TAES: Traffic-Aware Energy-Saving Base Station Sleeping and Clustering in Cooperative Networks

Jihwan Kim†, Hyang-Won Lee‡ and Song Chong†
†Dept. of Electrical Engineering, KAIST, Daejeon, Korea, kimji@netsys.kaist.ac.kr and songchong@kaist.edu
‡Dept. of Internet & Multimedia Engineering, Konkuk University, Seoul, Korea, leehw@konkuk.ac.kr

***Abstract*—We consider energy efficient base station sleeping and clustering problems in cooperative cellular networks where clusters of base stations jointly transmit to users. Our key idea of energy saving is to exploit a spatio-temporal fluctuation of traffic demand, which is to use minimal energy to provide capacity only slightly greater than varying traffic demand. Then, energy saving is possible without capacity loss. However, it is highly challenging to design traffic-aware algorithms without the future traffic demand information. To overcome this, we develop algorithms using queue instead of the future traffic information. For BS clustering problem, we propose an optimal algorithm that has polynomial complexity. For BS sleeping problem, which is a complex combinatorial problem, we propose two algorithms; One finds an optimal solution with reduced complexity compared to the exhaustive search, and the other finds a near-optimal solution with polynomial complexity. Through extensive simulations we show that the proposed algorithms can save significant energy when traffic load is low.**

I. Introduction

The great popularity of smartphones, tablets, and laptops, which are wirelessly connected to Internet, has caused an exponential traffic increase in cellular networks [1]. In order to provide sufficient capacity, a large number of base stations have been deployed, which leads substantial energy consumption. Studies in [2], [3] show that base stations already use about 60-80% of total energy consumption in cellular networks, and thus, energy efficiency of base stations is becoming a vital design goal in cellular networks.

In order to reduce energy consumption in base stations, there have been extensive attempts to develop energy-efficient resource management schemes, e.g., transmit power control [4], CPU speed scaling control [5], BS sleeping [6]–[10] and so on. Especially, BS sleeping techniques, in which underutilized base stations are allowed to sleep and traffic load of the sleeping BSs are transferred to neighbor BSs, has great potential to save energy by reducing static power consumption of BSs. The reason is that BSs are typically deployed to provide higher capacity than peak traffic volume but they are actually underutilized most of the time [11], [12]. Evaluations using a real traffic trace show that turning on/off BSs can possibly save a tremendous amount of energy (up to 90%), under simple sleeping schemes. The key question is then when and which base stations should go to sleep.

The goals of existing schemes for energy-efficient BS sleeping are mostly classified into two categories; i) minimizing energy consumption while satisfying static minimum data rate or QoS requirements [6], [7], and ii) maximizing energy efficiency defined as data rate per power consumption [8], [10]. These works are much more energy efficient than schemes focusing only on capacity, but it is hard to say that they are adapting well to spatio-temporal fluctuation of traffic demand, which is a key factor to save energy in our thinking.

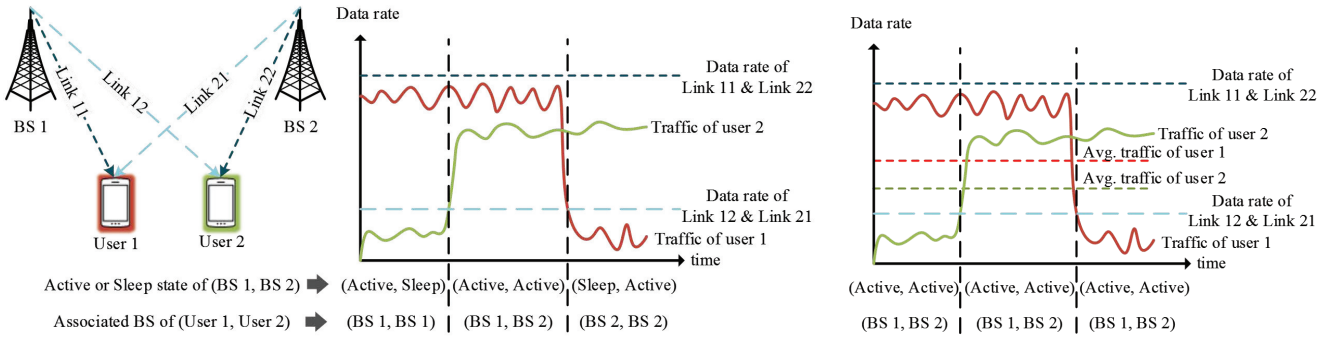
The concept of adapting to spatio-temporal fluctuation of traffic demand is as follows; If the network capacity is greater than traffic demand, energy for the remaining capacity would be wasted. Thus, by fitting capacity to the traffic demand, energy saving is possible without performance loss. However, since traffic demand is not static and varies over time and space, the capacity also has to change adaptively to the traffic demand. Fig. 1(a) shows a simple example of traffic adaptive BS sleeping and user association without energy waste, which uses only minimal BSs to support varying traffic demand.

If the future traffic dynamics are availabe, the algorithms [6], [7] minimizing energy consumption while guaranteeing minimum rate can ideally adapt to spatio-temporal traffic fluctuation, as shown in Fig. 1(a). However, it is a very strong assumption that the future traffic dynamics are known. Instead, the algorithms can be applied also with a relaxed assumption such that the average traffic demand is known. However, in the case of the average traffic demand, they cannot adapt to temporal fluctuation of traffic and waste energy consumption, as shown in Fig. 1(b). Thus, our goal is to develop a new traffic-aware algorithm without any additional information of traffic, which is challenging.

Another difficulty of BS sleeping problem is that it is tightly coupled to the problem determining which BSs should communicate with users. For example, in order to turn off an underutilized BS, existing users of the BS have to find new BSs to communicate with. In general, a user should communicate with only one BS (i.e., single-BS association), but now it is also possible that a user communicates with multiple BSs simultaneously, due to the advanced BS cooperation techniques such as coordinated multi-point (CoMP) [13] and distributed antenna systems (DAS) [14]. These techniques

This work was supported by the ICT R&D program of MSIP/IITP. [14-000-04-001, Development of 5G Mobile Communication Technologies for Hyper-connected smart services]. Jihwan Kim and Hyang-Won Lee were supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2012R1A1A1012610).

978-3-9018-8273-9/15/ ©2015 IFIP



(a) Traffic adaptation in ideal case

(b) Traffic adaptation with the average traffic demand; Energy is wasted since the temporal variation is ignored.

Fig. 1. Spatial and temporal traffic fluctuation and traffic adapting BS sleep mode and association controls; Each user can associate with only one BS, and an active BS provides static link capacity to the associated user.

can significantly mitigate the edge user effect (e.g., see [15]) that arises in single-BS association paradigms. The BS cooperation may require additional energy for signal processing and backhauling, but they can significantly improve spectral efficiency. Thus, enabling the BS cooperation has the potential for improving energy efficiency [8], [10], and we also consider scenarios in which a user can communicate with multiple BSs simultaneously.

Then, the problem is to make decisions jointly on *BS sleeping* and *BS clustering*, aiming to save energy via adapting to spatio-temporal traffic fluctuation which is unknown beforehand. Here, BS clustering is to form sets of BSs that will jointly transmit to users. We consider the user-centric clustering [16], [17] allowing that each user individually forms a BS cluster, which is the most general setting.

The main contributions are summarized as follows:

1) First, we develop BS sleeping and clustering algorithms to save energy via adapting to spatial and temporal traffic fluctuation. To be aware of traffic variations without the future traffic demand information, we use queue dynamics. For example, if the network capacity is excessive compared to traffic demand, then queues decrease and energy saving is possible by turning off BSs. Reversely, if the network capacity lacks, then queues increase and more BSs are activated to provide sufficient capacity. That way, our algorithms can save energy without capacity loss.
2) Next, we analyze the optimality and complexity of proposed algorithms. We show that the clustering algorithm finds an optimal solution for given BS sleep state with polynomial computation complexity (Section IV-A). For the sleeping problem, we define a *sleeping weight* representing whether or not to turn off a BS and propose two BS sleeping algorithms using the sleeping weight. One finds an optimal solution with reduced complexity compared to the exhaustive search (Section IV-B). The other finds a near-optimal solution using a greedy manner which requires only polynomial complexity. We also provide the performance gap between the greedy algorithm and the optimal algorithm (Section IV-C).
3) Finally, we verify the performance of our algorithms through extensive simulations. As expected, our algorithms achieve optimality or near-optimality with drastically reduced computing time compared to the exhaustive search (Section V-B). Also, our algorithms can adapt unknown traffic demand without the loss of capacity and save energy up to 80% when traffic demand is low (Section V-C).

II. System Model

We consider a cooperative wireless network consisting of a set $\mathcal{S}$ of disjoint cell sites. Each cell site $s \in \mathcal{S}$ has a central coordinator (or a processing unit), a set $\mathcal{B}_s$ of base stations (or radio units) and a set $\mathcal{K}_s$ of users. Each base station and user are included in only one cell site. We denote the whole BS set by $\mathcal{B} = \cup_{s\in\mathcal{S}}\mathcal{B}_s$, the whole user set by $\mathcal{K} = \cup_{s\in\mathcal{S}}\mathcal{K}_s$, and the cell site of user k by $s(k)$. Base stations have a sleep mode to reduce energy consumption and a joint transmission function by cooperating with other BSs in the same cell site. We assume a slotted system, and the time slot index is denoted by t. At the start of each time slot, we decide which BSs go to sleep mode and how to form a cluster for each user, and during the time slot, actual transmissions occur under the given BS sleep mode and clustering state. We assume that the length of a time slot is sufficiently large compared to a time scale of user scheduling.

A. Achievable Data Rate

The achievable data rate during each time slot depends not only on BS sleep mode and BS clustering but also on user scheduling and precoder design for the joint transmission. The underlying scheduling and precoding policy is assumed as follows. In each cell site, only one user can receive data at a time, and users share time resource independently of BS sleep mode and clustering (e.g., round-robin scheduling). As a precoding policy for the joint transmission, the maximal ratio combining (MRC) method is used, which maximizes the desired signal strength in a multi-antenna system without considering interference [18]. This assumption enables to express the achievable data rate as a function of sleep mode and clustering decision in closed form and helps identify properties of optimal solutions. There may exist other scheduling and precoding policies that perform better with our BS sleeping

and clustering, however we leave this issue as future study. Also, we assume the downlink case since we consider an energy consumption of BSs.

We denote the sleep mode indicator of BS b by δ_b, which is 1 if active mode and 0 if sleep mode. We assume that each user individually forms a cluster of BSs in the same cell site for the joint transmission, and each BS adjusts its transmit power when it operates as a cluster of user k. Denote by p_{bk} the transmit power of BS b to user k. Then, p_{bk} has a constraint given by

$$0 \leq p_{bk} \leq P_b \delta_b, \quad \forall b \in \mathcal{B}_s, k \in \mathcal{K}_s, s \in \mathcal{S}, \tag{1}$$

where P_b is a transmit power budget of BS b. In this paper, clustering is to determine the values of p_{bk} for $\forall k \in \mathcal{K}$ and $\forall s \in \mathcal{S}$.

Under the basic scheduling and precoding policy, the achievable rate of user k is given by $\theta_k \gamma_k$ where θ_k is a time fraction allocated to user k during the one time slot and γ_k is an instantaneous rate of user k. Then, θ_k must satisfy $\sum_{k \in \mathcal{K}_s} \theta_k \leq 1, \forall s \in \mathcal{S}$, and $\theta_k = 1/|\mathcal{K}_s|$ in round-robin case. Using Shannon capacity, the instantaneous rate of user k for given sleep mode δ and clustering p can be approximated as

$$\gamma_k(\delta, p) = B \log_2 \left(1 + \frac{\sum_{b \in \mathcal{B}_{s(k)}} G_{bk} p_{bk}}{I_k(\delta, p) + N_k} \right) \tag{2}$$

where B is the system bandwidth, G_{bk} is the channel condition between BS b and user k, N_k is the noise power of user k and $I_k(\delta, p)$ is the interference from other cell sites. For analytical tractability, we approximate the instantaneous rate by the value derived under the worst case interference, i.e., $I_k(\delta, p) = I_{k,worst} = \sum_{b \in \mathcal{B} \setminus \mathcal{B}_{s(k)}} G_{bk} P_b$ in (2). Under this approximation, the data rate of user k depends only on decisions in its cell site $s(k)$. From now on we focus on a single cell site.

B. Queueing Structure

Denote by $A_k(t)$ the traffic arrival rate of user k at time slot t. We assume that $A_k(t)$ is an i.i.d. process and there is an upper bound of arrival rates during one time slot such that $A_k(t) \leq A_{max}, \forall k \in \mathcal{K}$ and $\forall t$.

Arrival traffic is queued in the central coordinator and transmitted to each user k with rate $\gamma_k(\delta(t), p(t))\theta_k$ at each time slot t. Then, queue dynamics of user k at time slot t is expressed as

$$q_k(t+1) = [q_k(t) - \gamma_k(\delta(t), p(t))\theta_k]^+ + A_k(t), \tag{3}$$

where $[\cdot]^+$ denotes the projection onto the set of non-negative real numbers. We call $\{A_k(t)\}_{k \in \mathcal{K}, t}$ feasible arrival if there exists at least one policy to stabilize all queues.

C. Energy Consumption

We adopt a power consumption model of BS in [19]. Although a base station is comprised of various components consuming energy such as power amplifier, radio frequency circuit, processing unit, cooling system, and power supply, power consumption can be approximated by two types, static power consumption and dynamic power consumption proportional to transmit power. Then, the power consumption of BS b for given δ and p is expressed as

$$P_b^{BS}(\delta, p) = P_b^{cc} \delta_b + \sum_{k \in \mathcal{K}} P_b^{tx} p_{bk} \theta_k, \tag{4}$$

where P_b^{cc} is a static power including the power consumption of RF circuit, processing, cooling and power supply, and P_b^{tx} is a scaling factor of transmit power reflecting the impact of power amplifier, cooling and power supply. This simple model can capture the impact of sleeping and clustering decisions on power consumption.

III. Base Station Sleeping and Clustering Problem

Our goal is to minimize energy consumption while supporting traffic demand for every user. The challenge here is that the traffic demands are unknown. We apply Lyapunov optimization techniques to acheive this issue.

Let $\bar{P}$ be the average power consumption in networks and $\bar{d}_k$ be the average data rate of user k. They are determined by BS sleeping and clustering decisions at every time slot, such that $\bar{P} = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{b \in \mathcal{B}} E\left[P_b^{BS}(\delta(t), p(t))\right]$ and $\bar{d}_k = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} E\left[\gamma_k(\delta(t), p(t))\theta_k\right]$ (if the limits exist). Let $\bar{a}_k$ be the average arrival rate of user k, i.e., $\bar{a}_k = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} E\left[A_k(t)\right]$. Then, the problem can be expressed as

$$\min \bar{P} \ \text{ s.t. } \bar{a}_k \leq \bar{d}_k, \qquad \forall k \in \mathcal{K}. \tag{5}$$

Note that we do not use the value of $\bar{a}_k$ for BS sleeping and clustering decisions. Instead, we use the queue dynamics evolving as (3). If queue stability is guaranteed, i.e., $\lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{t=T-1} E\left[q_k(t)\right] < \infty, \forall k \in \mathcal{K}$, the constraint in (5) is also guaranteed. First, we formulate the BS sleeping and clustering problem at time t as follows,

[P-SC] BS Sleeping and Clustering Problem

$$\max_{\delta(t), p(t)} \sum_{k \in \mathcal{K}} q_k(t) \gamma_k(\delta(t), p(t)) \theta_k - V \sum_{b \in \mathcal{B}} P_b^{BS}(\delta(t), p(t))$$

$$\text{s.t. } 0 \leq p_{bk}(t) \leq P_b \delta_b(t), \quad \forall b \in \mathcal{B}, \forall k \in \mathcal{K}, \tag{6}$$

$$\delta_b(t) \in \{0, 1\}, \quad \forall b \in \mathcal{B}, \tag{7}$$

where V is a constant. Solving **[P-SC]** at every time slot is same to solve problem (5) by Lyapunov theorem [20], [21]. Details are in our technical report [22]. Now we focus on problem **[P-SC]** for an arbitrary time slot and omit time index t for simplicity.

In the objective function of **[P-SC]**, there are two terms, sum of queue-weighted data rates aiming to stabilize queue length and sum of BS power consumptions aiming to minimize the power. Since high energy consumption is required to obtain high data rate, the two terms conflict. The value of V determines a compromise between the two objectives.

The optimal sleeping and clustering decision of **[P-SC]** saves energy adaptively to spatio-temporal fluctuation of traffic arrivals. For instance, if there are less traffic arrivals, so queue

lengths are small, then saving energy becomes more important, and thus base stations go to sleep and BS cluster sizes are reduced. Reversely, when high traffic, base stations wake up and BS cluster sizes are increased to enhance data rates. These operations can reflect a temporal variation of traffic arrivals. Also, a queue length of each user is weighted to each user's data rate. Then, base stations near the users who have large queue length more aggressively use energy to increase the users' data rate. Thus, if we design BS sleeping and clustering policy solving **[P-SC]**, then the policy can save the energy consumption by exploiting both temporal and spatial variation of traffic arrivals.

IV. Traffic-Aware Energy-Saving Algorithm

We first analyze the properties of the optimal solution in problem **[P-SC]** for given BS sleep mode δ and propose i) a clustering algorithm finding an optimal solution in polynomial time. Next, we define *sleeping weight*, representing whether or not to turn off a BS, and analyze its properties. Inspired by the properties, we propose two BS sleeping algorithms, which are jointly executed with the clustering algorithm; One is ii) an optimal sleeping algorithm with reduced complexity, and the other is iii) a greedy sleeping algorithm finding a near-optimal solution in polynomial time with provable optimality gap.

A. Base Station Clustering Algorithm

First, given a BS sleep mode δ, the problem **[P-SC]** can be formulated as

[P-CL] BS Clustering Problem

$$\max_{p \text{ s.t. } (6)} \sum_{k\in\mathcal{K}} \left(q_k \gamma_k(\delta,p) - V \sum_{b\in\mathcal{B}} P_b^{tx} p_{bk} \right) \theta_k.$$

This problem is a convex optimization problem for p, and we can derive a sufficient and necessary condition for optimality based on the Karush-Kuhn-Tucker (KKT) condition [23], as follows

1) $\frac{B}{\ln 2}\frac{q_k G_{bk}}{T_k(\delta,p)} - V P_b^{tx} - \bar{\lambda}_{bk} + \underline{\lambda}_{bk} = 0, \quad \forall b \in \mathcal{B}, \forall k \in \mathcal{K}$
2) $\bar{\lambda}_{bk}(\delta_b P_b - p_{bk}) = 0, \quad \underline{\lambda}_{bk} p_{bk} = 0, \quad \forall b \in \mathcal{B}, \forall k \in \mathcal{K}$
3) $0 \le p_{bk} \le P_b \delta_b, \quad \bar{\lambda}_{bk}, \underline{\lambda}_{bk} \ge 0, \quad \forall b \in \mathcal{B}, \forall k \in \mathcal{K}$

where $T_k(\delta,p) = \sum_{b\in\mathcal{B}} G_{bk} p_{bk} + I_{k,worst} + N_k$, and $\bar{\lambda}$ and $\underline{\lambda}$ are the Lagrangian multipliers for constraint (6). Now we develop a clustering algorithm to find p satisfying the above KKT conditions in polynomial time.

Let us define the *clustering weight* of BS b for user k for given δ and p as follows:

$$w_{bk}^{CL}(\delta,p) = \frac{B}{\ln(2)} \frac{q_k G_{bk}}{T_k(\delta,p)} - V P_b^{tx}. \tag{8}$$

Using the KKT conditions, it can be shown that the clustering weights have the following properties only in optimal clustering p^*.

1) $w_{bk}^{CL}(\delta,p^*) > 0 \rightarrow p_{bk}^* = P_b \delta_b$
2) $w_{bk}^{CL}(\delta,p^*) = 0 \rightarrow 0 \le p_{bk}^* \le P_b \delta_b$
3) $w_{bk}^{CL}(\delta,p^*) < 0 \rightarrow p_{bk}^* = 0$

That is, if a clustering p satisfies these properties, then p is an optimal clustering. Consequently, we need to find a clustering p satisfying the above conditions. First, we can obtain the following result from the definition of the clustering weight.

Lemma 1: Suppose any δ and p satisfying constraints (6) and (7) and two BSs b_1 and b_2 such that $G_{b_1k}/P_{b_1}^{tx} \ge G_{b_2k}/P_{b_2}^{tx}$ for user k. Then, the clustering weights $w_{b_1k}^{CL}(\delta,p)$ and $w_{b_2k}^{CL}(\delta,p)$ satisfy the followings;

$$\text{If } w_{b_2k}^{CL}(\delta,p) \ge 0, \text{ then } w_{b_1k}^{CL}(\delta,p) \ge 0.$$
$$\text{If } w_{b_1k}^{CL}(\delta,p) \le 0, \text{ then } w_{b_2k}^{CL}(\delta,p) \le 0.$$

Proof: See our technical report [22]. ■

The lemma holds also for the optimal clustering case. This means that the order of G_{bk}/P_b^{tx} determines whether clustering weights are positive or negative in optimal clustering. Furthermore, the order is independent to clustering p.

Based on Lemma 1 and the optimal clustering properties, we develop Traffic-Aware Energy-Saving (TAES) clustering algorithm, as shown in Algorithm 1. For user k, BS b has a high priority for clustering when it provides large signal strength enhancement per unit power consumption, i.e., large G_{bk}/P_b^{tx}. This metric can be interpreted as an energy efficiency. Thus, base stations are selected as a cluster for user k in the order of G_{bk}/P_b^{tx}, until one of the last two end conditions is satisfied, i.e., there is no more base station or a clustering weight becomes negative (lines 6 or 12 in Algorithm 1). Then, we can prove the following result.

Theorem 1: For given sleep mode δ, TAES clustering algorithm finds the optimal solution of **[P-CL]**.

Proof: See our technical report [22]. ■

Algorithm 1 TAES Clustering Algorithm

```
1:  procedure TAES_Clustering_Algorithm(δ)
2:      for each user k ∈ K do
3:          Initialize p_bk = 0, ∀b ∈ B
4:          Set C = B                          ▷ C: candidate BS set
5:          Set T_k = I_{k,worst} + N_k
6:          while C ≠ ∅ do
7:              Set b* = arg max_{b∈C} G_bk / P_b^tx
8:              Set C = C \ {b*}
9:              if (B/ln(2)) (q_k G_{b*k} / (V P_{b*}^tx)) ≥ T_k then
10:                 if (B/ln(2)) (q_k G_{b*k} / (V P_{b*}^tx)) ≥ T_k + P_{b*} G_{b*k} δ_{b*} then
11:                     Set p_{b*k} = P_{b*} δ_{b*}
12:                 else
13:                     Set p_{b*k} = (B/ln(2)) (q_k / (V P_{b*k}^tx)) − T_k / G_{b*k}
14:                     Break
15:                 end if
16:                 Set T_k = T_k + p_{b*k} G_{b*k}
17:             end if
18:         end while
19:     end for
        Solution: p = {p_bk}_{b∈B,k∈K}
20: end procedure
```

B. Base Station Sleeping Algorithm: Optimal approach

Now we consider the BS sleeping problem in a situation that optimal clustering is possible. Denote by $f(\delta, p)$ the objective function of **[P-SC]**, i.e, $f(\delta,p) = \sum_{k\in\mathcal{K}} q_k\gamma_k(\delta,p)\theta_k - V\sum_{b\in\mathcal{B}} P_b^{BS}(\delta,p)$. Let us define the *sleeping weight* of BS b as

$$w_b^{SL}(\delta) = \sum_{k\in\mathcal{K}} \left[w_{bk}^{CL}(\delta, p^*(\delta))\right]^+ P_b\theta_k - VP_b^{cc} \qquad (9)$$

where $p^*(\delta)$ is an optimal clustering for given δ. Let $\mathcal{B}^A$ be an active BS set and δ be the sleep mode vector corresponding to $\mathcal{B}^A$, i.e., $\delta_b = 1$ if $b \in \mathcal{B}^A$ and $\delta_b = 0$ otherwise. For both active BS set and sleep mode vector, we use the same notation of objective function and sleeping weight, as $f(\mathcal{B}^A) = f(\delta, p^*(\delta))$ and $w_b^{SL}(\mathcal{B}^A) = w_b^{SL}(\delta), \forall b \in \mathcal{B}$. Then, the sleeping weight has following two properties.

***Lemma** 2:* For any active BS set $\mathcal{B}^A \subset \mathcal{B}$, if $\tilde{b} \in \mathcal{B}\backslash\mathcal{B}^A$ and $w_b^{SL}(\mathcal{B}^A \cup \{\tilde{b}\}) > 0$, then $f(\mathcal{B}^A) < f(\mathcal{B}^A \cup \{\tilde{b}\})$. Similarly, if $\tilde{b} \in \mathcal{B}^A$ and $w_b^{SL}(\mathcal{B}^A\backslash\{\tilde{b}\}) < 0$, then $f(\mathcal{B}^A) < f(\mathcal{B}^A\backslash\{\tilde{b}\})$.

Proof: See our technical report [22]. ■

***Lemma** 3:* The sleeping weight is a monotonic decreasing function of δ. That is, if $\tilde{\delta}_b \geq \delta'_b, \forall b \in \mathcal{B}$, then $w_b^{SL}(\tilde{\delta}) \leq w_b^{SL}(\delta'), \forall b \in \mathcal{B}$.

Proof: See our technical report [22]. ■

From Lemma 2 and Lemma 3, we can get an idea to reduce the search space. Suppose active BS set $\tilde{\mathcal{B}}^A$ and BS $\tilde{b}$ such that $\tilde{b} \notin \tilde{\mathcal{B}}^A$ and $w_{\tilde{b}}^{SL}(\tilde{\mathcal{B}}^A \cup \{\tilde{b}\}) > 0$. By Lemma 2, the objective function increases when BS $\tilde{b}$ is activated, and it still holds for any active BS set $\mathcal{B}^A$ such that $\mathcal{B}^A \subseteq \tilde{\mathcal{B}}^A$ by Lemma 3. Conversely, if the objective function increases when BS $\bar{b} \in \tilde{\mathcal{B}}^A$ goes to sleep for given $\tilde{\mathcal{B}}^A$, then it also increases for active BS set $\mathcal{B}^A$ such that $\mathcal{B}^A \supseteq \tilde{\mathcal{B}}^A$. Then, starting from the smallest BS active set (or the largest BS active set), we can iteratively classify BSs into three groups; One is a group of BSs that have to be activated, another is a group of BSs that have to be deactivated, and the other is a group of remaining BSs. Then, we only need to determine sleep modes for the third group, and thus, the search space can be reduced. The detailed procedure is represented in Algorithm 2.

It is meaningful to mention that although Algorithm 2 describes only the BS sleep mode control the clustering algorithm in Algorithm 1 is repeatedly executed to calculate sleeping weights w_b^{SL} which is a function of the optimal clustering for a given sleep mode. Thus, it can be said that the optimal sleeping algorithm finds both BS sleeping and clustering solutions.

The optimal sleeping algorithm still requires the exhaustive search over undecided BSs in phase II, but its search space can be significantly reduced by procedures in phase I. We show how much the search space is reduced via simulations in Section V. In spite of the reduced search space, the algorithm finds an optimal active BS set.

***Theorem** 2:* TAES optimal algorithm in Algorithm 2 finds the optimal solution of **[P-SC]**.

Proof: See [22]. ■

Algorithm 2 TAES Optimal Algorithm

procedure TAES_OPTIMAL_ALGORITHM
Phase I:
  Set $\mathcal{B}^A = \emptyset, \mathcal{B}^S = \emptyset, isEnd = 0$
    ▷ $\mathcal{B}^A$: active BS set, $\mathcal{B}^S$: sleep BS set
  while $isEnd = 0$ **do**
    Set $isEnd = 1$
    for each BS $b \in \mathcal{B}\backslash(\mathcal{B}^A \cup \mathcal{B}^S)$ **do**
      if $w_b^{SL}(\mathcal{B}^A) < 0$ **then**
        Set $\mathcal{B}^S = \mathcal{B}^S \cup \{b\}$, $isEnd = 0$
      end if
    end for
    for each BS $b \in \mathcal{B}\backslash(\mathcal{B}^A \cup \mathcal{B}^S)$ **do**
      if $w_b^{SL}(\mathcal{B}\backslash\mathcal{B}^S) > 0$ **then**
        Set $\mathcal{B}^A = \mathcal{B}^A \cup \{b\}$, $isEnd = 0$
      end if
    end for
  end while
Phase II:
  Set $\mathcal{Y}^* = \arg\max_{\mathcal{B}^A\subseteq\mathcal{Y}\subseteq\mathcal{B}\backslash\mathcal{B}^S} f(\mathcal{Y})$
  Solution: sleep BS set = $\mathcal{B}\backslash\mathcal{Y}^*$, active BS set = $\mathcal{Y}^*$
end procedure

C. Base Station Sleeping Algorithm: Greedy approach

Although the proposed TAES optimal algorithm finds an optimal solution, in the worst case where $\mathcal{B}^A$ and $\mathcal{B}^S$ in phase I of Algorithm 2 are empty sets, the complexity of the algorithm may be as high as that of the exhaustive search. Thus, we now propose a greedy BS sleeping algorithm.

Our greedy sleeping algorithm is described in Algorithm 3. The key idea is to sequentially turn off the BS which increases the objective value the most. In phase I, the algorithm sequentially turn off BSs starting from the state that all BSs are activated. Note that TAES optimal algorithm finds BSs to turn off starting from the state that all BSs are deactivated. Due to this difference, there may exist a BS, which increases the objective value when it becomes active, among BSs, which are already determined to sleep. Thus, in phase II, the algorithm finds a set of BSs (denote by $\mathcal{Z}$), which can possibly increase the objective value, among BSs, which are determined to sleep in phase I. Here, the nonempty set $\mathcal{Z}$ implies non-zero gap between greedy and optimal solutions. In phase III, the algorithm sequentially activates a BS among BSs in $\mathcal{Z}$, which most significantly increases the objective value. Then, we can derive the optimality gap as follows.

***Theorem** 3:* Let $\mathcal{B}^*$ be an optimal active BS set and $\tilde{\mathcal{B}}^A$ be an active BS set of TAES greedy algorithm. Then, a solution of TAES greedy algorithm has a following performance bound.

$$f(\tilde{\mathcal{B}}^A) \geq f(\mathcal{B}^*) - \sum_{b\in\mathcal{Z}} w_b^{SL}(\tilde{\mathcal{B}}^A\backslash\mathcal{Z})$$

where $\mathcal{Z}$ is a BS set generated in phase II of TAES greedy algorithm.

Proof: See our technical report [22]. ■

***Corollary** 1:* If $\mathcal{Z}$ is an empty set after phase II of TAES greedy algorithm, then the greedy solution is optimal.

Algorithm 3 TAES Greedy Algorithm

procedure TAES_GREEDY_ALGORITHM
Phase I:
  Set $\mathcal{C} = \mathcal{B}$ ▷ $\mathcal{C}$: candidate BS set for activating
  Set $\mathcal{B}^A = \mathcal{B}$
  while $\mathcal{C} \neq \emptyset$ **do**
    Set $b^* = \arg\min_{b \in \mathcal{C}} w_b^{SL}(\mathcal{B}^A \backslash \{b\})$
    Set $\mathcal{C} = \mathcal{C} \backslash \{b^*\}$
    if $w_{b^*}^{SL}(\mathcal{B}^A) < 0$ **then**
      Set $\mathcal{B}^A = \mathcal{B}^A \backslash \{b^*\}$
    end if
  end while
Phase II:
  Set $\mathcal{Z} = \emptyset$ ▷ $\mathcal{Z}$: BS set for checking
  for each BS $b \in \mathcal{B} \backslash \mathcal{B}^A$ **do**
    if $w_b^{SL}(\mathcal{B}^A) \geq 0$ **then**
      Set $\mathcal{Z} = \mathcal{Z} \cup \{b\}$
    end if
  end for
Phase III:
  Set $\mathcal{C} = \mathcal{Z}$ ▷ $\mathcal{C}$: candidate BS set for activating
  while $\mathcal{C} \neq \emptyset$ **do**
    Set $b^* = \arg\max_{b \in \mathcal{C}} f(\mathcal{B}^A \cup \{b^*\})$
    if $f(\mathcal{B}^A \cup \{b^*\}) > f(\mathcal{B}^A)$ **then**
      Set $\mathcal{B}^A = \mathcal{B}^A \cup \{b^*\}$, $\mathcal{C} = \mathcal{C} \backslash \{b^*\}$
    else
      Break
    end if
  end while
  Solution: sleep BS set =$\mathcal{B} \backslash \mathcal{B}^A$, active BS set = $\mathcal{B}^A$
end procedure

D. Complexity of TAES Algorithms

We now analyze the computational complexity of proposed TAES algorithms. First, it is easy to check that TAES clustering algorithm has $\mathcal{O}(|\mathcal{K}||\mathcal{B}|^2)$ complexity. TAES clustering algorithm is repeatedly used when calculating the sleeping weight or objective value for a given BS sleep state. Since the exhaustive search method calculates objective values for all possible BS sleep states, its compleixty is $\mathcal{O}\left(|\mathcal{K}||\mathcal{B}|^2 2^{|\mathcal{B}|}\right)$.

On the other hand, TAES optimal algorithm iterates at most $|\mathcal{B}|^2$ in phase I and each iteration requires $\mathcal{O}(|\mathcal{K}||\mathcal{B}|^2)$ complexity for sleeping weight comparison. After phase I, the number of BSs requiring to check is reduced to $|\mathcal{B}| - |\mathcal{B}^A| - |\mathcal{B}^S|$, and thus, the complexity is $\mathcal{O}\left(|\mathcal{K}||\mathcal{B}|^2\left(|\mathcal{B}|^2 + 2^{(|\mathcal{B}|-|\mathcal{B}^A|-|\mathcal{B}^S|)}\right)\right)$. Actually, the complexity of TAES optimal algorithm is equivalent to exhaustive search in the worst case, i.e., when $\mathcal{B}^A$ and $\mathcal{B}^S$ are empty sets. The effect of search space reduction is shown in Section V-B.

Finally, TAES greedy algorithm does not search BS states, and instead, it compares sleeping weights or objective values at most $|\mathcal{B}|$ times in phases I and III. Since the sleeping weights and objective values are computed at most $|\mathcal{B}|$ times for one comparison, the compleixty is $\mathcal{O}\left(|\mathcal{K}||\mathcal{B}|^4\right)$. In Table I, we summarize the complexity discussed above.

TABLE I
COMPUTATIONAL COMPLEXITY OF ALGORITHMS

Algorithm	Complexity
TAES Clustering	$\mathcal{O}\left(\|\mathcal{K}\|\|\mathcal{B}\|^2\right)$
TAES Optimal	$\mathcal{O}\left(\|\mathcal{K}\|\|\mathcal{B}\|^2\left(\|\mathcal{B}\|^2 + 2^{(\|\mathcal{B}\|-\|\mathcal{B}^A\|-\|\mathcal{B}^S\|)}\right)\right)$
TAES Greedy	$\mathcal{O}\left(\|\mathcal{K}\|\|\mathcal{B}\|^4\right)$
Exhaustive search	$\mathcal{O}\left(\|\mathcal{K}\|\|\mathcal{B}\|^2 2^{\|\mathcal{B}\|}\right)$

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed TAES algorithms in various scenarios. First, we verify that TAES optimal/greedy algorithms can achieve optimal/near-optimal value of the one-slot problem **[P-SC]** with very low complexity compared to the exhaustive search. Next, we demonstrate that TAES algorithms save power consumption without capacity loss by adapting to traffic variation.

A. Simulation Setup

We use a sample network topology composed of 10 BSs and 20 users in 0.5×0.5 km^2, in which BSs and users are randomly located according to the uniform distribution. For traffic generation, we assume that traffic arrivals follow a Poisson process with arrival rate λ and each arrival file follows exponential distribution with average size μ. We use homogeneous arrival rate λ for all users and change average file size of each user k, μ_k, in order to control traffic load. The path-loss model is given by $128.1 + 37.6\log_{10}(d)$ where d is the distance from BS to user in km. As one of performance metrics, we measure per-flow delay using Little's law. Other simulation parameters are summarized in Table II.

TABLE II
SIMULATION PARAMETER

Parameter	Value
Transmit power budget, P_b	0.2 W
Power consumption model [19]	
Transmit power, P_b^{tx}	4
Circuit Power, P_b^{cc}	2 W
Queue update time length	1 msec
Control time length	10 msec
Simulation time	10 sec
Mean inter-arrival time, $1/\lambda$	20 msec
System bandwidth	10 MHz
Background noise	-169 dBm/Hz

The following algorithms are compared with TAES algorithms.

- Throughput optimal policy (Thr. Opt.): In order to maximize capacity, all BSs are active and each user forms a BS cluster with a maximum size.
- Static policy (Static-M): Each user forms a static BS cluster with size M and each BS goes to sleep when there is no traffic demand from users who select the BS as one of cluster BSs.
- Traffic-aware algorithm (Power Min.): Assuming that we know the average traffic demand of each user k,

denoted by C_k, this policy finds a solution of a power minimization problem with minimum data rate guarantee constraints such that

$$\min_{\delta,p \text{ s.t.}(6),(7)} \sum_{b\in\mathcal{B}} P_b^{BS}(\delta,p) \text{ s.t. } \gamma_k(\delta,p)\theta_k \geq C_k, \forall k \in \mathcal{K}.$$

Due to the complexity problem, we use a greedy method to determine δ.

- Exhaustive search: This policy finds an optimal solution of **[P-SC]** by an exhaustive search for δ.
- Only clustering: BS clustering is determined by TAES clustering algorithm and BS sleeping is determined by on-demand manner same to static-M policy.
- Only sleeping: BS sleeping is determined by TAES greedy algorithm while each user forms a BS cluster with a maximum size.

B. Verification of TAES algorithm

We first examine the complexity and performance of optimal and greedy TAES algorithms, compared to the exhaustive search algorithm finding the optimal solution of **[P-SC]**. As a metric for the complexity, we use a simulation run time to compute a solution of **[P-SC]**. We collect results under various the number of BSs, the number of users and queue length. Total 200 queue samples are tested per given the number of BSs and users. Since the complexity dominantly depends on the number of BSs and the number of users, the run time results for varying queue length are averaged.

Fig. 2 shows that the average run time of TAES greedy algorithm is drastically reduced compared to the exhaustive search algorithm when the number of BSs is large. In the TAES optimal algorithm, although the average run time increases exponentially for the number of BSs, it is reduced compared to the exhaustive search by the effect of the search space reduction. Especially, when the number of users is 1, the run time is reduced similarly to TAES greedy algorithm.

Next, we verify the performance of TAES algorithms. From collected results for testing the complexity, we plot distributions of objective values achieved by TAES algorithms compared to the optimal value (i.e., achieved objective value/optimal objective value), as shown in Fig. 3. The optimal value is obtained from the exhaustive search algorithm. As it proved, TAES optimal algorithm achieves optimal values for all scenarios. TAES greedy algorithm achieves optimal values in most scenarios, and also non-optimal values are very close to optimal values compared to TAES algorithms using only one of clustering or sleeping. The greedy algorithm based on an objective function value[1] achieves comparable performance to TAES greedy algorithm, but its performance bound is unknown.

As a result, TAES optimal algorithm always achieves optimal performance but it may require high complexity while TAES greedy algorithm achieves near-optimal performance with very low complexity. Thus, we use TAES greedy algorithm in the subsequent simulations.

[1]The greedy algorithm based on an objective function value starts from a state that the entire BSs are sleep and iteratively finds a BS to be active, which makes the largest objective value when it becomes active.

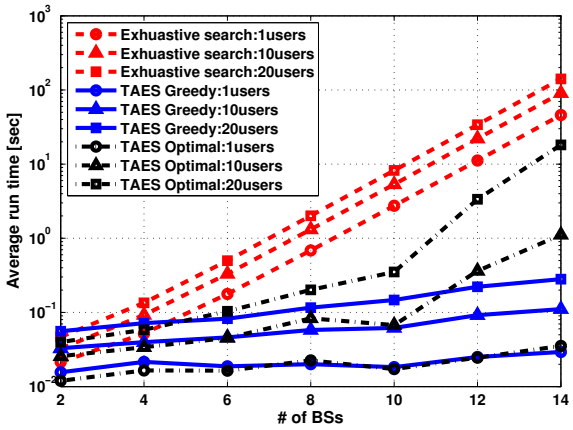


Fig. 2. Complexity comparison of TAES algorithms and the exhaustive search

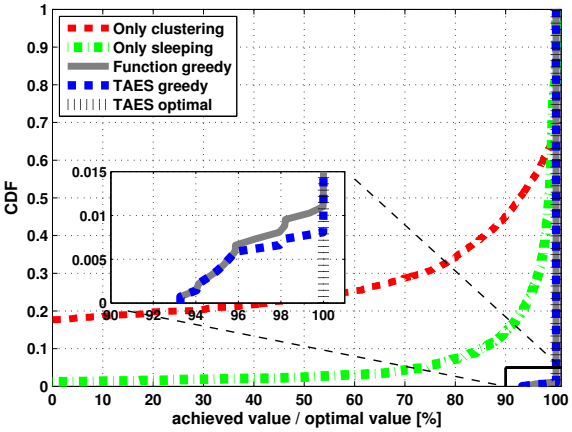
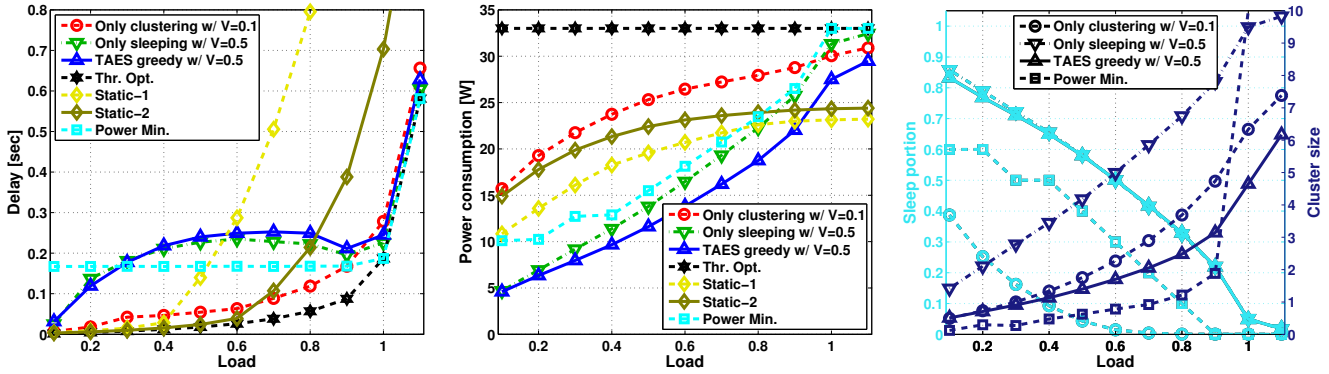


Fig. 3. Cumulative distribution function (CDF) of objective values; Samples are collected from the complexity comparison scenario of Fig. 2.

C. Traffic-aware and energy saving effect

Next, we show that TAES algorithms can reduce power consumption by adapting to traffic arrivals. We change traffic arrivals in a manner that the average traffic demand of each user, $\lambda\mu_k$, is proportional to its maximum capacity. Then, traffic loads of all users, which are defined as the average traffic demand divided by the maximum capacity, are changed identically. The maximum capacity is achieved when all BSs are active and each user forms the maximum cluster.

Fig. 4 shows two notable results on traffic-aware algorithms (only clustering, only sleeping, TAES greedy and Power Min.). First, traffic-aware algorithms can serve arrivals when the arrivals are less than the maximum capacity, although delay increases compared to throughput optimal algorithm, as shown in Fig. 4(a). Note that the static algorithms cannot serve all arrivals less than the maximum capacity. Second, traffic-aware algorithms save more power consumption as traffic load decreases, as shown in Fig. 4(b). Fig. 4(c) shows that traffic-aware algorithms determine active BS set and cluster size adaptively to traffic arrivals. This is why our algorithms outperform other algorithms. Power Min. algorithm also jointly



(a) Delay comparison (b) Power consumption comparison (c) Average sleep mode portion and cluster size

Fig. 4. Performance comparisons for varying traffic load

controls clustering and sleeping, but it reflects only average traffic arrivals without consideration of temporal fluctuation.

VI. Conclusion

In this paper, we studied the joint BS sleeping and clustering problem for energy saving in cooperative wireless networks. In order to exploit spatio-temporal fluctuation of traffic demand, we use queue dynamics and develop Traffic-Aware Energy-Saving (TAES) algorithms by applying stochastic optimization theory. In TAES algorithms, if the network capacity is excessive compared to traffic demand (and thus the network backlog decreases), then energy is saved by turning off BSs. That way, TAES algorithms save energy without capacity loss as well as they do not require any information for the future traffic variations. For BS clustering problem, we proposed an optimal algorithm that has polynomial complexity. For BS sleeping problem, we proposed TAES optimal algorithm and TAES greedy algorithm. TAES optimal algorithm finds an optimal solution with reduced search space compared to the exhaustive search. TAES greedy algorithm finds a near-optimal solution with polynomial complexity with provable optimality gap. Simulation results show that TAES algorithms can save energy up to 80% while guaranteeing the maximum capacity by adapting to traffic variations.

References

[1] Cisco System, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012-2017*. A Cisco White Paper, Feb. 2013.

[2] M. Marsan, L. Chiaraviglio, D. Ciullo, and M. Meo, "Optimal Energy Savings in Cellular Access Networks," in *Proc. IEEE International Conference on Communications Workshops*, June 2009, pp. 1–5.

[3] X. Wang, A. V. Vasilakos, M. Chen, Y. Liu, and T. T. Kwon, "A Survey of Green Mobile Networks: Opportunities and Challenges," *Mob. Netw. Appl.*, vol. 17, no. 1, pp. 4–20, Feb 2012.

[4] J. Kwak, K. Son, Y. Yi, and S. Chong, "Greening Effect of Spatio-Temporal Power Sharing Policies in Cellular Networks with Energy Constraints," *IEEE Transactions on Wireless Communications*, vol. 11, no. 12, pp. 4405–4415, December 2012.

[5] K. Son and B. Krishnamachari, "SpeedBalance: Speed-scaling-aware Optimal Load Balancing for Green Cellular Networks," in *Proc. IEEE INFOCOM*, March 2012, pp. 2816–2820.

[6] 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 Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1525–1536, September 2011.

[7] L. Saker, S.-E. Elayoubi, R. Combes, and T. Chahed, "Optimal Control of Wake Up Mechanisms of Femtocells in Heterogeneous Networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 3, pp. 664–672, April 2012.

[8] S. Han, C. Yang, and A. Molisch, "Spectrum and Energy Efficient Cooperative Base Station Doze," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 2, pp. 285–296, February 2014.

[9] P. Frenger, P. Moberg, J. Malmodin, Y. Jading, and I. Godor, "Reducing Energy Consumption in LTE with Cell DTX," in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, May 2011, pp. 1–5.

[10] G. Cili, H. Yanikomeroglu, and F. Yu, "Cell Switch off Technique Combined with Coordinated Multi-point (CoMP) Transmission for Energy Efficiency in beyond-LTE Cellular Networks," in *Proc. IEEE International Conference on Communications*, June 2012.

[11] E. Oh, B. Krishnamachari, X. Liu, and Z. Niu, "Toward dynamic energy-efficient operation of cellular network infrastructure," *IEEE Communications Magazine*, vol. 49, no. 6, pp. 56–61, June 2011.

[12] C. Peng, S.-B. Lee, S. Lu, H. Luo, and H. Li, "Traffic-driven Power Saving in Operational 3G Cellular Networks," in *Proc. ACM MobiCom*. New York, NY, USA: ACM, 2011, pp. 121–132.

[13] D. Lee, H. Seo, B. Clerckx, E. Hardouin, D. Mazzarese, S. Nagata, and K. Sayana, "Coordinated Multipoint Transmission and Reception in LTE-advanced: Deployment Scenarios and Operational Challenges," *IEEE Commun. Magazine*, vol. 50, no. 2, pp. 148–155, 2012.

[14] R. Heath, S. Peters, Y. Wang, and J. Zhang, "A Current Perspective on Distributed Antenna Systems for the Downlink of Cellular Systems," *IEEE Commun. Magazine*, vol. 51, no. 4, pp. 161–167, April 2013.

[15] S. C. Kyuho Son and G. Veciana, "Dynamic Association for Load Balancing and Interference Avoidance in Multi-cell Networks," *IEEE Trans. Wirel. Communications*, vol. 8, no. 7, pp. 3566–3576, July 2009.

[16] J. Kim, H.-W. Lee, and S. Chong, "Virtual Cell Beamforming in Cooperative Networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1126–1138, June 2014.

[17] J. Gong, S. Zhou, L. Geng, M. Zheng, and Z. Niu, "A Novel Precoding Scheme for Dynamic Base Station Cooperation with Overlapped Clusters," *IEICE Trans. Commu.*, vol. 96, no. 2, pp. 656–659, Feb. 2013.

[18] T. K. Y. Lo, "Maximum Ratio Transmission," in *Proc. IEEE International Conference on Communications*, vol. 2, 1999, pp. 1310–1314.

[19] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. Imran, D. Sabella, M. Gonzalez, O. Blume, and A. Fehske, "How much energy is needed to run a wireless network?" *IEEE Wireless Communications*, vol. 18, no. 5, pp. 40–49, October 2011.

[20] Y. Yao, L. Huang, A. Sharma, L. Golubchik, and M. Neely, "Data Centers Power Reduction: A Two Time Scale Approach for Delay Tolerant Workloads," in *Proc. IEEE INFOCOM*, March 2012.

[21] J. Kwak, O. Choi, S. Chong, and P. Mohapatra, "Dynamic Speed Scaling for Energy Minimization in Delay-tolerant Smartphone Applications," in *Proc. IEEE INFOCOM*, April 2014, pp. 2292–2300.

[22] J. Kim, H.-W. Lee, and S. Chong, "TAES: Traffic-Aware Energy-Saving Base Station Sleeping and Clustering in Cooperative Networks," *Technical Report*, 2015. [Online]. Available: http://netsys.kaist.ac.kr/publication/papers/Resources/R8.pdf

[23] S. Boyd and L. Vandenbergh, *Convex Optimization*. Cambridge University Press, 2004.