waiting time w_{t}, it connects to the best (i.e., *nearest*) of these SBSs to offload its desired traffic. Otherwise, it connects to MBS, and the current service request is assumed to be *blocked* in SCN tier.

In terms of interaction between UEs and SBSs, we assume that UEs do not know the location of sleeping SBSs. But rather, UEs have the perfect knowledge of distance to each non-sleeping SBSs, which can be estimated by monitoring/processing the downlink reference signals from these SBSs. The association between UEs and SBSs is set up such

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

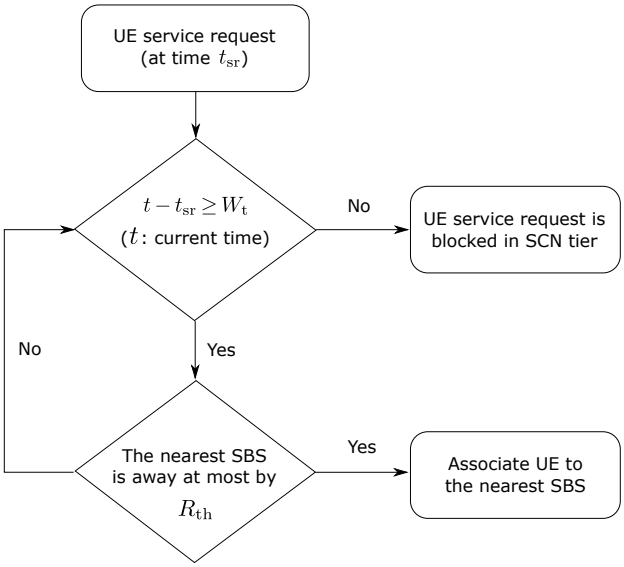


Fig. 1. Delayed access strategy of UEs in SCN. If an SBS is available by at most $R_{\rm th}$, transmission begins. Otherwise, the UE waits till its tolerable delay expires, after which the service request is blocked.

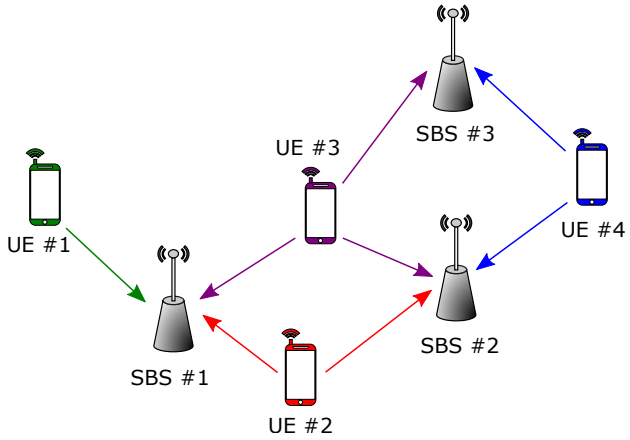


Fig. 2. A representative network of 3 SBSs and 4 UEs. The arrows indicate the SBSs that each UEs can receive service.

that each SBS serves a single UE at a time, and each UE does not change its SBS till the current service request is completely fulfilled. In addition, UEs use all available bandwidth once connected to an SBS, and quickly finish their service resulting in short service times.

B. SBS/UE Densities and Traffic Load

In our SCN, any UE does not not always offload its traffic using the nearest SBS (i.e., it may not be available), and it is also likely to employ other nearby SBSs to this end. Considering this general observation, we propose a more suitable traffic load variable which takes into account all possible set of SBSs, and can easily be updated along with the varying set of available SBSs. Let $n_{\rm c}$ and $n_{\rm u}$ be the range-dependent SBS and UE densities, respectively, which refer to average number of SBSs and UEs within a circular area of radius $R_{\rm th}$. Since location distribution for SBSs and UEs both follow HPPP, the respective Poisson distribution with the range-dependent SBS and UE densities are defined with the mean values $\nu_{\rm c} = \rho_{\rm c}\pi R_{\rm th}^2$ and $\nu_{\rm u} = \rho_{\rm u}\pi R_{\rm th}^2$, respectively. The probability that $n_{\rm c}$ SBSs and $n_{\rm u}$ UEs are present in the circular area of radius $R_{\rm th}$ are therefore given as $p_{\rm c}(i) = P\{n_{\rm c} = i\} = \frac{\nu_{\rm c}^i e^{-\nu_{\rm c}}}{i!}$ and $p_{\rm u}(j) = P\{n_{\rm u} = j\} = \frac{\nu_{\rm u}^j e^{-\nu_{\rm u}}}{j!}$, respectively.

We define the *load factor* for the jth UE as follows:

$$w_j = \begin{cases} \frac{1}{n(j)} & \text{if } n(j) > 0 \\ 0 & \text{if } n(j) = 0 \end{cases}, \tag{1}$$

where $n(j)$ is the number of SBSs that the jth UE can receive service (i.e., away by at most $R_{\rm th}$). Accordingly, *load value* L_i for the ith SBS is defined to be the sum of load factors associated with each UE off the ith SBS by at most a distance of $R_{\rm th}$, and is given as follows:

$$L_i = \sum_{j=1}^{\infty} w_j \mathbb{1}(i,j), \tag{2}$$

where $\mathbb{1}(i,j)$ is the indicator function which is 1 if ith SBS and jth UE are within $R_{\rm th}$ distance, and zero otherwise. Note that although the actual average traffic load to the SBS i is $\lambda_{\rm U}\lambda_{\rm F}L_i$, we instead use L_i directly since the rates for UE's service request ($\lambda_{\rm U}$) and file size to be offloaded ($\lambda_{\rm F}$) are the same for all the UEs across the SCN.

As an example, we consider a representative network given in Fig. 2. Defining $\mathcal{S}_i$ as the indices of SBSs that ith UE can receive service, we have $\mathcal{S}_1 = \{1\}$, $\mathcal{S}_2 = \{1,2\}$, $\mathcal{S}_3 = \{1,2,3\}$, and $\mathcal{S}_4 = \{2,3\}$. Using (1), load factors of UEs are computed as $w_1 = 1$, $w_2 = \frac{1}{2}$, $w_3 = \frac{1}{3}$, $w_4 = \frac{1}{2}$. The respective load values of SBSs are then given using (2) by $L_1 = 1 + \frac{1}{2} + \frac{1}{3} = \frac{11}{6}$, $L_2 = \frac{1}{2} + \frac{1}{3} + \frac{1}{2} = \frac{4}{3}$, and $L_3 = \frac{1}{3} + \frac{1}{2} = \frac{5}{6}$.

We note that instantaneous load value of any SBS possibly varies with the blocked calls, UE-SBS association policies, traffic patterns, and transmission rates in a particular UE-SBS topology. Therefore, the load value of a SBS may not represent exact load distribution perfectly, but is successful enough in giving a good measure of how much traffic any specific SBS handles.

C. Power Consumption Model for SBSs

We now briefly present the power consumption model for an arbitrary SBS, which is an important measure while evaluating the energy efficiency of the overall network. Note that since the transmission range of UEs is very limited, and respective delayed access strategy described in Fig 1 is the same for all OOS strategies, average transmit power of UE is expected to be invariant in all schemes. We therefore do not include the power consumption of UEs in this study, and take into account power consumption of SBSs only.

Considering a standard BS architecture, we assume that the hardware is composed of three blocks: microprocessor (i.e., to manage radio protocols, backhaul connection, etc.), field-programmable gate array (FPGA) (i.e., to process necessary

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

baseband algorithms), radio frequency (RF) front-end (e.g., power amplifiers, transmitter elements, etc.) [18]–[20]. In order to obtain power saving (i.e., *sleeping*) states, OOS strategies consider to turn off a fraction of SBSs not actively engaged in transmission. This can be done by turning off some or all of the hardware blocks, where it takes more time to boot up as more hardware blocks are turned off (i.e., deeper the state is).

TABLE I
SBS States, Boot-Up Times, and Power Consumption Levels

SBS State	Boot-up time (s)	Power Consumption (%)
Active	0	100
Idle	0	50
Standby	0.5	50
Sleep	10	15
Off	30	0

In Table I, we list the SBS states considered in this work together with respective boot-up times and normalized power consumption levels, which are available in the literature [18]–[20]. The description and assumptions for the SBS states are as follows.

- *Active*: The SBS is actively engaging in transmission with full power.
- *Idle*: The SBS is ready to transmit immediately, but not transmitting currently. Hence, RF front-end is not running, and the power consumption is therefore 50% of active state.
- *Standby*: In this light sleep state, the heater for oscillator is turned off intentionally, and RF front-end is not running at all.
- *Sleep*: The SBS is in a deep sleep with only necessary hardware parts (power supply, central processor unit (CPU), etc.) are up.
- *Off*: The SBS is completely offline.

Note that, the sleeping state should be put into either sleep or off states to achieve significant power savings, where the respective minimum boot-up time is 10 seconds. Since any sleeping SBS should be available right after its random sleep time T_s expires, it is not possible to put any SBS into either sleep or off states if $T_s < 10$ seconds. We assume that such SBSs are put into standby state, as shown in Table II, to capture the effect of turning off procedure, and meet the requirement to wake up immediately after T_s seconds. In addition, the power consumption during boot-up period is equal to that of the standby state since that particular SBS does not actively communicate with users.

Although deeper sleeping states provide more power savings, respective longer boot-up times result in UE service requests being blocked more in SCN tier. To effectively handle this fundamental tradeoff between energy consumption and boot-up time, the optimal sleep state should be selected based on UE's delay tolerance, transmit range, and SBS density. We leave the optimal choice of the energy-saving state and sleep time as a future research direction. Instead, we prefer a simple rule which puts each SBS into the deepest state as much as possible for maximum power savings, as given below. Sleep times up to 30 seconds are stand-by or sleep which are determined by hardware limitations. However, if the sleep time is greater than 30 seconds, then, SBS can be either in sleep or off mode. Decision between sleep or off mode is made by minimum power consumption rule by taking into consideration of both the power consumption during boot-up, $p_{\text{boot-up}}$, and sleep mode p_{sleep}.

TABLE II
Sleep State Choice Based on Sleep Time (T_s)

Sleep State	Sleep Time (T_s) (s)
Stand-by	$T_s \leq 10$
Sleep	$10 < T_s \leq 30$
Sleep	$T_s > 30, 10\,p_{\text{boot-up}} + (T_s - 10)\,p_{\text{sleep}} < 30\,p_{\text{boot-up}}$
Off	$T_s > 30, 10\,p_{\text{boot-up}} + (T_s - 10)\,p_{\text{sleep}} > 30 p_{\text{boot-up}}$

III. Analysis of Traffic Load Distribution

In this section, we analyze the distribution of the load variable as a successful measure of the actual traffic loads of SBSs. There are several studies in the literature where fitting distributions are used instead of deriving exact distributions, especially for Poisson Voronoi cell topologies [21]–[23]. Following a similar approach, we analyze distribution of the load variable L by considering the Gamma distribution, which is verified to have satisfactory fitting performance.

The probability density function (PDF) of the gamma distribution can be expressed in terms of shape parameter α and inverse scale parameter β as follows:

$$f(x;\alpha,\beta) = \frac{\beta^{\alpha} x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)}, \tag{3}$$

where $\Gamma(\cdot)$ is the gamma function [24]. Our goal is, therefore, to determine suitable expressions of the gamma parameters α and β in terms of SCN parameters ρ_u, ρ_c, and R_{th}. When the load variable L is assumed to be gamma-distributed with parameters α and β, the first and second moments are given as

$$\mathbb{E}[L] = \frac{\alpha}{\beta}, \qquad \mathbb{E}[L^2] = \frac{\alpha\,(1+\alpha)}{\beta^2}, \tag{4}$$

and the parameters to be determined can be expressed as

$$\alpha = \beta \mathbb{E}[L], \tag{5}$$

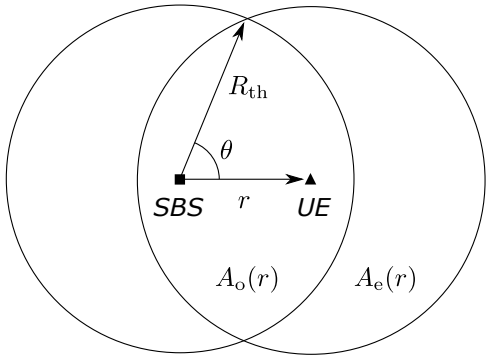
$$\beta = \frac{\mathbb{E}[L]}{\mathbb{E}[L^2] - \mathbb{E}[L]^2}. \tag{6}$$

As a result, the first and the second moments of L completely specifies the desired fitting distribution, and the rest of our analysis is therefore devoted to finding these moments.

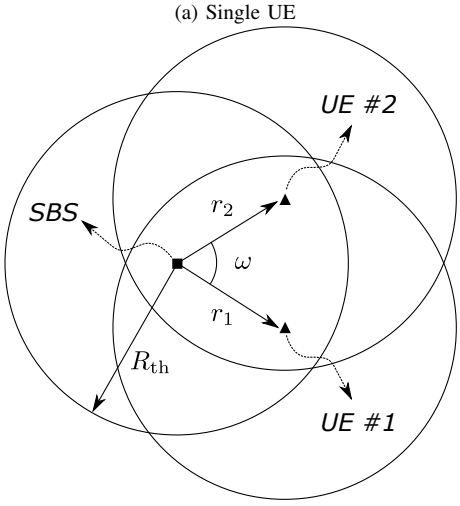
A. First Moment of Load Variable

The first moment of the load variable L for arbitrary SBS in the network is derived by focusing on a representative subnetwork shown in Fig. 3(a). In this framework, the target SBS (for which the load will be computed) is assumed to be located at the origin together with n_c additional SBSs and n_u UEs,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking



(a) Single UE



(b) Two UEs

Fig. 3. A representative SCN involving a single SBS at the origin, and arbitrary UEs off by at most $R_{\rm th}$.

which are distributed randomly over a circular area of radius $R_{\rm th}$.

The first moment of the load L can be expressed as a conditional sum over all possible number of SBSs and UEs as follows:

$$\mathbb{E}\left[L\right] = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \mathbb{E}\left[L \mid n_{\rm c}=i, n_{\rm u}=j\right]\, p_{\rm c}(i)\, p_{\rm u}(j), \tag{7}$$

and using the load definition of (2) in (7) yields

$$\mathbb{E}\left[L\right] = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \mathbb{E}\left[\sum_{k=1}^{j} w_k | n_{\rm c}=i\right]\, p_{\rm c}(i)\, p_{\rm u}(j), \tag{8}$$

$$= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty}\sum_{k=1}^{j} \mathbb{E}[w_k | n_{\rm c}=i]\, p_{\rm c}(i)\, p_{\rm u}(j). \tag{9}$$

We observe that the individual load factors in (9) (i.e., w_k's) are not necessarily the same since the number of SBSs which are away from each UE by at most $R_{\rm th}$ may not be the same. The expected values of the load factors are, however, the same (i.e., $\mathbb{E}[w_k|n_{\rm c}=i] = \mathbb{E}[w|n_{\rm c}=i]$ for $\forall\, k$) since SBSs are distributed uniformly. We may therefore rearrange (9) to obtain

$$\mathbb{E}\left[L\right] = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} j\, \mathbb{E}[w|n_{\rm c}=i]\, p_{\rm c}(i)\, p_{\rm u}(j), \tag{10}$$

$$= \sum_{i=0}^{\infty} \mathbb{E}[w|n_{\rm c}=i]\, p_{\rm c}(i) \sum_{j=1}^{\infty} j\, p_{\rm u}(j). \tag{11}$$

Realizing that the last summation in (11) is the definition of the expected value for the number of user (i.e., $n_{\rm u}$), which is Poisson distributed with rate $\nu_{\rm u}$, we obtain

$$\mathbb{E}\left[L\right] = \nu_{\rm u} \sum_{i=0}^{\infty} \mathbb{E}[w|n_{\rm c}=i]\, p_{\rm c}(i), \tag{12}$$

which reduces to the problem of finding *average traffic load* contributed by a *single UE*.

Theorem 1: The average traffic load contributed by a single UE conditioned on the number of SBSs is given as

$$\mathbb{E}[w|n_{\rm c}=i] = \frac{2}{R_{\rm th}} \sum_{v=0}^{\infty}\sum_{k=0}^{i} \binom{i}{k} \int_0^{R_{\rm th}} \frac{[\nu_e(r)]^v\, e^{-\nu_e(r)}}{(k+v+1)\, v!} \times p_{A_{\rm o}}(r)^k (1-p_{A_{\rm o}}(r))^{i-k} r\,{\rm d}r, \tag{13}$$

where

$$p_{A_{\rm o}}(r) = \frac{2r^2 - \theta + \frac{1}{2}\sin(2\theta)}{\pi R_{\rm th}^2},$$

$$\nu_{\rm e}(r) = \rho_{\rm c}\left(\pi R_{\rm th}^2 - 2r^2 + \theta - \frac{\sin(2\theta)}{2}\right),\ \theta = \cos^{-1}\left(\frac{r}{2R_{\rm th}}\right).$$

Proof: See Appendix A. ∎

Employing (13) and $p_{\rm c}(i) = \frac{\nu_{\rm c}^i e^{-\nu_{\rm c}}}{i!}$ in (12), the first moment of L is readily obtained as

$$\mathbb{E}\left[L\right] = \frac{2\nu_{\rm u}}{R_{\rm th}} \sum_{v=0}^{\infty}\sum_{i=0}^{\infty}\sum_{k=0}^{i} \frac{\nu_{\rm c}^i e^{-\nu_{\rm c}}}{(k+v+1) i! v!} \binom{i}{k} \times \int_0^{R_{\rm th}} [\nu_e(r)]^v\, e^{-\nu_e(r)} p_{A_{\rm o}}(r)^k (1-p_{A_{\rm o}}(r))^{i-k} r\,{\rm d}r, \tag{14}$$

which is a function of the UE density $\nu_{\rm u}$, the SBS density $\nu_{\rm c}$, and the threshold distance $R_{\rm th}$.

B. Second Moment of Load Variable

Following the same approach of (7), the second moment of L can be written as

$$\mathbb{E}\left[L^2\right] = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \mathbb{E}\left[L^2 | n_{\rm c}=i, n_{\rm u}=j\right]\, p_{\rm c}(i)\, p_{\rm u}(j),$$

$$= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \mathbb{E}\left[\left(\sum_{k=1}^{j} w_k\right)^2 \middle| n_{\rm c}=i\right]\, p_{\rm c}(i)\, p_{\rm u}(j), \tag{15}$$

which can be manipulated as follows

$$\mathbb{E}\left[L^2\right] = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \left(\sum_{k=1}^{j} \mathbb{E}\left[w_k^2 | n_{\rm c}=i\right] + \sum_{k=1}^{j}\sum_{\substack{l=1\\ l\neq k}}^{j} \mathbb{E}[w_k w_l | n_{\rm c}=i]\right) \times p_{\rm c}(i)\, p_{\rm u}(j)$$

$$= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \left(j\, \mathbb{E}\left[w^2 | n_{\rm c}=i\right] + j(j-1)\, \mathbb{E}\left[w_k w_l | n_{\rm c}=i\right]\right) \times p_{\rm c}(i) p_{\rm u}(j) \tag{16}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

for any $k \neq l$. Following the discussion in obtaining (12) from (11), and employing first and second-order statistics of the Poisson distribution, we have

$$\sum_{j=1}^{\infty} j(j-1)\, p_{\mathrm{u}}(j) = \mathbb{E}\left[n_{\mathrm{u}}^2\right] - \mathbb{E}\left[n_{\mathrm{u}}\right] = {\nu_{\mathrm{u}}}^2, \tag{17}$$

and (16) accordingly becomes

$$\mathbb{E}\left[L^2\right] = \nu_{\mathrm{u}} \sum_{i=0}^{\infty} \mathbb{E}\left[w^2 | n_{\mathrm{c}} = i\right] p_{\mathrm{c}}(i) + \nu_{\mathrm{u}}^2 \sum_{i=0}^{\infty} \mathbb{E}\left[w_k w_l | n_{\mathrm{c}} = i\right] p_{\mathrm{c}}(i). \tag{18}$$

Theorem 2: The second order moments of the traffic load contributed by a single UE conditioned on the number of SBSs are given as

$$\mathbb{E}\left[w^2 | n_{\mathrm{c}} = i\right] = \frac{2}{R_{\mathrm{th}}} \sum_{v=0}^{\infty} \sum_{k=0}^{i} \binom{i}{k} \int_0^{R_{\mathrm{th}}} \frac{\left[\nu_e(r)\right]^v e^{-\nu_e(r)}}{(k+v+1)^2 v!} \times p_{A_{\mathrm{o}}}(r)^k (1 - p_{A_{\mathrm{o}}}(r))^{i-k} r \mathrm{d}r, \tag{19}$$

$$\mathbb{E}\left[w_k w_l | n_{\mathrm{c}} = i\right] = \frac{4}{R_{\mathrm{th}}^2} \sum_{\nu_1=0}^{\infty} \sum_{\nu_2=0}^{\infty} \sum_{\nu_{\mathrm{c}}=0}^{\infty} \int_0^{R_{\mathrm{th}}} \int_0^{R_{\mathrm{th}}} \int_{-2\pi}^{2\pi} \sum_{m_1=0}^{i} \cdots \sum_{m_{N-1}=0}^{i-\sum_{v=0}^{N-2} m_v} i! \frac{\left[\nu_{\mathrm{e,c}}(r)\right]^{\nu_{\mathrm{c}}} e^{-\nu_{\mathrm{e,c}}(r)}}{\nu_{\mathrm{c}}!\, m_1! \ldots m_N!} \prod_{v=1}^{N} p_{A_v}(\boldsymbol{r}, w)^{m_v} \times \prod_{s=1}^{2} \frac{\left[\nu_{\mathrm{e},s}(r)\right]^{\nu_{\mathrm{s}}} e^{-\nu_{\mathrm{e},s}(r)}}{\nu_{\mathrm{s}}!\, \left(n_{\mathrm{o},s}(\boldsymbol{r}, w) + \nu_s + \nu_{\mathrm{c}} + 1\right)} \times g(\omega)\, r_1 r_2\, \mathrm{d}\omega\, \mathrm{d}r_1 \mathrm{d}r_2, \tag{20}$$

where $p_{A_v}(\boldsymbol{r}, w)$ is the probability of an arbitrary constituent area $A_v(\boldsymbol{r}, w)$, N is the number of constituent areas spanning the disc of radius R_{th} around the origin, $\nu_{\mathrm{e,c}}(r)$ and $\nu_{\mathrm{e},s}(r)$ are the range-dependent densities, and

$$g(\omega) = \begin{cases} \dfrac{\omega}{2\pi(2\pi+1)} & \text{if } \omega \in [-2\pi, 0], \\ \dfrac{1-\omega}{4\pi^2} & \text{if } \omega \in [0, +2\pi]. \end{cases} \tag{21}$$

Proof: See Appendix B. ∎

Incorporating (19) and (20) of Theorem 2 into (18), we obtain the second moment of L is readily obtained as a function of densities ν_{u} and ν_{c}, and the distance R_{th}.

As a result, the respective parameters α and β of the fitting gamma distribution can be computed using the first order moment $\mathbb{E}[L]$ given in (14), and the second order moment $\mathbb{E}[L^2]$ given in (18) (i.e., calculated using (19) and (20)), based on the relations given in (5) and (6). The cumulative distribution function (CDF) of load distribution can therefore be written as

$$F_L(x) = P\{L < x\} = e^{-\nu_{\mathrm{u}}} + (1 - e^{-\nu_{\mathrm{u}}}) \int_{0^+}^{x} \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha-1} e^{-\beta y} \mathrm{d}y, \tag{22}$$

where first term represents the void probability $P\{L = 0\}$ (i.e., no user is around the SBS). Using (22), the respective PDF of load distribution can be written as

$$f_L(x) = \begin{cases} e^{-\nu_{\mathrm{u}}} & \text{if } x = 0 \\ (1 - e^{-\nu_{\mathrm{u}}}) \dfrac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} & \text{if } x > 0 \end{cases}. \tag{23}$$

IV. Load Based On/Off Scheduling

In this section, we study OOS strategies with a goal of having more energy-efficient SCNs. In this respect, we first consider a random OOS algorithm (i.e., ROO) to set up a simple benchmark to evaluate performance of smarter OOS strategies. We then propose two novel load based OOS algorithms, which are called CLB and DLB, and establish a good compromise between energy-efficiency and network throughput. We also consider a more sophisticated OOS strategy, which is called WUC, where the macrocell has the full capability to wake up any sleeping SBSs. Finally, we investigate the computational complexity of each scheme.

Our OOS strategies assume an online topology such that set of available (i.e., non-sleeping) SBSs, which are those in idle and active modes, changes dynamically keeping the total number of active SBSs the same to have a fair comparison across different OOS algorithms. In other words, for each sleeping SBS to wake up, the OOS algorithms choose the *best idle* SBS to turn off. Moreover, we consider a wide range for the fraction of non-sleeping SBSs, which is referred to as *on-ratio*, among the all SBSs in the SCN. As a result although we are turning off an SBS in response to another SBS having waken up recently, we consider the impact of a different number of non-sleeping, and, equivalently, active SBSs, by changing on-ratio as much as possible within its reasonable limit (e.g., $[0.1, 1]$).

We also assume that any UE can get service from the available SBSs, which are either *currently idle* or *become idle* within the waiting time period, as discussed in Section II-A. In particular, ROO, CLB, and DLB strategies assume no capability at the central controller to wake up a sleeping SBS during its random sleep time. The WUC strategy, however, assumes that the central controller can give order to wake up a sleeping SBS to make it available within the waiting time (i.e., which would otherwise not become available).

A. Random On/Off Scheduling

In this strategy, a central controller (e.g., macrocell) turns off a randomly selected idle SBS, and assigns a random sleep time. Each sleeping SBS wakes up automatically after its sleep time expires, and the central controller decides which SBS to turn off in return. The overall procedure is given in Algorithm 1.

B. Centralized Load Based On/Off Scheduling

The CLB can be considered to be the load based alternative of ROO, which operates in a centralized fashion as described in Algorithm 2. In CLB, the central controller turns off the SBS with the minimum instantaneous load value computed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

Algorithm 1 Random On/Off Scheduling

Input: The sleep time of ith SBS has expired
$\text{SBS}_{\text{nextToSleep}} \leftarrow \text{ROO}(i, \mathcal{S}_{\text{all}})$ ▷ $\mathcal{S}_{\text{all}}$ is the set of all SBSs
turn off $\text{SBS}_{\text{nextToSleep}}$
procedure ROO(i,$\mathcal{S}_{\text{all}}$) ▷ ROO algorithm
  $\mathcal{S}_{\text{idle}} \leftarrow \text{find}_{1 \le \ell \le |\mathcal{S}_{\text{all}}|}(\text{ state}(\mathcal{S}_{\text{all}}(\ell)) == \text{idle})$
  $j \leftarrow \text{rand}(1, |\mathcal{S}_{\text{idle}}|)$
  return $\mathcal{S}_{\text{idle}}(j)$
end procedure

using (2) as a response to each SBS that has just waken up. Note that the algorithm needs the load values of idle and only; however, UE shares its instantaneous load factor with the idle and active SBSs because of two reasons: *i*) each active SBS may return to idle status after completion of the transmission, therefore, may be available within w_t time, *ii*) density of non-sleeping SBSs (i.e. idle and active) do not change therefore, distribution of load is can be obtained, which allows implementation of on/off decision in distributed manner.

Algorithm 2 Centralized Load Based On/Off Scheduling

Input: The sleep time of ith SBS has expired
$\text{SBS}_{\text{nextToSleep}} \leftarrow \text{CLB}(i, \mathcal{S}_{\text{all}})$ ▷ $\mathcal{S}_{\text{all}}$ is the set of all SBSs
turn off $\text{SBS}_{\text{nextToSleep}}$
procedure CLB(i,$\mathcal{S}_{\text{all}}$) ▷ CLB algorithm
  $\mathcal{S}_{\text{idle}} \leftarrow \text{find}_{1 \le \ell \le |\mathcal{S}_{\text{all}}|}(\text{ state}(\mathcal{S}_{\text{all}}(\ell)) == \text{idle})$
  compute L_ℓ by (2) for $\ell = 1, \ldots, |\mathcal{S}_{\text{idle}}|$
  $j \leftarrow \text{argmin}_{1 \le \ell \le |\mathcal{S}_{\text{idle}}|} L_\ell$
  return $\mathcal{S}_{\text{idle}}(j)$
end procedure

C. Distributed Load Based On/Off Scheduling

The DLB algorithm is a distributed version of the centralized CLB algorithm, where the overall operation does not need a central controller. In DLB approach, whenever a sleeping SBS is to wake up (i.e., after expiration of its random sleep time), that specific SBS is designated to be the decision-maker to decide the next SBS to be turned off. The decision-maker SBS first determines its all idle first-hop neighbours (i.e., within a distance of at most R_{th}) as the candidate SBSs to be turned off. The instantaneous load values of the candidate SBSs are then collected (e.g., via BS-BS communication using X2 backhaul link [25]), and the one with the minimum instantaneous load is chosen by the decision-maker SBS as the one to turn off next.

An important feature of DLB is the mechanism specifying when to stop searching candidates in a *wider* neighborhood. To this end, the algorithm checks the following relation

$$1 - (1 - P\{L < L_{\min}\})^{|S(k+1)|} < \kappa \tag{24}$$

where $L_{\min}$ is the minimum instantaneous load associated among the SBSs traversed up to k hops, and κ is a threshold probability. Given the cardinality of $|S(k+1)|$ idle SBSs at the next hop, $k+1$, (24) checks the probability of finding an SBS with a lower load than that of $L_{\min}$. Note that (24) can be computed readily using the analytical load CDF in (22). If inequality of (24) is correct, then the algorithm stops searching for a better candidate SBS, and decides to turn off the current candidate. Otherwise, the algorithm widens its search to second-hop neighbours (i.e., those in $2R_{\text{th}}$ distance). Likewise, algorithm continues to widen its search till it becomes less likely to find an SBS with a lower load than that of the existing candidate (i.e., for which (24) turns out to be true). Algorithm stops either maximum search range $k_S R_{\text{th}}$ for an integer k_S is reached or condition (24) is satisfied. The complete procedure is given in Algorithm 3.

Algorithm 3 Distributed Load Based On/Off Scheduling

Input: The sleep time of ith SBS has expired
$\text{SBS}_{\text{nextToSleep}} \leftarrow \text{DLB}(i, \kappa)$
turn off $\text{SBS}_{\text{nextToSleep}}$
procedure DLB(i,κ) ▷ DLB algorithm
  $L_{\min} \leftarrow \infty$, $k \leftarrow 1$,
  while $1 - (1 - F_L(L_{\min}))^{|S(k+1)|} > \kappa$ **do**
    $//\mathcal{S}_{\text{all}}$ is the set of all SBSs
    $\mathcal{S} \leftarrow \text{find}_{1 \le \ell \le |\mathcal{S}_{\text{all}}|}(\text{ dist}(\mathcal{S}_{\text{all}}(\ell), \mathcal{S}_{\text{all}}(i)) \le k R_{\text{th}})$
    $\mathcal{S}_{\text{idle}} \leftarrow \text{find}_{1 \le \ell \le |\mathcal{S}|}(\text{ state}(\mathcal{S}(\ell)) == \text{idle})$
    if $\mathcal{S}_{\text{idle}} \neq \varnothing$ **then**
      compute L_ℓ by (2) for $\ell = 1, \ldots, |\mathcal{S}_{\text{idle}}|$
      $j \leftarrow \text{argmin}_{1 \le \ell \le |\mathcal{S}_{\text{idle}}|} L_\ell$
      $L_{\min} = L_j$
    else
      $k \leftarrow min(k+1, k_S)$
    end if
  end while
  return $\mathcal{S}_{\text{idle}}(j)$
end procedure

Note that the load distribution indeed changes as the SBSs are turned on and off. The stopping criterion of (24), which is based on the load distribution, is, however, still a valid condition. To get some intuition, consider two networks where fixed proportion of SBSs are switched off randomly using LB algorithms. Let L_{R} and L_{LD} be the random load values of any SBS operated by ROO and LB on/off, respectively. For such a scenario, we have

$$\begin{aligned} & P\{L_{\text{LD}} < l\} < P\{L_{\text{R}} < l\} \\ & \iff 1 - P\{L_{\text{LD}} < l\} > 1 - P\{L_{\text{R}} < l\}, \\ & \iff (1 - P\{L_{\text{LD}} < l\})^{|S|} > (1 - P\{L_{\text{R}} < l\})^{|S|}, \\ & \iff 1 - (1 - P\{L_{\text{LD}} < l\})^{|S|} < 1 - (1 - P\{L_{\text{R}} < l\})^{|S|}, \end{aligned} \tag{25}$$

which shows that the stopping criterion in (24) is still a valid condition for the networks operated by LB algorithms. Some tighter bounds on (24) can be found by utilizing order statistics of load distribution, which is left as a future work.

D. Wake-up Control Based On/Off Scheduling

We finally consider a more complex approach, which is called wake-up control (WUC) and given in Algorithm 4. This algorithm is, indeed, very similar to the CLB algorithm, except

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

that the central controller now has the full control to wake up any sleeping SBS (even before the respective sleep time expires). By this way, any of the UE service requests, which could not otherwise be met by available idle SBSs, might be handled by incorporating the sleeping SBSs. To do so, the candidate sleeping SBSs should be within the communication range, and be able to wake up within the tolerable delay of that UE holding the current request. More specifically, the boot-up time of the candidate sleeping SBSs (i.e., given in Table I) should end within the tolerable delay. Note that once the central controller places a wake-up order for the nearest candidate SBS, it is classified as *reserved* to avoid from placing another wake-up order for the same SBS (for another UE request). Although this approach decreases the blocking probability of SCN, the energy consumption is likely to increase since sleeping SBSs getting wake-up orders cannot remain in their low-power consumption states.

Algorithm 4 WUC Based Service Request Handling

Input: UE service request arrival at t_{now} ▷ t_{now} is the current time
$t_{\text{due}} \leftarrow t_{\text{now}} + w_{\text{t}}$
while $t_{\text{now}} \leq t_{\text{due}}$ **do**
    $\text{SBS}_{\text{best}} = \text{WUC}(t_{\text{due}}, t_{\text{now}}, \mathcal{S}_{\text{all}})$ ▷ $\mathcal{S}_{\text{all}}$ is the set of all SBSs
    update t_{now}
end while
if $\text{SBS}_{\text{best}} == \varnothing$ **then**
    service request is blocked
else
    associate UE to SBS_{best}
end if
procedure $\text{WUC}(t_{\text{due}}, t)$ ▷ WUC algorithm
    $\mathcal{S} \leftarrow \text{find}_{1\leq \ell \leq |\mathcal{S}_{\text{all}}|}\,(\,\text{dist}(\mathcal{S}_{\text{all}}(\ell), \text{UE}\,) \leq R_{\text{th}}\,)$
    $\mathcal{S}_{\text{cand}} \leftarrow \text{find}_{1\leq \ell \leq |\mathcal{S}|}\,(\,t + \text{bootupTime}(\mathcal{S}(\ell)) \leq t_{\text{due}}\,)$
    if $\mathcal{S}_{\text{cand}} == \varnothing$ **then**
        return $\varnothing$
    else
        $j \leftarrow \text{argmin}_{1\leq \ell \leq |\mathcal{S}_{\text{cand}}|}\,\text{dist}(\mathcal{S}_{\text{cand}}(\ell), \text{UE}\,)$
        return $\mathcal{S}_{\text{cand}}(j)$
    end if
end procedure

E. Computational Complexity of the OOS Schemes

In this section, we investigate the computational complexity of the OOS schemes considered in this work. To this end, we assume a large circular area with a radius of $k_{\text{m}}R_{\text{th}}$ such that $A = \pi\,(k_{\text{m}}R_{\text{th}})^2$. This area is populated with N_{c} SBSs and N_{u} UEs such that and $\rho_{\text{c}} = \frac{N_{\text{c}}}{A}$ and $\rho_{\text{u}} = \frac{N_{\text{u}}}{A}$. Let Π_{I}, Π_{S}, and Π_{A} are the probability of any SBS being in the *idle*, *sleep*, and *active* modes, respectively. The computational complexity of each OOS scheme can then be given as follows.

1) ROO Algorithm: In the ROO algorithm, the computational complexity is equivalent to the time complexity of generating a random number, which is $O(1)$.

2) CLB Algorithm: In the CLB algorithm, the central controller chooses the SBS among the set of all idle SBS, which is $\Pi_{\text{I}}N_C$. Sorting SBSs based on their loads, and selecting the minimum one has a complexity of $O((\Pi_{\text{I}}N_C)^2)$, which is smaller than $O((1-\Pi_{\text{S}})^2N_C^2)$ since $\Pi_{\text{I}} = 1 - \Pi_{\text{S}} - \Pi_{\text{A}} < 1 - \Pi_{\text{S}}$.

3) LBD Algorithm: In the LBD algorithm, the local breadth-first search (BFS) is bounded either by the maximum search range $k_{\text{S}}R_{\text{th}}$, or by the load-based stopping criterion given by (24) before reaching the maximum search range. In the worst case scenario, local BFS therefore reaches the maximum search range. The upper bound on the computational complexity can then be found without considering the load-based stopping criterion. Assuming that N_S be the mean number of traversed idle SBSs during the local BFS, we have

$$N_S \leq \rho_{\text{c}}\Pi_{\text{I}}(k_{\text{S}}R_{\text{th}})^2 \tag{26}$$

$$< \rho_{\text{c}}(1-\Pi_{\text{S}})(k_{\text{S}}R_{\text{th}})^2 = (1-\Pi_{\text{S}})\left(\frac{k_s}{k_{\text{m}}}\right)^2 N_{\text{c}}, \tag{27}$$

where (26) is formulated based on the observation that the load-based stopping criterion may be satisfied before the maximum search range is reached.

It is well-known that the BFS has a time complexity of $O(N_{\text{S}}+E_{\text{S}})$, where E_{S} is the number of edges, or, equivalently, the number of neighborhoods between the SBSs. We need the average degree of SBSs, (i.e. average number of SBSs within SBS's communication range) , $\nu_{\text{c}_{\text{LBD}}}$, to compute E_{S}. Note that the sum of load factors for each UE is 1 by definition of (1). Except UE's with no neighboring SBS, sum of load factors UEs is equal to the sum of load values of SBSs. If we assume that no SBS is turned off,

$$(1-e^{-\nu_c})N_{\text{u}} \approx \sum_{i=1}^{N_c} L(i), \tag{28}$$

where $(1-e^{-\nu_c})$ is due to the UEs having no SBS in the communication range. If A is very large (28) holds with equality, and the respective approximation is due to the edge effects. Similarly, if Π_{S} portion of the SBSs are turned off randomly

$$(1-e^{-\nu_c(1-\Pi_{\text{S}})})N_{\text{u}} \approx \sum_{i=1}^{N_c(1-\Pi_{\text{S}})} L(i), \tag{29}$$

$$< \sum_{i=1}^{N_c(1-\Pi_{\text{S}})} L_{\text{LBD}}(i), \tag{30}$$

where (30) is due to the fact that the LBD algorithm keeps the SBSs having larger load values in idle mode. We therefore observe that the only way for the approximation in (29) to hold is to increase ν_{c}. The average SBS degree therefore increases with LBD, and $\nu_{\text{c}_{\text{LBD}}}$ satisfies

$$(1-\Pi_{\text{S}})\nu_{\text{c}} \leq \nu_{\text{c}_{\text{LBD}}} \leq \nu_{\text{c}}. \tag{31}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

A bound on E_{S} can then be obtained as

$$E_{\mathrm{S}} = \frac{1}{2}\sum_{i=1}^{N_{\mathrm{S}}} \text{degree (i)} \tag{32}$$

$$= \mathbb{E}[N_{\mathrm{S}}]\mathbb{E}[\text{SBS degree}] \tag{33}$$

$$< \frac{1}{2}\nu_{\mathrm{c_{LBD}}}(1-\Pi_{\mathrm{S}})\left(\frac{k_s}{k_{\mathrm{m}}}\right)^2 N_{\mathrm{c}}, \tag{34}$$

$$< \frac{1}{2}\nu_{\mathrm{c}}(1-\Pi_{\mathrm{S}})\left(\frac{k_s}{k_{\mathrm{m}}}\right)^2 N_{\mathrm{c}}, \tag{35}$$

$$= \frac{1}{2}(1-\Pi_{\mathrm{S}})\frac{k_s^2}{k_{\mathrm{m}}^4}N_{\mathrm{c}}^2, \tag{36}$$

where the coefficient $1/2$ in (32) is due to counting each SBS twice for single SBS neighborhood, (33) is due to the independent locations of SBSs in HPPP, (34) follows from (27), and, finally, (35) is due to (31).

SBS reaches average of $\nu_c\Pi_{\mathrm{I}}$ SBS immediately, which has mean sorting complexity $(\nu_c\Pi_{\mathrm{I}})^2$. If stopping condition (24) is not satisfied, further SBSs reached by range expansion will be evaluated. Since the initial sorting is done, adding a new load value to a sorted array has linear complexity with array size. Besides, in our algorithm, maximum search range can expanded at most by R_{th} at one time. If current search range is $(i-1)R_{\mathrm{th}}$, and algorithm is reaching new idle SBSs in iR_{th} range, expected sorting complexity can be written as

$$O(S_{LBD}) = (\nu_c\Pi_{\mathrm{I}})^2 + \sum_{i=2}^{k_{\mathrm{S}}}\sum_{j=0}^{\infty}\sum_{t=1}^{\infty}\sum_{m=0}^{t-1}(j+m)p_c(j)p_c(t) \tag{37}$$

$$= (\nu_c\Pi_{\mathrm{I}})^2 + \sum_{i=2}^{k_{\mathrm{S}}}\sum_{j=0}^{\infty}\sum_{t=1}^{\infty}\left(jt + \frac{t(t-1)}{2}\right)p_c(j)p_c(t) \tag{38}$$

$$= (\nu_c\Pi_{\mathrm{I}})^2 + \sum_{i=2}^{k_{\mathrm{S}}}\Bigg[\sum_{j=0}^{\infty}\sum_{t=0}^{\infty} jtp_c(j)p_c(t) + \sum_{j=0}^{\infty}p_c(j)\sum_{t=0}^{\infty}\frac{1}{2}t(t-1)p_c(t)\Bigg] \tag{39}$$

where $p_c(j)$ is Poisson with mean $(i-1)^2\Pi_{\mathrm{I}}\nu_c$. Since between radius $(i-1)R_{\mathrm{th}}$ and iR_{th}, $(2i-1)\Pi_{\mathrm{I}}\nu_c$, there are $(i^2-(i-1)^2)\nu_c\Pi_{\mathrm{I}}$ SBSs, $p_c(t)$ is Poisson with mean $(2i-1)\nu_c\Pi_{\mathrm{I}}$. Note that $p_c(j)$ and $p_c(t)$ are independent due to the disjoint areas. Then, (39) can be rewritten as

$$O(S_{LBD}) = (\nu_c\Pi_{\mathrm{I}})^2 + \sum_{i=2}^{k_{\mathrm{S}}}\left[(2i^3-i^2)(\Pi_{\mathrm{I}}\nu_c)^2 + \frac{1}{2}((2i-1)^2(\nu_c\Pi_{\mathrm{I}})^2)\right] \tag{40}$$

$$= (\nu_c\Pi_{\mathrm{I}})^2 + (\nu_c\Pi_{\mathrm{I}})^2\sum_{i=2}^{k_{\mathrm{S}}}(2i^3+i^2-2i+\frac{1}{2}) \tag{41}$$

$$\approx O(k_{\mathrm{S}}^4(\nu_c\Pi_{\mathrm{I}})^2) \tag{42}$$

$$= O(k_{\mathrm{S}}^4\Pi_{\mathrm{I}}^2\frac{N_c^2}{k_{\mathrm{m}}^4}) < O((1-\Pi_{\mathrm{S}})^2\left(\frac{k_{\mathrm{S}}}{k_{\mathrm{m}}}\right)^4 N_c^2) \tag{43}$$

Note that the DLB algorithm has a reasonable complexity for $\frac{k_s^4}{k_{\mathrm{m}}^4} \ll 1$ since it becomes more likely to find an SBS with sufficiently low load value without having to search through the entire SCN.

4) WUC Algorithm: In the WUC algorithm, central controller wakes up an SBS within the communication range of UE, which takes $O(\Pi_{\mathrm{S}}\nu_{\mathrm{c}}) = O(\Pi_{\mathrm{S}}\frac{N_{\mathrm{c}}}{k_{\mathrm{m}}^2})$ steps.

V. Simulation Results

In this section, we present numerical results for the performance of i) proposed load definition in representing the actual traffic load of SCN, and ii) novel load based OOS strategies. In particular, performance of the novel CLB and DLB algorithms are evaluated in comparison to the ROO and WUC algorithms as the benchmark OOS strategies, and the static topology without dynamic OOS approach. We assume a circular area with a radius of 250 m for the deployment of UEs and SBSs, and the results are averaged over 1000 iterations and 10000 seconds of simulation time. In terms of overall SCN traffic, we consider two main scenarios: low network utilization (1%) and (relatively) high network utilization (20%). In both scenarios, UE traffic profile (i.e., service request rate and associated file size) is assumed to be adequate so that there is enough room to effectively apply OOS strategies (i.e, all SBSs would otherwise be occupied all the time). For delayed access scheme, we assume a sufficiently large but reasonable UE delay tolerance of 60 sec (as well as zero tolerable delay), which enables WUC algorithm to attain its best performance, and, hence, the performance gap between WUC and other strategies becomes apparent. All the simulation parameters are listed in Table III.

TABLE III
Simulation Parameters

Parameter	Value
SBS density (ρ_{c})	0.0005 m^{-2}
User density (ρ_{u})	0.0005 m^{-2}
Service request rate (λ_{U})	$\{0.001, 0.01\}$ s^{-1}
Average file size ($1/\lambda_{\mathrm{F}}$)	$\{1, 2\}$ MB
Sleep rate (λ_{S})	$\{0.001, 0.002\}$ s^{-1}
Tolerable delay (w_{t})	$\{0, 60\}$ s
Threshold distance (R_{th})	50 m
Bandwidth (BW)	1 MHz
Signal-to-Noise Ratio (SNR)	20 dB
Threshold probability for DLB (κ)	0.3
Maximum search range for DLB	$3 \times R_{\mathrm{th}}$
Path loss exponent (α)	4

A. Performance Metrics

In the performance analysis, we consider the following criteria.

- *Blocking Probability*: The fraction of rejected service requests among all, which is basically due to sleeping or fully occupied (i.e., actively transmitting) SBSs, which is given as

$$\mathsf{P}_{\mathsf{block}} = \frac{\text{number of rejected service requests}}{\text{total number of service requests}}. \tag{44}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

- *Average Throughput*: The total number of bits transmitted averaged over the total simulation time, which is also normalized with respect to the number of users as follows

$$\mathsf{R_{SCN}} = \frac{\text{total number of transmitted bits}}{\text{number of users} \times \text{simulation time}} \text{(bps)}. \quad (45)$$

The number of transmitted bits in (45) is given by the Shannon capacity formula as follows

$$\mathsf{R} = \mathsf{BW} \log_2 (1 + \mathsf{SINR}), \quad (46)$$

where BW is the transmission bandwidth, and SINR is the signal-to-interference-plus-noise ratio. Assuming the association between ith UE and jth SBS, the respective SINR at the UE side is defined as follows

$$\mathsf{SINR}_{ij} = \frac{d_{ij}^{-\alpha}}{\sum_{\ell \neq j} d_{i\ell}^{-\alpha} + 1/\mathsf{SNR}}, \quad (47)$$

where d_{ij} is the distance between ith UE and jth active SBS, α is the path loss (PL) exponent, and SNR is the signal-to-noise ratio.

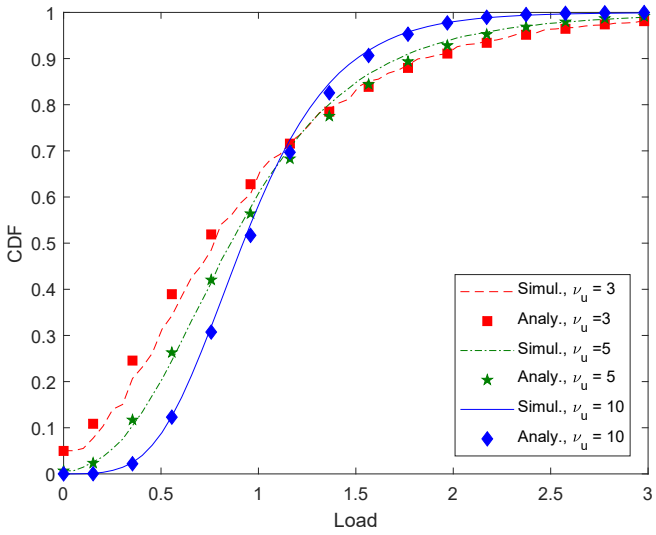
- *Normalized Energy Efficiency*: The amount of energy consumed for each transmitted bit averaged over the total simulation time, which is also normalized by the number of users and the maximum power $\mathsf{P_{max}}$ associated with the active state.

$$\mathsf{EE} = \frac{\mathsf{R_{SCN}}}{\text{total energy consumption}} \times \mathsf{P_{max}} \text{(bps/joule)}. \quad (48)$$

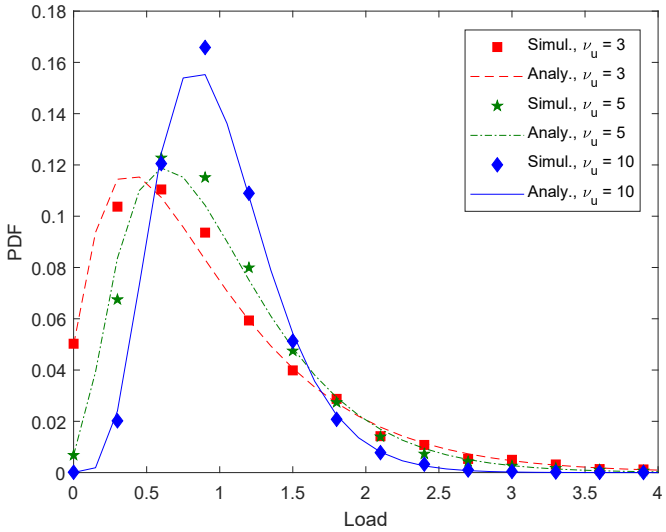
Note that the power consumption of an SBS at each state is given in Table I as the fraction of the maximum power $\mathsf{P_{max}}$, and we therefore use these power fractions while computing (48).

B. Load Distribution Verification

In Fig. 4, we depict the CDF and PDF of the SCN traffic load for range-dependent UE densities of $\nu_{\mathrm{u}} = \{3, 5, 10\}$, where extensive simulation results are provided along with the analytical results computed using (22). We observe that the analytical results nicely match the simulations for all three UE densities, which verifies the respective derivation in Section III. Accuracy of approximation depends on achievable precision of load values which ultimately depends the precision of load factor. Precision of load increment improves as load factor becomes small. Therefore, as we increase UE's range, load factor of UE becomes smaller, and the approximate load distribution converges to the exact load distribution. We, therefore, start observing finely matched load distribution in low load regime as the average number of UEs around SBS increases.



(a) CDF



(b) PDF

Fig. 4. Analytical and simulation results for load distribution for range-dependent UE densities of $\nu_{\mathrm{u}} = \{3, 5, 10\}$ and $\rho_{\mathrm{c}}/\rho_{\mathrm{u}} = 1$.

C. Low Utilization Performance

In this subsection, we consider the performance of OOS strategies under a low network utilization scenario, where the UE service request rate and average file size are $1/\lambda_{\mathrm{U}} = 1000\,\mathrm{s}$ and $1/\lambda_{\mathrm{F}} = 1\,\mathrm{MB}$, respectively. Together with the UE and SBS densities given in Table III, respective network utilization is on the order of 1% based on the utilization results that we have left out due to the space limitations. In Fig. 5, we present blocking probability results for all the algorithms under consideration against varying on-ratio. In particular, we take into account the effect of sleep rate λ_{S} (or equivalently sleep period $1/\lambda_{\mathrm{S}}$) and waiting time w_{t} by assuming $1/\lambda_{\mathrm{S}} \in \{500\,\mathrm{s}, 1000\,\mathrm{s}, \infty\}$ and $w_{\mathrm{t}} \in \{0, 60\,\mathrm{s}\}$ as well as a deterministic sleep time of $T_s = 1000\,\mathrm{s}$. Note that $1/\lambda_{\mathrm{S}} \to \infty$ corresponds to a scenario with no dynamic OOS events, i.e., topology of non-sleeping SBSs does not change once it is initialized at the beginning. We therefore describe the respective load based algorithm simply with LB since either centralized or distributed strategy (i.e., in CLB and DLB) is only applicable with dynamic on/off events occurring after

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

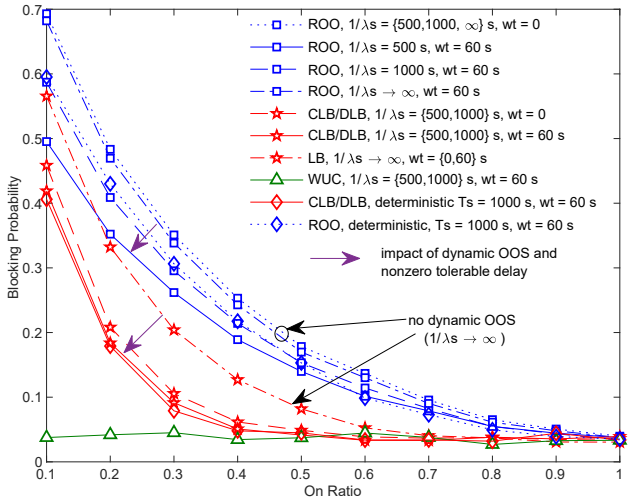
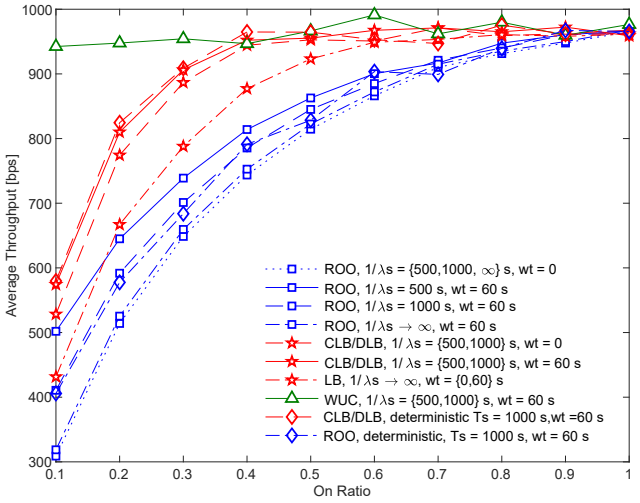


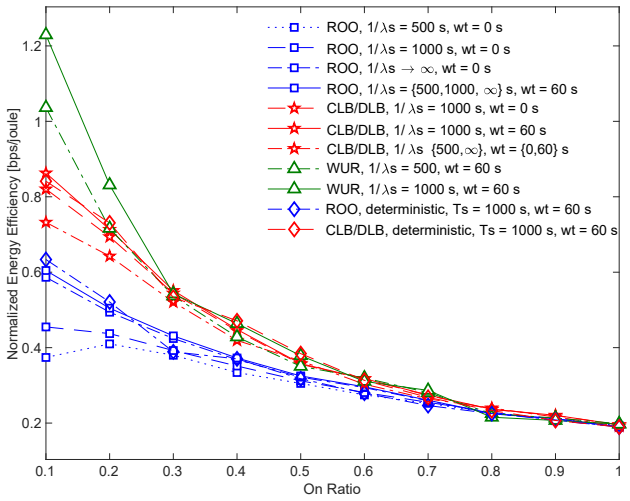
Fig. 5. Blocking probability $\mathsf{P}_{\mathsf{block}}$ along with on ratio (i.e., fraction of non-sleeping SBSs) for $1/\lambda_{\mathrm{S}} \in \{500\,\mathrm{s}, 1000\,\mathrm{s}, \infty\}$ and $w_{\mathrm{t}} \in \{0, 60\,\mathrm{s}\}$ assuming low network utilization of 1% (i.e., $1/\lambda_{\mathrm{U}} = 1000\,\mathrm{s}$ and $1/\lambda_{\mathrm{F}} = 1\,\mathrm{MB}$).

initialization.

We observe in Fig. 5 that blocking probabilities for any OOS algorithm decrease as either more SBSs become available (i.e., increasing on ratio), or tolerable delay gets larger (i.e., more room to meet UE service request). In particular, the load based CLB and DLB perform much better than the random scheme ROO in terms of achieving less blocking events (i.e., rejected UE requests). Note that CLB and DLB actually have the same performance for any choice of on ratio, and we therefore referred to this common performance as CLB/DLB. This equity underscores the power of DLB especially for large-scale SCNs in the sense that DLB does not need information of *all* SBSs (i.e., in contrast to CLB) to decide the next SBS to turn off, and is hence more efficient to implement. Considering a wide range of reasonable non-sleeping SBS fractions (i.e., greater than 0.5 for a realistic SCN), CLB/DLB is shown to attain the performance of more complex WUC scheme, where ROO still falls short of that level.

In Fig. 5, the response of random and load based algorithms to the choice of sleep period $1/\lambda_{\mathrm{S}}$ and waiting time w_{t} are observed to have some interesting differences. Assuming zero tolerable delay (i.e., $w_{\mathrm{t}} = 0$), the blocking probability of random scheme ROO does not change at all along with $1/\lambda_{\mathrm{S}}$ even considering the no dynamic OOS case (i.e., $1/\lambda_{\mathrm{S}} \to \infty$). When we consider nonzero tolerable delay (i.e., $w_{\mathrm{t}} = 60\,\mathrm{s}$), we start observing significant performance improvement in ROO along with decreasing $1/\lambda_{\mathrm{S}}$, where the best performance occurs at $1/\lambda_{\mathrm{S}} = 500\,\mathrm{s}$. On the other hand, load based CLB/DLB achieves significantly better performance for $1/\lambda_{\mathrm{S}} = \{500\,\mathrm{s}, 1000\,\mathrm{s}\}$ (as compared to no dynamic OOS case) even under zero tolerable delay condition. When a nonzero tolerable delay (i.e., $w_{\mathrm{t}} = 60$ s) is further assumed, the best performance is even superior to that of the zero tolerable delay, but the respective performance gap remains marginal.





(b) Normalized Energy Efficiency, EE

Fig. 6. Average throughput and normalized energy efficiency along with on ratio (i.e., fraction of non-sleeping SBSs) for $1/\lambda_{\mathrm{S}} \in \{500\,\mathrm{s}, 1000\,\mathrm{s}, \infty\}$ and $w_{\mathrm{t}} \in \{0, 60\,\mathrm{s}\}$ assuming low network utilization of 1% (i.e., $1/\lambda_{\mathrm{U}} = 1000\,\mathrm{s}$ and $1/\lambda_{\mathrm{F}} = 1\,\mathrm{MB}$).

As a result, CLB/DLB is more robust to *delay intolerance* while random scheme ROO requires *longer tolerable delays* for performance improvement. In addition, applying OOS dynamically is useful for ROO only when the delay tolerance is sufficiently large, while dynamic OOS improves performance of CLB/DLB in both delay tolerant and intolerant SCNs. As a final remark, the deterministic sleep time of $T_s = 1000\,\mathrm{s}$ does not cause any major change in the performance of the algorithms.

In Fig. 6, we present the respective network throughput and normalized energy efficiency results. We observe that the network throughput performance in Fig. 6(a) shows closely related behavior to the blocking probability results (i.e., network throughput increases with decreasing blocking probability, and vice versa). In particular, we observe no significant average

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

throughput loss when as many as 40% of SBSs are in sleeping states. On the other hand, the average throughput of ROO keeps decreasing continuously as more SBSs are put into sleeping states, which finally reads as high as 20% throughput loss for non-sleeping SBSs fraction of 40%.

The normalized energy efficiency results in Fig. 6(b) involve some interesting conclusions as follows. 1) Energy efficiency of ROO is worse than that of CLB/DLB whereas CLB/DLB is as energy-efficient as the more complex WUC scheme for non-sleeping SBS fractions greater than 30%. 2) Although ROO attains the maximum throughput only under nonzero tolerable delay (see Fig. 6(a)), the maximum energy efficiency can be achieved under both zero and nonzero tolerable delays. In particular, while the maximum energy efficiency of ROO is invariant to sleep period under nonzero tolerable delay, the best sleep period turns out to be $\lambda_S \to \infty$ under zero tolerable delay. As a result, *the energy efficiency for ROO under zero tolerable delay gets maximized when OOS scheme is not applied dynamically* (i.e., no on/off events after initialization). 3) Although network throughput for CLB/DLB is maximized for $1/\lambda_S \in \{500\,\text{s}, 1000\,\text{s}\}$ with a significant gap between the no dynamic OOS case (i.e., $\lambda_S \to \infty$), the energy efficiency gets maximized only for $1/\lambda_S = 1000\,\text{s}$ under any choice of tolerable delay. *Regardless of the particular tolerable delay in CLB/DLB, assigning short sleep time is therefore as energy inefficient as keeping SBSs in sleep states for very long*, which identifies an optimal sleep period in between. As for the blocking probability results, the deterministic sleep time of $T_s = 1000\,\text{s}$ does not cause any major change in the performance of the algorithms.

D. High Utilization Performance

We now consider a high network utilization scenario with the UE service request rate of $1/\lambda_U = 100\,\text{s}$ and the average file size of $1/\lambda_F = 1\,\text{MB}$. The respective utilization is on the order of 20%. We assume a representative finite sleep time period together with no dynamic OOS case, i.e., $1/\lambda_S \in \{1000\,\text{s}, \infty\}$, together with both zero and nonzero tolerable delays, i.e., $w_t \in \{0, 60\,\text{s}\}$. In Fig. 7, we present blocking probability results along with on ratio. As before, we observe that the performances of CLB and DLB are much better than that of ROO, and are the same as that of WUC whenever at least 50% of the SBSs are non-sleeping. In addition, DLB has a close performance to CLB, as before. We also observe that the performance of any OOS algorithm improves together with either nonzero tolerable delay, or applying dynamic OOS (i.e., $1/\lambda_S = 1000\,\text{s}$ instead of $1/\lambda_S \to \infty$) on top of that. Regardless of the particular OOS strategy, the blocking probabilities are observed to be higher than those in Fig. 5 as the fraction of non-sleeping SBSs decreases, which is basically due to the increased network utilization.

In order to give some intuition into the impact of any estimation error in the relative distance between SBSs and UEs, we consider a scenario in which UE location is known with some error. More specifically, the coordinate (x, y) of a UE is used as $(x + e_x, y + e_y)$ within the computations, where e_x and e_y are uncorrelated Gaussian noise with zero-mean and variance

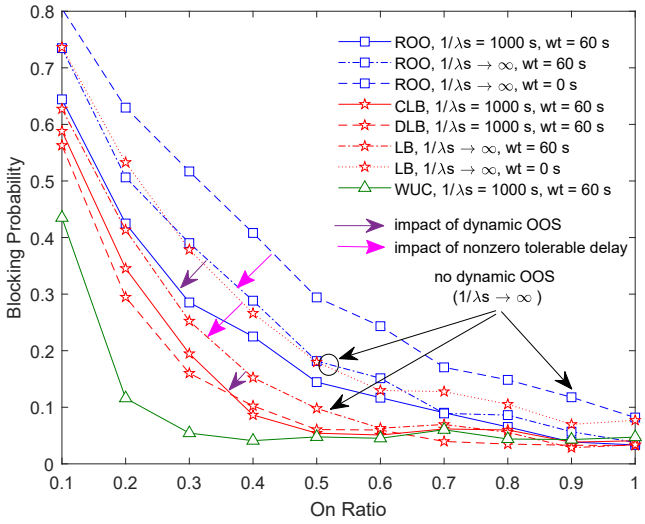


Fig. 7. Blocking probability $\mathsf{P}_{\mathsf{block}}$ along with on ratio (i.e., fraction of non-sleeping SBSs) for $1/\lambda_S \in \{1000\,\text{s}, \infty\}$ and $w_t \in \{0, 60\,\text{s}\}$ assuming high network utilization of 20% (i.e., $1/\lambda_U = 100\,\text{s}$ and $1/\lambda_F = 2\,\text{MB}$).

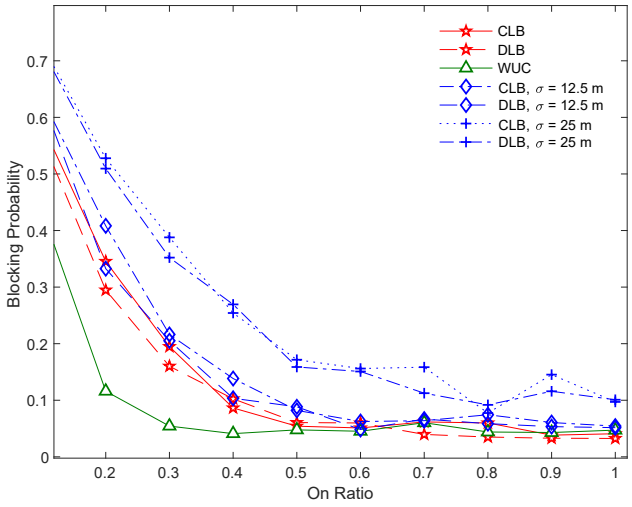
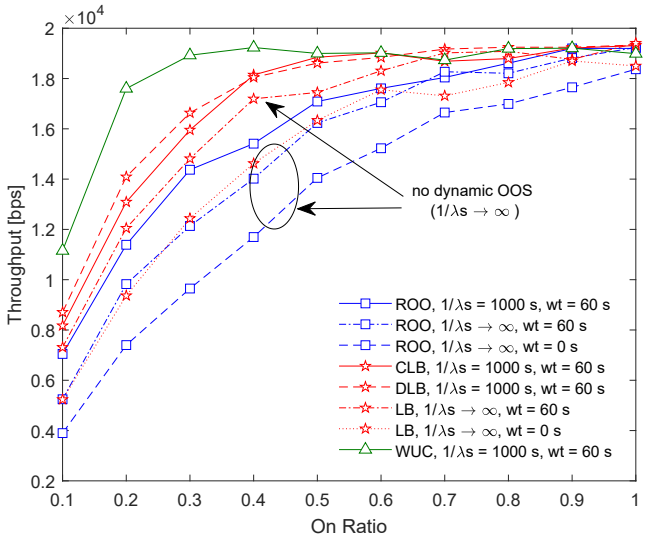


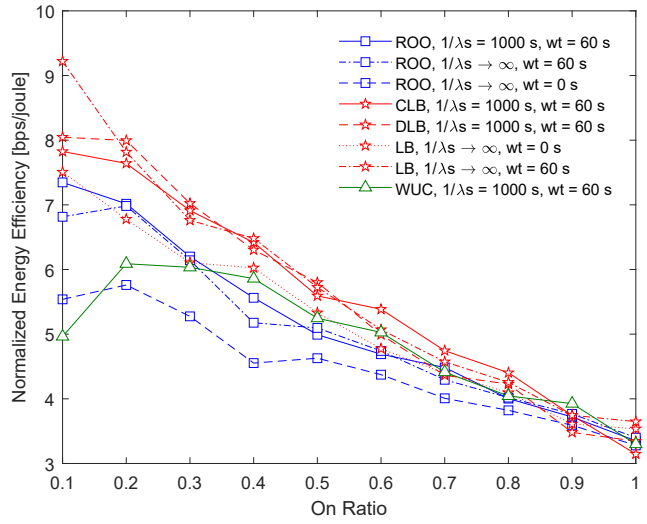
Fig. 8. Blocking probability $\mathsf{P}_{\mathsf{block}}$ along with on ratio (i.e., fraction of non-sleeping SBSs) for $1/\lambda_S = 1000\,\text{s}$ and $w_t = 60\,\text{s}$ assuming high network utilization of 20% (i.e., $1/\lambda_U = 100\,\text{s}$ and $1/\lambda_F = 2\,\text{MB}$), and location error with the standard deviation of $\sigma \in \{12.5\,\text{m}, 25\,\text{m}\}$.

σ^2. We depict the blocking probability of such a scenario on Fig. 8 using the same setting of Fig. 7, except the estimation error. We observe that the blocking probability deteriorates wits increasing estimation error (i.e., $\sigma \in \{12.5\,\text{m}, 25\,\text{m}\}$). The reason for this behavior is that UEs start demanding service from farther SBSs along with increasing location error, and the service time therefore becomes longer (due to decreasing throughput) which in turn leads to more service requests being blocked.

In Fig. 9, we demonstrate the average throughput and normalized energy efficiency performances against on ratio. As

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking





(b) Normalized Energy Efficiency, EE

Fig. 9. Average throughput and normalized energy efficiency along with on ratio (i.e., fraction of non-sleeping SBSs) for $1/\lambda_S \in \{1000\,\text{s}, \infty\}$ and $w_t \in \{0, 60\,\text{s}\}$ assuming high network utilization of 20% (i.e., $1/\lambda_U = 100\,\text{s}$ and $1/\lambda_F = 2\,\text{MB}$).

before, the average throughput results in Fig. 9(a) indicate that the performance of CLB and DLB are much better than that of ROO, and are the same as WUC for a broad range of non-sleeping SBS fractions (i.e., greater than 0.5). In particular, the average throughput of either CLB or DLB remains almost unchanged even when 50% of the SBSs are put into sleeping states, while the respective loss in ROO throughput appears to be between 10%-30% for the same on ratio. Note that the average throughput results in Fig. 9(a) are much higher as compared to that of Fig. 6(a) owing to the increased network utilization. In addition, the average throughput increases for all the OOS algorithms as UEs become more delay tolerant.

We also present the respective normalized energy efficiency results in Fig. 9(b) for this high utilization scenario. We observe that the energy efficiency of CLB and DLB gets maximized with the nonzero tolerable delay (i.e., $w_t \in 60\,\text{s}$), which is superior to not only ROO but also more sophisticated WUC scheme. This interesting result indicates that although the average network throughput is maximized (through decreasing blocking probabilities) by the deliberate wake-up control mechanism of WUC, the resulting scheme becomes less energy-efficient. In other words, while the network rejects less number of UE requests by further incorporating the sleeping SBSs, the overall network starts consuming more power since not all SBSs are allowed to complete their full sleep period. As a result, the energy efficiency of WUC deteriorates, and falls even below ROO under certain settings. We therefore conclude that, *in contrast to low utilization scenario, the energy efficiency of WUC can be poor under high network utilization, although the associated average throughput might still be the best.*

VI. Conclusion

In this study, we consider OOS strategies to have energy-efficient SCNs. In particular, we propose a novel load definition for the SCN traffic, and derived its approximate distribution rigorously. Two novel load based OOS algorithms (i.e., CLB and DLB) are also proposed together with two benchmark strategies ROO (i.e., simple baseline) and WUC (i.e., sophisticated). We show that CLB and DLB perform better than ROO, and have similar performance as compared to WUC under low traffic periods. Assuming relatively high network utilization, CLB and DLB turns out to be even more energy-efficient then WUC. We finally show that the performance of CLB can be efficiently attained by DLB in a distributed fashion relying on the statistics of the traffic load. As a future work, traffic load model can be extended to capture diverse mobile usage patterns by including the distributions of inter-arrival time, and file size distributions. Besides, wake-up control and load based schemes can be extended by considering mobile power consumption, and macrocell serving capacity in mid-traffic and high-traffic profiles. Moreover, optimal strategies to choose the best sleep time and energy-saving state can be developed based on load variable.

Appendix A
Proof of Theorem 1

In order to compute the average load factor conditioned on the number of SBS (i.e., $\mathbb{E}[w|n_c = i]$), we choose an arbitrary UE that is off the origin (i.e., the target SBS of interest) by a distance r with $r \le R_{\text{th}}$, as shown in Fig. 3(a). Because any user can only receive service from the SBSs separated by at most a distance of R_{th}, the SBSs that contribute into the load factor are those lying in the overlapping area $A_o(r)$ and the user exclusion area $A_e(r)$, as shown in Fig. 3(a). These areas can be expressed parametrically as follows

$$A_o(r) = 2r^2 - \theta + \frac{1}{2}\sin(2\theta), \tag{49}$$

$$A_e(r) = \pi R_{\text{th}}^2 - A_o(r), \tag{50}$$

where $\theta = \cos^{-1}\left(\frac{r}{2R_{\text{th}}}\right)$ is also depicted in Fig. 3(a).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking

The conditional load factor involved in (12) could be expressed as follows

$$\mathbb{E}[w|n_{\rm c}=i]=\int_0^{R_{\rm th}}\mathbb{E}[w|r,n_{\rm c}=i]f_r(r){\rm d}r, \tag{51}$$

where $f_r(r)=2r/R_{\rm th}$. The average load factor in (51), which is conditioned on the distance r and the number of SBSs i (i.e., located within a circle of radius $R_{\rm th}$ around the origin), can be expressed as a sum in the form of a binomial expansion as follows

$$\mathbb{E}[w|r,n_{\rm c}=i]=\sum_{k=0}^{i}\binom{i}{k}\mathbb{E}[w|r,n_{A_{\rm o}}(r)=k] \times p_{A_{\rm o}}(r)^k(1-p_{A_{\rm o}}(r))^{i-k}, \tag{52}$$

where $n_{A_{\rm o}}(r)$ stands for the number of SBSs in the overlapping area $A_{\rm o}(r)$, and $p_{A_{\rm o}}(r)$ is the probability of an SBS being in $A_{\rm o}(r)$. Since SBSs are distributed uniformly, we have $p_{A_{\rm o}}(r)=A_{\rm o}(r)/\pi R_{\rm th}^2$. In addition, each term in the summation of (52) considers a case in which k SBSs exist in the overlapping area $A_{\rm o}(r)$ out of a total of i SBSs off the origin by at most the distance $R_{\rm th}$.

While computing the average load expression at the right side of (52) by employing the definition given in (1), one should take into account k SBSs from the overlapping area $A_{\rm o}(r)$, v SBSs from the user exclusion area $A_{\rm e}(r)$, and the single SBS located at the origin as follows

$$\begin{aligned}\mathbb{E}[w|r,n_{A_{\rm o}(r)}=k]&=\sum_{v=0}^{\infty}\frac{1}{k+v+1}P\left\{n_{A_{\rm e}}(r)=v\right\}\\&=\sum_{v=0}^{\infty}\frac{[\nu_e(r)]^v\,e^{-\nu_e(r)}}{(k+v+1)\,v!},\end{aligned} \tag{53}$$

where $n_{A_{\rm e}}(r)$ is the random variable representing the number of SBSs in the user exclusion area $A_{\rm e}(r)$, which follows the Poisson distribution with rate $\nu_{\rm e}(r)=\rho_{\rm c}A_{\rm e}(r)=\nu_{\rm c}-\rho_{\rm c}A_{\rm o}(r)$. The average traffic load in (13) is readily obtained by employing (52) and (53) in (51). ■

Appendix B
Proof of Theorem 2

The expectation $E[w^2|n_{\rm c}=i]$ in (19) can be computed following the steps of (51)-(53) together with the modified version of (53) given as

$$\mathbb{E}[w^2|r,n_{A_{\rm o}}(r)=k]=\sum_{v=0}^{\infty}\frac{[\nu_{\rm e}(r)]^v\,e^{-\nu_{\rm e}(r)}}{(k+v+1)^2\,v!}. \tag{54}$$

The computation of the expectation $\mathbb{E}[w_kw_l|n_{\rm c}=i]$ in (20) is cumbersome due to the correlation between the individual load factors w_k and w_l. Because $\mathbb{E}[w_kw_l|n_{\rm c}=i]$ requires a second-degree analysis, we modify Fig. 3(a) by adding a second user, and obtain Fig. 3(b). This new coordinate system has a SBS located at the origin, as before, and two UEs off this SBS by random distances r_1 and r_2, both of which have the common distribution with $f_r(r)=2r/R_{\rm th}$.

We may have various orientations for relative positions of two UEs in Fig. 3(b), and therefore introduce a new variable ω which describes the difference of user angles with respect to the origin. Note that ω is actually the difference of two uniform random variables distributed between 0 and 2π, and, hence, with the distribution given by (21) [26]. The second-order expectation of interest could be accordingly written as

$$\begin{aligned}\mathbb{E}[w_kw_l|n_{\rm c}=i]=\int_0^{R_{\rm th}}\int_0^{R_{\rm th}}\int_{-2\pi}^{2\pi}E\left[w_kw_l|\boldsymbol{r},\omega,n_{\rm c}=i\right]f_r(r_1)\,f_r(r_2)\\\times g(\omega)\,{\rm d}\omega\,{\rm d}r_1\,{\rm d}r_2,\end{aligned} \tag{55}$$

which is counterpart of (51) in the first moment computation, and where $\boldsymbol{r}=[r_1\ r_2]$.

To compute the expectation at the right side of (55), we need to consider various geometric orientations of two UEs around the origin, as in Fig. 10. Among them, Case-I has a circular triangular overlapping area whereas Case-II and Case-III specify non-triangular overlapping areas. While the condition for the existence of a circular triangle area and respective area formulations are given in [27], the non-triangular areas should be computed by employing (49).

In order to express the term $E[w_kw_l|\boldsymbol{r},\omega,n_{\rm c}=i]$ in the form of multinomial expansion, we need to take into account the number of constituent areas (i.e., N) forming the circular area of radius $R_{\rm th}$ around the origin (i.e., where the SBS is located). Note that the expectation in (55) assumes $i+1$ SBSs in this circular region. Indeed, N is a function of the angle ω given in Fig. 3(b), and all 3 cases sketched in Fig. 10 occurs for a certain set of ω values [27]. Based on these 3 orientations in Fig. 10, Case-I and Case-II have $N=4$ constituent areas while Case-III has $N=3$. As a counterpart of (52), the desired expansion could therefore be given as

$$\begin{aligned}E[w_kw_l|\boldsymbol{r},\omega,n_{\rm c}=i]=\sum_{m_1=0}^{i}\cdots\sum_{m_{N-1}=0}^{i-\sum_{v=0}^{N-2}m_v}\\E[w_kw_l|\boldsymbol{r},\omega,\boldsymbol{n}(\boldsymbol{r},w)=\boldsymbol{m}]f(\boldsymbol{m};\boldsymbol{p}(\boldsymbol{r},w)),\end{aligned} \tag{56}$$

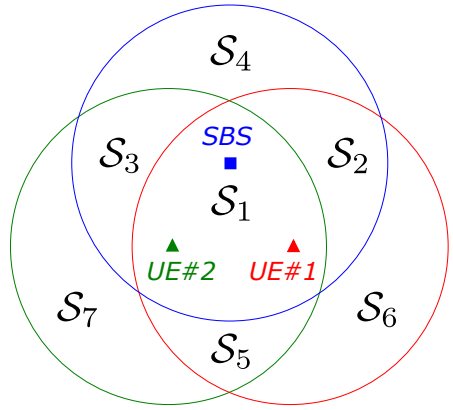
where $\boldsymbol{n}(\boldsymbol{r},w)$ is the vector of the number of SBSs in each of the constituent areas, $\boldsymbol{p}(\boldsymbol{r},w)$ is the vector of multinomial probabilities associated with each of these areas, and $\boldsymbol{m}$ is the vector of summation indices. Each term of the summation in (56) corresponds to a unique distribution of the total of i SBSs over the constituent areas. Specifically, the number of SBSs in the constituent area $A_v(\boldsymbol{r},w)$ is $n_v(\boldsymbol{r},w)=m_v$ for $v=1,2,\ldots,N$ with $\sum_{v=1}^{N}m_v=i$.

The probability mass function (PMF) in (56) is given as

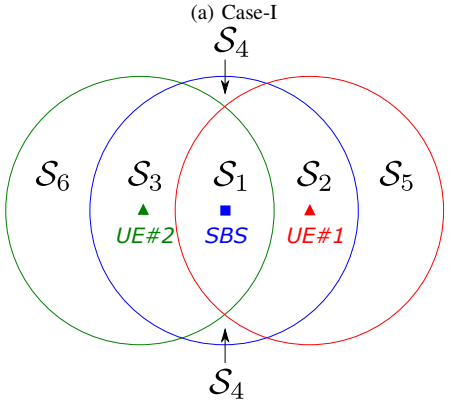
$$f\left(\boldsymbol{m};\boldsymbol{p}(\boldsymbol{r},w)\right)=i!\prod_{v=1}^{N}(m_v!)^{-1}\prod_{v=1}^{N}p_{A_v}(\boldsymbol{r},w)^{m_v}, \tag{57}$$

where $p_{A_v}(\boldsymbol{r},w)$ is the individual probability entry of $\boldsymbol{p}(\boldsymbol{r},w)$ associated with the constituent area $A_v(\boldsymbol{r},w)$, and is therefore given to be $p_{A_v}(\boldsymbol{r},w)=A_v(\boldsymbol{r},w)/\pi R_{\rm th}^2$ owing to the uniform distribution of SBSs in space. Note that m_v SBSs in $A_v(\boldsymbol{r},w)$ can be placed in $m_v!$ different ways, and this makes $\prod_{v=1}^{N}m_v!$ considering all constituent areas. Since the total of i SBSs can

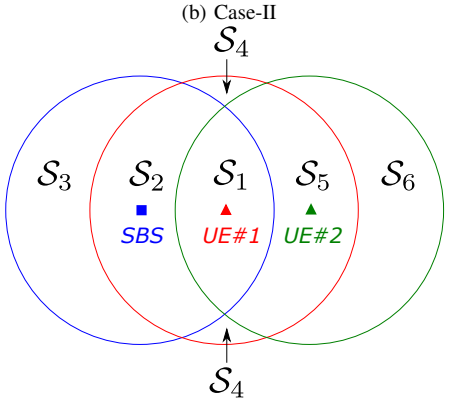
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGCN.2019.2931700, IEEE Transactions on Green Communications and Networking



(a) Case-I



(b) Case-II



(c) Case-III

Fig. 10. Relative orientations of two UEs around a single SBS.

be ordered in $i!$ different ways, $i!\prod_{v=1}^{N}(m_v!)^{-1}$ in (57) takes into account all possible relative SBS placements.

Following the philosophy behind (53), and employing the PMF in (57), the expectation in the summation of (56) can be computed as follows

$$E[w_k w_l|\boldsymbol{r},\omega,\boldsymbol{n}(\boldsymbol{r},w)] = \sum_{v_1=0}^{\infty}\sum_{v_2=0}^{\infty}\sum_{v_c=0}^{\infty} P\left\{n_{e,c}(\boldsymbol{r},w)=v_c\right\} \times \prod_{s=1}^{2}\frac{P\left\{n_{e,s}(\boldsymbol{r},w)=v_s\right\}}{n_{o,s}(\boldsymbol{r},w)+v_s+v_c+1}, \tag{58}$$

$$= \sum_{v_1=0}^{\infty}\sum_{v_2=0}^{\infty}\sum_{v_c=0}^{\infty}\frac{\left[\nu_{e,c}(r)\right]^{v_c}e^{-\nu_{e,c}(r)}}{v_c!} \times \prod_{s=1}^{2}\frac{\left[\nu_{e,s}(r)\right]^{v_s}e^{-\nu_{e,s}(r)}}{v_s!\left(n_{o,s}(\boldsymbol{r},w)+v_s+v_c+1\right)}, \tag{59}$$

where $n_{e,c}(\boldsymbol{r},w)$ and $n_{e,s}(\boldsymbol{r},w)$ are the number of SBSs in the common exclusion area $A_{e,c}(\boldsymbol{r},w)$ and distinct exclusion area $A_{e,v}(\boldsymbol{r},w)$ for the sth UE, respectively, which follow the Poisson distribution with rates $\nu_{e,c}(r)=\rho_c A_{e,c}(r)$ and $\nu_{e,s}(r)=\rho_c A_{e,s}(r)$, respectively, with $s=1,2$. We show all the exclusion and overlapping areas in Table IV for the orientations considered in Fig. 10.

TABLE IV
OVERLAPPING AND EXCLUSION AREAS

	Case-I	Case-II	Case-III
$A_{e,c}(\boldsymbol{r},w)$	$\mathcal{S}_5$	$\emptyset$	$\mathcal{S}_5$
$A_{o,1}(\boldsymbol{r},w)$	$\mathcal{S}_1\bigcup\mathcal{S}_2$	$\mathcal{S}_1\bigcup\mathcal{S}_2$	$\mathcal{S}_1\bigcup\mathcal{S}_2$
$A_{e,1}(\boldsymbol{r},w)$	$\mathcal{S}_6$	$\mathcal{S}_5$	$\mathcal{S}_4$
$A_{o,2}(\boldsymbol{r},w)$	$\mathcal{S}_1\bigcup\mathcal{S}_3$	$\mathcal{S}_1\bigcup\mathcal{S}_3$	$\mathcal{S}_1$
$A_{e,2}(\boldsymbol{r},w)$	$\mathcal{S}_7$	$\mathcal{S}_6$	$\mathcal{S}_6$

Note that $n_{o,s}(\boldsymbol{r},w)$ in (58) is a given (i.e., deterministic) value representing the number of SBSs in the overlapping area $A_{o,s}(\boldsymbol{r},w)$, with $s=1,2$. More specifically, $n_{o,s}(\boldsymbol{r},w)$ is the sum of the entries of $\boldsymbol{n}(\boldsymbol{r},w)$ associated with the constituent areas forming $A_{o,s}(\boldsymbol{r},w)$, which are explicitly given in Table IV for $s=1,2$. As an example, we have $n_{o,1}(\boldsymbol{r},w)=n_1(\boldsymbol{r},w)+n_2(\boldsymbol{r},w)$ and $n_{o,2}(\boldsymbol{r},w)=n_1(\boldsymbol{r},w)+n_3(\boldsymbol{r},w)$ for Case-I, where $n_i(\boldsymbol{r},w)$ is the number of SBSs in the area $\mathcal{S}_i$ for $i=1,2,3$.

As a particular case, since $A_{e,c}(\boldsymbol{r},w)$ does not exist for Case-II, (59) simplifies to

$$E[w_k w_l|\boldsymbol{r},\omega,\boldsymbol{n}(\boldsymbol{r},w)] = \sum_{v_1=0}^{\infty}\sum_{v_2=0}^{\infty}\prod_{s=1}^{2}\frac{\left[\nu_{e,s}(r)\right]^{v_s}e^{-\nu_{e,s}(r)}}{v_s!\left(n_{o,s}(\boldsymbol{r},w)+v_s+1\right)}. \tag{60}$$

Finally, incorporating (56), (57), and (59) into (55), we finally obtain (20).

REFERENCES

[1] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. T. Sukhavasi, C. Patel, and S. Geirhofer, "Network densification: the dominant theme for wireless evolution into 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 82–89, 2014.

[2] T. Q. Quek, G. de la Roche, İ. Güvenç, and M. Kountouris, *Small cell networks: Deployment, PHY techniques, and resource management*. Cambridge University Press, 2013.

[3] I. Hwang, B. Song, and S. S. Soliman, "A holistic view on hyper-dense heterogeneous and small cell networks," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 20–27, 2013.

[4] S. Samarakoon, M. Bennis, W. Saad, M. Debbah, and M. Latva-aho, "Ultra dense small cell networks: Turning density into energy efficiency," *IEEE J. Select. Areas Commun. (JSAC)*, vol. 34, no. 5, pp. 1267–1280, 2016.

[5] I. Siomina and D. Yuan, "Analysis of cell load coupling for lte network planning and optimization," *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 2287–2297, June 2012.

[6] A. J. Fehske and G. P. Fettweis, "Aggregation of variables in load models for interference-coupled cellular data networks," in *2012 IEEE International Conference on Communications (ICC)*, June 2012, pp. 5102–5107.

[7] H. Klessig, A. Fehske, G. Fettweis, and J. Voigt, "Cell load-aware energy saving management in self-organizing networks," in *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*, Sep. 2013, pp. 1–6.

[8] C. Li, J. Zhang, and K. B. Letaief, "Throughput and energy efficiency analysis of small cell networks with multi-antenna base stations," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2505–2517, 2014.

[9] Y. S. Soh, T. Q. Quek, M. Kountouris, and H. Shin, "Energy efficient heterogeneous cellular networks," *IEEE J. Select. Areas Commun. (JSAC)*, vol. 31, no. 5, pp. 840–850, 2013.

[10] A. Merwaday and I. Güvenç, "Optimisation of FeICIC for energy efficiency and spectrum efficiency in LTE-advanced HetNets," *IET Electronics Letters*, vol. 52, no. 11, pp. 982–984, 2016.

[11] C. Peng, S.-B. Lee, S. Lu, and H. Luo, "GreenBSN: Enabling energy-proportional cellular base station networks," *IEEE Trans. Mobile Computing*, vol. 13, no. 11, pp. 2537–2551, Nov 2014.

[12] Cisco, "Fog computing and the internet of things: Extend the cloud to where the things are," 2015. [Online]. Available: https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf

[13] Y. Zeng, K. Xiang, D. Li, and A. V. Vasilakos, "Directional routing and scheduling for green vehicular delay tolerant networks," *Wireless Networks*, vol. 19, no. 2, pp. 161–173, 2013.

[14] S. He, X. Li, J. Chen, P. Cheng, Y. Sun, and D. Simplot-Ryl, "EMD: energy-efficient P2P message dissemination in delay-tolerant wireless sensor and actor networks," *IEEE J. Select. Areas Commun. (JSAC)*, vol. 31, no. 9, pp. 75–84, 2013.

[15] Y. Cao and Z. Sun, "Routing in delay/disruption tolerant networks: A taxonomy, survey and challenges," *IEEE Commun. Surveys & Tutorials*, vol. 15, no. 2, pp. 654–677, 2013.

[16] X. Guo, Z. Niu, S. Zhou, and P. R. Kumar, "Delay-constrained energy-optimal base station sleeping control," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1073–1085, May 2016.

[17] H. Celebi and I. Guvenc, "Load analysis and sleep mode optimization for energy-efficient 5G small cell networks," in *Proc. IEEE Int. Conf. Commun. (ICC) Workshops*, May 2017, pp. 1159–1164.

[18] C. Liu, B. Natarajan, and H. Xia, "Small cell base station sleep strategies for energy efficiency," *IEEE Trans. Vehic. Technol.*, vol. 65, no. 3, pp. 1652–1661, Mar. 2016.

[19] I. Ashraf, F. Boccardi, and L. Ho, "SLEEP mode techniques for small cell deployments," *IEEE Commun. Mag.*, vol. 49, no. 8, pp. 72–79, Aug. 2011.

[20] W. Vereecken, I. Haratcherev, M. Deruyck, W. Joseph, M. Pickavet, L. Martens, and P. Demeester, "The effect of variable wake up time on the utilization of sleep modes in femtocell mobile access networks," in *Proc. Annual Conf. Wireless On-Demand Network Systems and Services (WONS)*, Jan. 2012, pp. 63–66.

[21] M. Tanemura, "Statistical distributions of poisson voronoi cells in two and three dimensions," *FORMA*, vol. 18, no. 4, pp. 221–247, 2003.

[22] J.-S. Ferenc and Z. Néda, "On the size distribution of poisson voronoi cells," *Physica A: Statistical Mechanics and its Applications*, vol. 385, no. 2, pp. 518–526, 2007.

[23] C. Davies, "Size distribution of atmospheric particles," *Journal of Aerosol Science*, vol. 5, no. 3, pp. 293–300, 1974.

[24] S. M. Ross, *Introduction to Probability Models*, 10th ed. Academic Press, 2009.

[25] A. Blogowski, O. Klopfenstein, and B. Renard, "Dimensioning X2 backhaul link in LTE networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2012, pp. 2768–2773.

[26] D. Milios, "Probability distributions as program variables," *Master's thesis, School of Informatics, University of Edinburgh*, 2009.

[27] M. Fewell, "Area of common overlap of three circles," Defence Science and Technology Organisation, Report DSTO-TN-0722, Oct. 2006.

Haluk Çelebi received the BS degree Electronics and Communications Engineering from Istanbul Technical University, Turkey, in 2005, and M.S. degree from Columbia University in 2011. He is currently a Ph.D. candidate at Columbia University. His research interests include design and performance analysis of energy-efficient cellular networking.

Yavuz Yapıcı received the BS and MS degrees in Electrical and Electronics Engineering from Bilkent University, Ankara, Turkey, in 2002 and 2005, respectively, and the Ph.D. degree in the same department of Middle East Technical University, Ankara, in 2011. He is currently a Research Associate with the Department of Electrical and Computer Engineering, North Carolina State University. His research interests are mainly on the next-generation wireless technologies with a particular emphasis on millimeter-wave and UAV communications.

İsmail Güvenç (SM'10) received the Ph.D. degree in electrical engineering from the University of South Florida, Tampa, FL, USA, in 2006. He was with Mitsubishi Electric Research Labs in 2005, DOCOMO Innovations from 2006 to 2012, and Florida International University from 2012 to 2016. Since 2016, he has been an Associate Professor with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh,NC, USA. His recent research interests include 5G wireless systems, communications and networking with drones, and heterogeneous wireless networks. He has authored over180 conference/journal papers and book chapters, and several standardization contributions. He has co-authored/co-edited three books for Cambridge University Press. He has invented/co-invented in some 30 U.S. patents. He was a recipient of the USF Outstanding Dissertation Award in 2006, the Ralph E. Powe Junior Faculty Enhancement Award in 2014, the NSF CAREER Award in 2015, and the FIU College of Engineering Faculty Research Award in 2016. He served as an Editor for the IEEE COMMUNICATIONS LETTERS from 2010 to 2015, the IEEE WIRELESS COMMUNICATIONS LETTERS from2011 to 2016, and as a guest editor for several other journals. He has been serving as an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS since 2016.

Henning Schulzrinne is Levi Professor of Computer Science at Columbia University. He received the Ph.D. degree from the University of Massachusetts in Amherst, Massachusetts. He was a member of the technical staff at AT&T Bell Laboratories and Associate Department Head at GMD-Fokus (Berlin), before joining the Computer Science and EE departments at Columbia University. He served as chair of Computer Science from 2004 to 2009 and as Engineering Fellow, Technical Advisor and Chief Technology Officer of the Federal Communications Commission (FCC) from 2010 until 2017. Protocol standards codeveloped by him, including RTP, RTSP and SIP, are now used by almost all Internet telephony and multimedia applications. He is a Fellow of the ACM and IEEE.